TABLE OF CONTENTS

Contributed Articles

Association for Computing Machinery

Advancing Computing as a Science & Profession

Beyond Tokens: A Survey on Decoding Methods for Large Language and Vision-Language Models

Haoran Wang[1], Xiongxiao Xu[2], Philip S. Yu[3], Kai Shu[1]
[1] Emory University, [2] Illinois Institute of Technology, [3] University of Illinois Chicago
haoran.wang@emory.edu, xxu85@hawk.illinoistech.edu, psyu@uic.edu, kai.shu@emory.edu

ABSTRACT

Large language models (LLMs) and large vision-language models (LVLMs) have demonstrated impressive generative capabilities, yet ensuring their outputs align with user intent is still challenging. While most existing approaches address this issue at the training stage, inference-time approaches like decoding methods offer a more efficient and scalable solution. Decoding methods control model generation by guiding token-level selection, performing sequence-level generation, or generating tokens in parallel to accelerate the process. In this survey, we identify three emerging paradigms from recent works on decoding methods for LLMs and LVLMs, provide a systematic review of these methods, highlight ongoing challenges, and discuss potential future research directions. Our goal is to underscore the efficiency and effectiveness of decoding methods and offer a practical view of their applications. Paper lists and more resources on decoding methods for LLMs and LVLMs can be found at https://github.com/wang2226/Awesome-LLM-Decoding.

1. INTRODUCTION

Large language models (LLMs) and large language-vision models (LVLMs) have demonstrated that scaling up both model size and training datasets can greatly improve the model's generative capabilities. However, with the rise of massive foundational models such as Llama3 405B [87] and Megatron [118], which boasts 530 billion parameters, the focus has increasingly shifted toward developing *more efficient and scalable approaches* to control generation during inference time (§ 2). One popular approach is prompt engineering, due to its simplicity and effectiveness. However, it is highly task-specific, requiring human expertise to craft optimal prompts, and is susceptible to prompt sensitivity issues [109; 83], where minor variations in prompt wording can lead to significant performance differences. Other techniques, such as ROME [85] and ITI [64], modify model internals at inference time but suffer from limited generalizability and scalability. Recently, there has been increasing interest in decoding methods, which play a critical role in controlling next-token prediction and text generation.

Decoding methods have a rich history in language modeling, ranging from greedy decoding and beam search to sampling-based approaches like top-p sampling [43]. These methods transform the vector representations produced by the model into coherent text while controlling the quality and attributes of the generated output. Recent works have shown that adopting more advanced decoding strategies can effectively *mitigate hallucination* [20; 148; 185; 144; 33], *improve safety* [73; 182; 161], *enhance visual grounding* [25; 59], *improve reasoning* [159; 186], and *increase robustness against noisy context* [116; 55; 104]. Beyond improving text generation, decoding methods can also *provide interpretability of the models*. For instance, [149] showed that modifying the decoding process can elicit chain-of-thought reasoning paths from LLMs. Finally, a recent line of research [119; 155; 150; 61; 15] has focused on *improving the efficiency of LLM and LVLM generation* by decoding multiple tokens simultaneously to accelerate generation.

In this survey, we use decoding to denote inference-time procedures that transform model output distributions into output sequences, including token selection, search strategies, and parallel generation. While decoding is operationally part of generation in autoregressive models, we distinguish it from training-time alignment and prompt engineering, and focus specifically on inference-time control mechanisms.

This survey provides a comprehensive overview of advanced decoding methods for LLMs and LVLMs, highlighting their capabilities, applications, and potential. We first review inference-time generation control methods (§ 2) and classical decoding strategies (§ 3). We then introduce three modern decoding paradigms (§ 4), followed by their applications (§ 5). Finally, we discuss open challenges and future directions (§ 6). Figure 1 presents a typology of key concepts related to decoding methods in LLMs and LVLMs. Although we cover decoding methods for both LLMs and LVLMs, the current literature is substantially richer for text-based LLM decoding. Accordingly, this survey primarily emphasizes text decoding, while visual, video, and code decoding are discussed as emerging extensions.

Major Paradigm Shift in Language Modeling As highlighted by [75], the field of language modeling and its related tasks has undergone several paradigm shifts. Initially, there was a shift from fully supervised learning to the *pre-train and fine-tune* approach [105; 99; 30], largely driven by the success of pre-trained LMs like BERT [27]. More recently, there has been another major shift toward the *pre-train, prompt, and predict* paradigm, where prompt engineering [106; 7; 107] has become a popular approach for adapting LMs to downstream tasks through carefully designed prompts.

However, both fine-tuning and prompt engineering face challenges when applied to LLMs and LVLMs. Fine-tuning models with tens of billions of parameters requires substan-

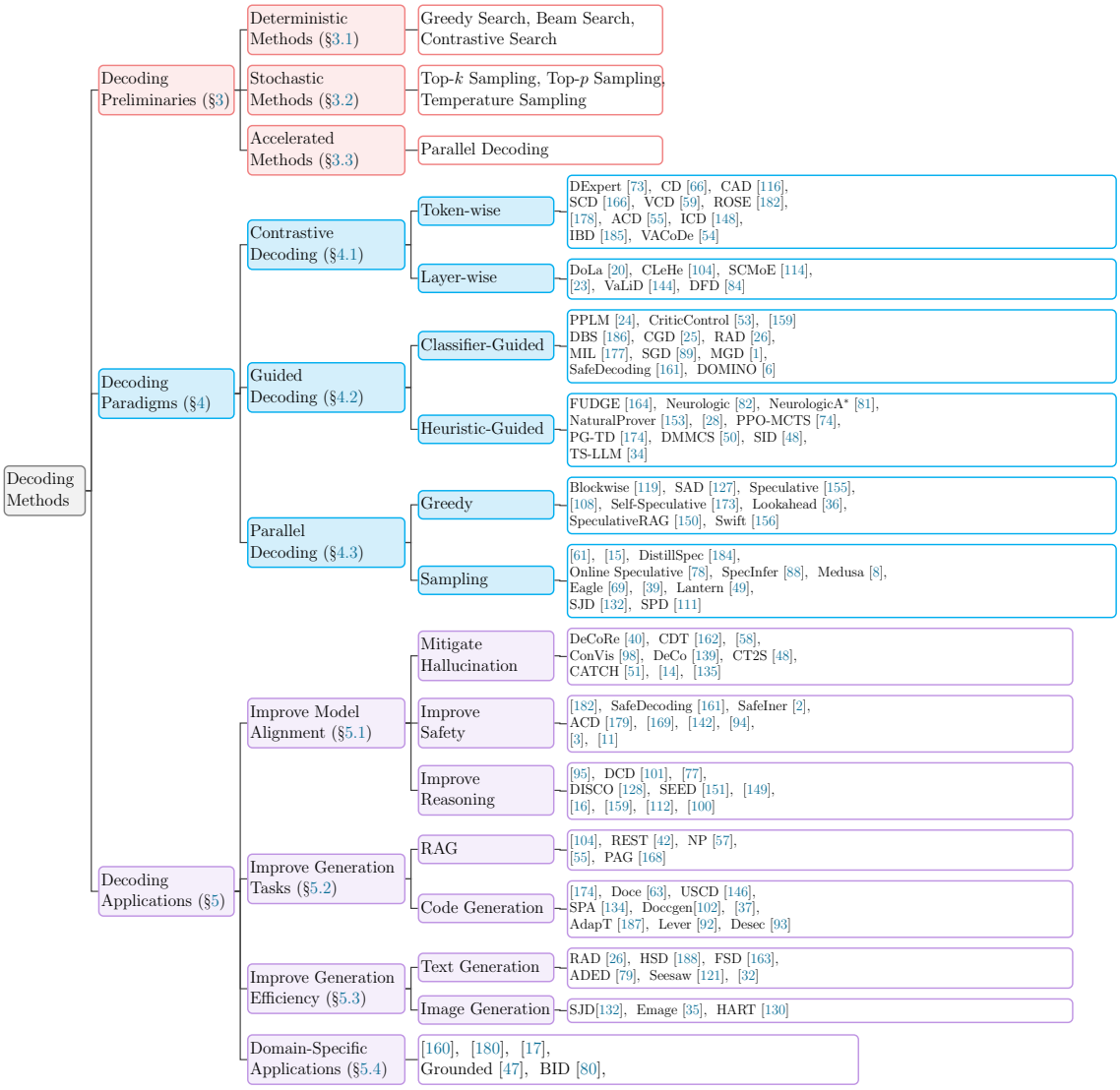

Figure 1: Typology of decoding methods for LLMs and LVLMs.

tial computational resources, raising concerns about energy consumption, scalability, and accessibility. While prompt engineering is computationally efficient, it remains highly task-specific and requires expert knowledge to craft effective prompts. Moreover, LLMs have been shown to have prompt sensitivity issues [109; 83], where even slight changes in prompt format can result in considerable variations in performance. Recently, a growing body of work has focused on advanced decoding methods [29; 154]. Figure 2 illustrates the evolution of decoding methods, highlighting key milestones from classical search to modern contrastive and speculative paradigms. We argue that the field is undergoing another significant paradigm shift toward universal, effective, and scalable methods for controlling LLM generation [70].

2. INFERENCE METHODS TO IMPROVE GENERATION

LLMs and LVLMs have demonstrated incredible performance across a wide range of NLP tasks. However, bridging the gap between their training objectives and user expectations remains challenging. While LLMs are trained to minimize contextual word prediction errors using large datasets, users expect the models to follow instructions in a helpful and safe manner. This highlights the need for alignment [113; 125] to ensure that models behave in ways that align with human values. To achieve controllable text generation and generate aligned outputs, researchers have proposed various methods, which can be broadly categorized into training-stage and inference-stage approaches, as outlined by [70]. Training-stage methods include fine-tuning [167; 172; 183; 171; 181; 152] and reinforcement learning [120; 96; 22; 52; 137; 170], which leverage reward signals to guide model outputs toward specific control objectives.

In this section, we focus on methods that improve generation during the inference stage for three key reasons. First, inference-time methods enable real-time improvements without the need for re-training or altering the underlying model, making them an *efficient approach* for controlling genera-

tion across models of any size. Second, these methods are typically *model-agnostic*, meaning they can be applied to most decoder-only transformers, thus enhancing their versatility. Finally, some inference-time techniques like decoding methods offer better *interpretability*. We categorize these inference-time approaches into three categories: *(1) prompting*, *(2) latent space manipulation*, and *(3) decoding algorithms*.

2.1 Prompt Engineering

Prompt engineering directly controls text generation by crafting specific prompts for the task. The primary goal of this approach is to guide model outputs by providing clear instructions or examples. [117] introduced AutoPrompt, an automated method for creating prompts across a wide range of tasks using a gradient-guided search approach. To provide a lightweight alternative to fine-tuning, [67] proposed prefix-tuning, which optimizes a sequence of continuous, task-specific vectors known as the "prefix" for natural language generation tasks. Additionally, [60] introduced prompt tuning, a straightforward yet effective technique for learning soft prompts that condition frozen language models to perform specific downstream tasks. More recently, [72] proposed Direct Large Model Alignment (DLMA), an automatic alignment method that generates preference data using contrastive prompt pairs, calculates a self-rewarding score, and applies the DPO algorithm to align LLMs. Black-Box Prompt Optimization (BPO) [18] optimizes user prompts to align with LLMs' input understanding, achieving the user's intent without modifying model parameters. By leveraging human preferences, BPO outperforms traditional prompt engineering in aligning LLMs with user goals. Moving beyond text, [62] proposed ALPRO, a video-text pre-training framework that aligns features without explicit object detectors.

2.2 Latent Space Manipulation

Latent space manipulation modifies the model's internal structure for controlled generation, such as attention heads or adding steering vectors to the activation layers. The core idea is that the necessary information to generate the target output is already encoded within the model's structure, eliminating the need for re-training or fine-tuning. By operating directly on the latent space, these techniques enhance output accuracy, diversity, and coherence, while remaining computationally efficient.

[12] introduced GENhance, a generative framework that enhances attributes through a learned latent space. Additionally, [124] extracts latent vectors, called steering vectors, directly from PLM decoders without fine-tuning. When added to the model's hidden states, these vectors allow for controlled generation. Extending steering vectors to LLMs, [136] introduced activation engineering, a method for modifying activations during inference to steer model outputs. This approach can also be utilized to control alignment [10; 145; 9; 141; 103].

To effectively control in-context learning, [76] proposed a method that first creates the in-context vector from the latent embedding of the LLM, and then shifts the latent states of the LLM using these vectors to more effectively follow the demonstration examples. Furthermore, [56] explores strategies to steer LLM outputs toward specific styles, such as sentiment, emotion, or writing style, by incorporating style vectors into the activations of hidden layers during text generation.

On a separate line of work, [85] analyzed the storage and recall of factual associations in autoregressive transformer language models, finding that these associations correspond to localized, directly editable computations. They modify feed-forward weights using Rank-One Model Editing (ROME) to alter factual associations within LLMs. Subsequently, [86] developed MEMIT, a method for directly updating a language model with many memories, demonstrating its ability to scale to thousands of associations for LLMs. More recently, [64] introduced Inference-Time Intervention (ITI) to enhance the truthfulness of LLMs. Specifically, it shifts model activations during inference, guided by a set of directions across a limited number of attention heads.

2.3 Decoding Algorithm

Decoding algorithms are applied during the decoding phase of transformer-based generative models to modify the logits or probability distribution of the model's output. It guides the generated text toward desired attributes by adjusting these probabilities, offering dynamic control over the text generation process, and ensuring the output aligns with specific requirements. Methods such as temperature scaling, top-k sampling, and nucleus sampling can be used to influence the diversity, creativity, or coherence of the generated text by altering the probability distribution.

3. PRELIMINARIES OF DECODING STRATEGIES

This section reviews classical token-level decoding strategies such as greedy search, beam search, and sampling, which form the foundational building blocks for modern decoding approaches. We define key concepts and outline the general objectives of text generation. **Auto-regressive Language Generation** operates by predicting the next token in the sequence based on the preceding tokens. Formally, considering a sequence of tokens $w = (w_1, w_2, ..., w_t)$, the probability distribution of a word sequence can be decomposed into the product of conditional next word distributions:

$$P(w_{1:T}|W_0) = \prod_{t=1}^{T} P(w_t|w_{1:t-1}, W_0)$$

with W_0 being the initial context word sequence. The length T of the word sequence is usually determined on-the-fly and corresponds to the timestamp $t = T$ the EOS token is generated from $P(w_t|w_{1:t-1}, W_0)$. The decoding problem is equivalent to selecting the most probable sequence given the probability distribution.

3.1 Deterministic Methods

Deterministic methods generate text by selecting the continuation with the highest probability determined by the LM. However, these methods often lead to model degeneration, where the output becomes unnatural, marked by repetitive and overly predictable language. As a result, the text lacks variety and fails to reflect natural human expression.

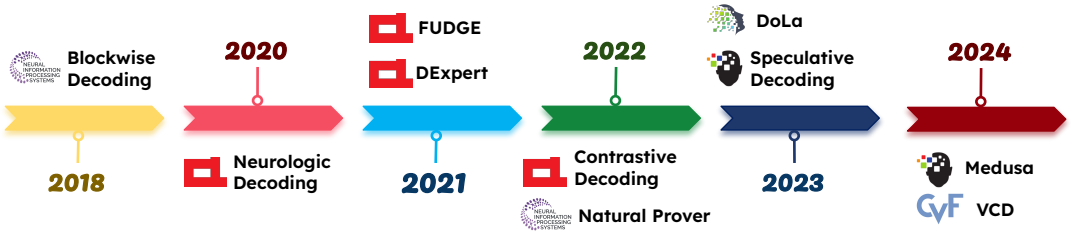

Figure 2: Timeline illustrating the evolution of decoding methods for LLMs and LVLMs.

3.1.1 *Greedy Search*

Greedy search selects the word with the highest probability as its next word $\underset{w}{\text{argmax}} P(w|w_{1:t-1})$ at each timestep t until reaching either an end-of-sequence (EOS) token or a maximum timestep T. The primary drawback of greedy search is that it may overlook high-probability words that are accessible only after a lower-probability word. As a result, greedy decoding does not formally guarantee a global optimum of the decoding objective, as its choices are only locally optimal. Despite its simplistic approach, greedy decoding remains a widely used generation algorithm. For example, it is employed in Google's Gemini report [131] and is commonly available in standard language model APIs.

3.1.2 *Beam Search*

Beam search reduces the risk of overlooking high-probability word sequences by maintaining a fixed number of the most likely hypotheses (or "beams") at each timestep, ultimately selecting the hypothesis with the highest overall probability. While beam search consistently finds output sequences with higher probabilities than greedy search, it does not guarantee the most likely output. Beam search performs well in tasks where the desired generation length is relatively predictable, such as machine translation or summarization [90; 165]. However, this approach is less suited for open-ended generation tasks, like dialogue or story generation, where the desired output length can vary significantly.

Beam search is prone to generating repetitive outputs. To address the lack of diversity in the generated sequences, [138] proposed Diverse Beam Search (DBS). This algorithm divides the beam into multiple sub-groups and introduces an inner iteration at each timestep to maximize diversity between these groups.

3.1.3 *Contrastive Search*

Contrastive search [123; 122] mitigates string repetition by penalizing the selection of previously generated token sequences. This method can suppress repetitions more effectively than beam search while utilizing a comparable amount of computational resources.

3.2 Stochastic Methods

Human-generated text exhibits greater variance in token probabilities, reflecting a diverse range of word choices, often unexpected. In contrast, the output from deterministic methods shows minimal variance, resulting in more predictable and potentially repetitive text. To address these limitations, stochastic approaches introduce randomness during the decoding process, leading to more diverse and natural text generation. Commonly used stochastic techniques include top-k sampling, top-p sampling, and temperature scaling.

3.2.1 *Top-k Sampling*

In Top-k sampling [31; 44], the k most probable next words are selected, and their probability mass is redistributed to form a new distribution. Specifically, the method first identifies the top k tokens with the highest probabilities for the current sequence. The probabilities of these k tokens are then normalized to sum to 1, resulting in a truncated distribution. A token is then randomly sampled from this distribution and appended to the current sequence. This process is repeated iteratively until a termination condition is satisfied. GPT-2 employed this sampling strategy, which significantly contributed to its effectiveness in story generation.

While Top-k sampling is both effective and significantly more efficient than beam search, it has limitations in specific scenarios, particularly in two edge cases. First, when the next-token distribution is widely spread and approaches a uniform distribution, Top-k sampling may arbitrarily exclude numerous potentially interesting tokens, thereby reducing the diversity of the generated text. Conversely, when the distribution is highly concentrated, Top-k sampling might either include unnecessary tokens if k is too large or exclude equally probable ones if k is too small.

The primary challenge with Top-k sampling is how to determine an optimal k value. The ideal choice of k can vary based on the context and the shape of the probability distribution at each step. Using a fixed k value may prove too restrictive in some scenarios while being overly permissive in others.

3.2.2 *Top-p (Nucleus) Sampling*

To address the limitations of Top-K sampling, where restricting the sample pool to a fixed size K can lead to gibberish outputs for sharp distributions and stifle creativity for flat distributions, [43] proposed Top-P sampling. This method selects the smallest set of words whose cumulative probability meets or exceeds a predefined threshold p. Rather than sampling exclusively from the top k most probable words, Top-p redistributes the probability mass across this dynamic set. This adaptive approach allows the size of the word set (i.e., the number of included words) to increase or decrease based on the shape of the next-token probability distribution.

3.2.3 *Temperature Sampling*

Temperature sampling is among the most commonly used decoding strategies. Determining the optimal temperature

value typically involves ad-hoc experimentation tailored to the specific application. The core idea of temperature sampling is to control the "sharpness" of the probability distribution by introducing a temperature parameter t. This parameter is applied in the softmax function after the transformer's final layer to compute token probabilities. The temperature t directly influences the level of randomness in the sampling process, with higher values increasing randomness and lower values reducing it.

3.3 Speculative Decoding

In addition to output quality, decoding strategies must also consider inference efficiency due to the ever-growing size of models. One of the main drawbacks of autoregressive decoding is that its token-by-token generation can lead to increased inference latency, scaling with both the length of the generated sequence and the model's size. To accelerate inference for LLMs, speculative decoding [119; 61; 15] has been introduced, allowing for the simultaneous decoding of multiple tokens per step. Specifically, in each decoding step, speculative decoding first efficiently drafts multiple tokens as speculation for future decoding steps of the target LLM, and then utilizes the LLM to verify all drafted tokens in parallel. Only those tokens that meet the LLM's verification criteria are accepted as final outputs, ensuring generation quality [157].

4. DECODING PARADIGMS

Earlier decoding methods (§ 3) primarily focused on token-level diversity and fluency. More recent work extends decoding toward sequence-level control, structured guidance, and generation efficiency. In this survey, we identify three paradigms in recent decoding methods for LLMs and LVLMs: **contrastive decoding**, **guided decoding**, and **parallel decoding**. Table 1 lists recent works categorized by their decoding paradigms. Unlike earlier token-centric decoding methods, these paradigms emerge from recent efforts that explicitly optimize sequence-level quality, controllability, or efficiency.

4.1 Contrastive Decoding

The primary goal of contrastive decoding is to enhance the quality of generated outputs by contrasting positive and negative examples during the decoding process. Unlike greedy or sampling-based decoding methods, which offer only basic control at the token level, contrastive decoding operates at both the token and layer levels, allowing for greater control over the quality of the entire generated sequence.

Definition 1 (Contrastive Decoding). Contrastive decoding (CD) searches for the next token that maximizes a weighted difference in likelihood between two logits z^+ and z^-.

$$P(W_t|w_{<t}) = softmax(z^+ + \alpha(z^+ - z^-))$$

where α controls the strength of the modification. Contrastive decoding methods can be broadly classified into two categories based on the level at which contrastive examples are utilized: *token-wise CD* and *layer-wise CD*.

4.1.1 Token-wise Contrastive Decoding

Token-wise CD contrasts the token-level probability distributions produced by pairs of contrasting examples. These contrasting examples can be model-generated. Specifically, an expert model generates logits z^+, representing the user-desired direction while a weak or base model generates logits z^-, providing a baseline probability distribution. The contrastive decoding mechanism then adjusts token probabilities by weighting the difference between z^+ and z^-, regulated by the parameter α. For example, DExpert [73] utilizes both expert and anti-expert models to guide output for language detoxification and sentiment-controlled generation. During the decoding process, each token is assigned a high probability if it is deemed likely by the expert LM and unlikely by the anti-expert LM. Similarly, [66] proposed a contrastive method that directly compares off-the-shelf LMs by computing the difference between their log probabilities. With the increasing size of LLMs, the efficiency benefits of advanced decoding methods have become more apparent. ROSE [182] applies contrastive decoding to pairs of carefully crafted reverse prompts to enhance LLM safety. Addressing the role of contextual information in text generation, [116] proposed context-aware decoding (CAD), which uses a contrastive output distribution to highlight differences in token probabilities when the model generates text with and without contextual input. Building on this, [178] developed a method that combines contrastive decoding with adversarial irrelevant passages as negative samples, improving robust contextual grounding by contrasting relevant and irrelevant contexts. Extending these approaches to noisy contexts, adaptive contrastive decoding (ACD) [55] effectively leverages contextual influences to address challenges posed by noisy or imperfect inputs. To further enhance the efficiency of contrastive decoding, [166] introduced Speculative Contrastive Decoding (SCD), a simple yet powerful approach that utilizes predictions from smaller LMs to achieve faster decoding while maintaining generation quality.

In the context of LVLMs, various CD methods have been proposed to enhance accuracy and mitigate hallucination. Visual Contrastive Decoding (VCD) [59] ensures visual grounding by comparing outputs from original and distorted visual inputs. Image-biased decoding (IBD) [185] contrasts predictions from conventional and image-biased LVLMs to improve image-text alignment and reduce hallucinations. Instruction Contrastive Decoding (ICD) [148] refines outputs by comparing distributions from standard and perturbed instructions, filtering hallucinated concepts. More recently, Visual Augmented Contrastive Decoding (VACoDe) [54] introduced an adaptive approach, selecting optimal augmentations per task via a novel softmax distance metric, surpassing single-augmentation methods.

4.1.2 Layer-wise Contrastive Decoding

Recent studies have shown that transformer models tend to encode lower-level information, such as part-of-speech tags, in the earlier layers, whereas more semantic information is encoded in the later layers [133]. For instance, [21] found that knowledge neurons are concentrated in the topmost layers of the BERT model. Moreover, [85] demonstrated that factual knowledge can be edited by manipulating a specific set of feedforward layers within an autoregressive model. Building on these observations, contrastive decoding has been extended to operate on layers within the models, besides just token-level adjustments.

Decoding by Contrasting Layers (DoLa) [20] enhances fac-

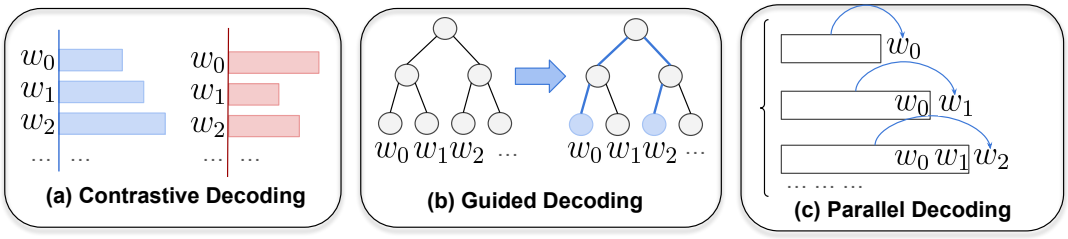

Figure 3: Illustration of different decoding paradigms. **Contrastive decoding** selects the next token by maximizing the contrast between two underlying probability distributions. **Guided decoding** determines the next token based on the highest score from a guidance function. **Parallel decoding** generates multiple token candidates simultaneously and selects the most probable one.

tual knowledge in LMs by leveraging modular encoding and contrastive decoding. It derives the next-token distribution by contrasting logits from different transformer layers projected onto the vocabulary space, based on the observation that factual knowledge is localized in specific layers. In a similar fashion, [23] proposed an entropy-guided method to extrapolate token probabilities beyond the last layer for more accurate contrastive decoding, instead of relying solely on the final layer. To reduce noise from retrieved context, [104] employed an entropy-based document-parallel ensemble decoding, which prioritizes low-entropy distributions from retrieved documents. Specifically, it compares this low-entropy ensemble with the model's high-entropy internal distribution, emphasizing reliable external information. By leveraging the modular and hierarchical nature of factual knowledge within LLMs, Dynamic Focus Decoding (DFD) [84] adaptively adjusts the decoding focus based on distributional differences across layers.

More recently, [114] addressed the underutilization of Mixture-of-Experts (MoE) models, where unchosen experts do not contribute to the output. They proposed Self-Contrast Mixture-of-Experts (SCMoE), an inference strategy that improves performance by contrasting strong and weak expert activations within the model. For LVLMs, [144] found that key visual features in early layers often distort as they propagate to the output. Building on this finding, they introduced Visual Layer Fusion Contrastive Decoding (VaLiD), which uses uncertainty to guide visual layer selection, reducing distortions and mitigating hallucinations.

4.2 Guided Decoding

Selecting the right decoding "path" can greatly enhance the generation quality of LLMs and LVLMs. For example, chain-of-thought reasoning can emerge simply by modifying the decoding process [149]. To determine the optimal decoding path," various strategies have been proposed to guide the decoding process. We refer to these strategies collectively as guided decoding, which is defined as follows:

Definition 2 (Guided Decoding). Guided decoding (GD) searches for the next token that maximizes the score from a guidance function $\mathcal{G}$.

$$P'(w_t|w_{<t}) \propto \mathcal{G}(P(w_t|w_{<t}), C)$$

where $\mathcal{G}$ is a guidance function that adjusts the model's distribution based on certain criteria, and C denotes the control condition. We classify guided decoding into two main categories: *classifier-guided* and *heuristic-guided* decoding.

4.2.1 *Classifier-Guided Decoding*

Classifier-guided decoding utilizes an external classifier to influence the decoding process, allowing for control over specific attributes during text generation. The classifier can take various forms, such as a reward model, a pre-trained model, or an API, and adjust the model's output accordingly.

By using an attribute model to guide the decoding process, Plug and Play Language Model (PPLM) [24] controls text generation by combining a pre-trained LM with one or more simple attribute classifiers. To detoxify PLMs at the token level, [177] proposed Multiple Instance Learning Decoding (MIL-Decoding), which computes token-level toxicity scores and adjusts probabilities dynamically based on context. CriticControl [53] combines reinforcement learning and weighted decoding, using an LM-steering critic for controlled text generation. Similarly, Reward-Augmented Decoding (RAD) [26] modifies token probabilities using a unidirectional reward model, optimizing sampling for better attribute control.

To address uncertainty in multi-step reasoning, [159] proposed a decoding algorithm using self-evaluation guidance through stochastic beam search, improving prediction quality by enhancing search efficiency. To reduce reasoning errors in intermediate steps, Deductive Beam Search (DBS) [186] integrates chain-of-thought and deductive reasoning with a step-wise beam search and a verifier to check the validity of each step, reducing error accumulation. SafeDecoding [161] mitigates jailbreak risks by guiding the decoding process with a trained expert model to generate helpful and safe responses. For improved code generation in LLMs, [1] introduced monitor-guided decoding (MGD), where a monitor uses static analysis to guide decoding. Lastly, [6] proposed DOMINO, a decoding algorithm that enforces subword-aligned constraints with minimal overhead, addressing challenges in aligning sub-word tokens and grammar terminals.

For LVLMs, CLIP-Guided Decoding (CGD) [25] enhances visual grounding by using CLIP to guide the decoding process, with CLIP similarity to the image serving as a stronger indicator of hallucinations than token likelihoods. [33] introduced dropout decoding, inspired by dropout regularization, to reduce hallucinations by quantifying and masking uncertain visual tokens during inference. Summary-Guided Decoding (SGD) [89] addresses hallucinations by shortening token length through summarization, promoting greater visual detail while controlling image-related part-of-speech tokens to maintain text quality.

4.2.2 *Heuristic-Guided Decoding*

To provide more effective guidance within the token search space, heuristics are used to steer the decoding process with specific rules or search strategies. These heuristics could include constraints, results from lookahead search, or value functions.

Future Discriminators for Generation (FUDGE) [164] adjusts the probability distribution during generation by predicting the attribute probability of the evolving sequence and modifying the logits to align with desired attributes. To address the challenges PLMs face in adhering to constraints during generation, [82] proposed Neurologic Decoding, which enforces lexical constraints during decoding. [81] later improved it with A*-inspired lookahead heuristics for better performance. For generating mathematical proofs, [153] developed NaturalProver, a knowledge-grounded model that generates proofs by conditioning on background theorems and definitions, optionally enforcing them through constrained decoding. Lastly, [28] introduced Adaptive Decoding, a neural module that predicts the optimal sampling temperature for specific tasks.

Using heuristics from lookahead search results, Planning-Guided Transformer Decoding (PG-TD) [174] leverages a planning algorithm to conduct lookahead searches and guide the model in generating more effective programs for code generation with LLMs. To leverage value models trained as byproducts when aligning LMs with human preferences, [74] proposed an effective method for applying Monte-Carlo tree search decoding on top of PPO-trained policy and value models. This approach integrates the value network from PPO, enabling it to collaborate closely with the policy network during inference-time generation. More recently, TS-LLM [34] leverages tree search with a learned value function to guide LLM decoding, enhancing reasoning, planning, and decision-making capabilities. Integrative Decoding [19] improves actuality by implicitly incorporating self-consistency within its decoding objective.

To mitigate hallucinations in LVLMs, [48] introduced Self-Introspective Decoding (SID) with the Context and Text-aware Token Selection (CT2S) strategy. SID retains only the least important vision tokens after the early decoder layers, adaptively addressing hallucinations in vision-and-text associations during autoregressive decoding. This approach leverages the ability of pre-trained LVLMs to introspectively evaluate the significance of vision tokens based on prior vision, text, and generated content. For diagnostic captioning, [50] proposed Distance from Median Maximum Concept Similarity (DMMCS), a data-driven guided decoding method that incorporates medical image tags to generate more accurate diagnostic text.

4.3 Parallel Decoding

Unlike standard sequential decoding, parallel decoding executes multiple decoding processes concurrently where it first generates multiple candidate sequences simultaneously and then selects the most likely one based on a set of predefined criteria.

Definition 3 (Parallel Decoding). Parallel decoding first decodes multiple future tokens $y_1, y_2, ..., y_m$ simultaneously, a process often referred to as *drafting*.

$$\begin{cases} y_1 = \arg\max P(w_t|w_0) \\ y_2 = \arg\max P(y_2|y_1, w_0) \\ \vdots \\ y_m = \arg\max P(y_m|y_{1:m-1}, w_0) \end{cases}$$

Then, it aggregates these tokens $y_1, y_2, ..., y_m$ in parallel using the target LLM to speed up inference – a process known as *verification*.

$$P(w_t|w_{<t}) = \mathcal{M}(w|w_{\leq t}, \hat{w}_{\leq i}), i = 1, ..., K+1$$

where $\mathcal{M}$ is the draft model, and $\hat{w_1}$, $\hat{w_2}$, ..., $\hat{w_K}$ are the draft tokens outputted by $\mathcal{M}$. This *draft-then-verify* paradigm can be further classified into two categories: *greedy* and *sampling*.

4.3.1 *Greedy*

To enhance the decoding process in deep autoregressive models, [119] introduced blockwise decoding, a parallel approach in which predictions are made for multiple time steps simultaneously, followed by a rollback to the longest valid prefix as determined by a scoring model. To improve online inference efficiency in transformer-based models for instantaneous grammatical error correction, Shallow Aggressive Decoding (SAD) [127] uses a shallow decoder to aggressively decode as many tokens as possible in parallel. To overcome the efficiency limitations of autoregressive decoding in transformers, [108] redefine standard greedy autoregressive decoding for machine translation by adopting a parallel formulation that uses Jacobi and Gauss-Seidel fixed-point iteration methods to achieve faster inference.

Notably, [155] introduced Speculative Decoding (SpecDec), a method to accelerate autoregressive decoding through speculative execution, or draft-then-verify. It consists of two components: Spec-Drafter, an independent model for efficient token drafting, and Spec-Verification, a mechanism for validating the drafted tokens. To enhance inference efficiency in RAG, [150] proposed SpeculativeRAG, a framework in which a larger generalist LM verifies multiple RAG drafts generated in parallel by a smaller, distilled specialist LM. Each draft, based on distinct subsets of retrieved documents, reduces token counts and offers diverse perspectives, improving comprehension, mitigating position bias, and speeding up the RAG process by limiting the generalist LM to a single verification pass.

To eliminate the need for separate draft and verifier models, self-speculative decoding [173] uses a single LLM for both drafting and verification, avoiding additional training and memory overhead. More recently, [36] introduced Lookahead Decoding, a parallel algorithm that accelerates LLM decoding without relying on auxiliary models or data stores. This approach balances per-step log (FLOPs) with the total decoding steps, making it highly parallelizable on modern accelerators and compatible with memory-efficient attention mechanisms like FlashAttention. Lastly, [121] proposed a technique to facilitate dynamic reconfiguration of parallelization strategies across prefilling and decoding stages to improve the efficiency of distributed LLM inference.

4.3.2 *Sampling*

Similar to how stochastic decoding methods often surpass

Table 1: List of works categorized by the three paradigms identified in this survey. Based on chronological order.

Paradigm	Work	Description	Model	Year (↓)
Contrastive Decoding	DExpert [73]	Detoxification with contrastive decoding	PLM	2021
	CD [66]	Formulate Contrastive Decoding	PLM	2022
	CAD [116]	Context-aware Decoding	LLM	2023
	SCD [166]	Speculative contrastive decoding	LLM	2023
	DoLa [20]	Decoding by contrasting layers	LLM	2023
	VCD [59]	Visual contrastive decoding	LVLM	2024
	ROSE [182]	Reverse prompt contrastive decoding	LLM	2024
	Zhao et al. [178]	Multi-input contrastive decoding	LLM	2024
	CLeHe [104]	Entropy-based contrastive decoding	LLM	2024
	ACD [55]	Adaptive contrastive decoding	LLM	2024
	SCMoE [114]	Self-contrast Mixture-of-Experts	LLM	2024
	Das et al. [23]	Entropy guided extrapolative decoding	LLM	2024
	ICD [148]	Instruction contrastive decoding	LVLM	2024
	IBD [185]	Image-biased decoding	LVLM	2024
	VACoDe [54]	Visual augmented contrastive decoding	LVLM	2024
	VaLiD [144]	Visual Layer Fusion contrastive decoding	LVLM	2024
Guided Decoding	PPLM [24]	Attribute model guidance	PLM	2020
	Neurologic [82]	Constrained decoding	PLM	2020
	FUDGE [164]	Future discriminator guidance	PLM	2021
	Neurologic$A^\star$ [81]	Lookahead heuristics guidance	PLM	2021
	CriticControl [53]	Critic-guided decoding	PLM	2022
	NaturalProver [153]	Stepwise constrained decoding	PLM	2022
	MIL-Decoding [177]	Multiple instance learning guidance	PLM	2023
	RAD [26]	Reward augmented decoding	LLM	2023
	PPO-MCTS [74]	Value-guided Monte-Carlo tree search	LLM	2023
	PG-TD [174]	Planning-guided decoding	LLM	2023
	Xie et al. [159]	Self-evaluation guided beam search	LLM	2023
	DBS [186]	Decoding deducible rationale for CoT	LLM	2024
	DMMCS [50]	Data-driven guided decoding	LVLM	2024
	SGD [89]	Summary-Guided decoding	LVLM	2024
	SID [48]	Self-Introspective decoding	LVLM	2024
	TS-LLM [34]	AlphaZero-like tree-search	LLM	2024
	Dropout [33]	Dropout decoding	LVLM	2024
	MGD [1]	Monitor-guided decoding	LLM	2024
	SafeDecoding [161]	Expert model guidance	LLM	2024
	DOMINO [6]	Minimally-Invasive constrained decoding	LLM	2024
	CGD [25]	Clip-guided decoding	LVLM	2024
	DFD [84]	Dynamic Focus Decoding	LLM	2025
	AttnReal [135]	Attention reallocation	LVLM	2025
Parallel Decoding	Blockwise [119]	Blockwise parallel decoding	PLM	2018
	SAD [127]	Shallow aggressive decoding	PLM	2021
	SpecDec [155]	Speculative decoding for seq2seq generation	PLM	2023
	Santilli et al. [108]	Hybrid GS-Jacobi decoding	PLM	2023
	Self-speculative [173]	Self-speculative decoding	LLM	2023
	Speculative [61]	Speculative sampling	LLM	2023
	Speculative [15]	Speculative sampling with distributed serving	LLM	2023
	DistillSpec [184]	Speculative via knowledge distillation	LLM	2023
	SpecInfer [88]	Tree-based speculative verification	LLM	2023
	Online Speculative [78]	Online Speculative	LLM	2023
	Speculative RAG [150]	Speculative RAG	LLM	2024
	Lookahead [36]	Lookahead decoding	LLM	2024
	Medusa [8]	Multiple decoding heads	LLM	2024
	Eagle [69]	Extrapolative speculative sampling	LLM	2024
	Gagrani et al. [39]	Multimodal speculative	LVLM	2024
	Lantern [49]	Latent neighbor token acceptance relaxation	LVLM	2024
	SJD [132]	Speculative Jacobi decoding	LVLM	2024
	SPD [111]	Superposed decoding	LLM	2024
	Swift [156]	On-the-fly Self-speculative decoding	LLM	2024
	Seesaw [121]	Dynamic Model Resharding	LLM	2025

deterministic approaches, integrating speculative sampling can significantly boost the performance of parallel decoding. Speculative sampling [61] speeds up sampling from autoregressive models while preserving accuracy by extending speculative execution to the stochastic setting. It enhances exact decoding from large models by running them in parallel with approximation models, generating multiple tokens concurrently without altering the underlying distribution. Similarly, [15] used speculative sampling to generate multiple tokens per transformer call, optimizing distributed model serving. To align compact draft models with target models, DistillSpec [184] applies knowledge distillation before speculative decoding, customizing the divergence function. More recently, [111] introduced Superposed Decoding, which generates k drafts with a single autoregressive inference pass and predicts k^2 drafts per step using n-gram interpolation to filter out incoherent outputs.

To address low predictive accuracy in draft models, especially with diverse text inputs and significant gaps between draft and target models, [78] proposed Online Speculative Decoding. This method updates draft models based on user query data, improving predictions by adapting to query distributions. To minimize additional parameters or training for effective draft models, [156] introduced Swift, a plug-and-play speculative decoding solution that uses layer-skipping, leveraging the compact draft model by skipping intermediate layers of the target LLM.

Token tree verification combines multiple candidate draft sequences into a token tree, sharing prefixes, and applies a tree attention mask for efficient verification. [88] introduced SpecInfer, a tree-based parallel decoding algorithm that uses small speculative models to predict LLM outputs and organizes predictions into a token tree. This approach reduces latency and computational costs while maintaining model quality. [8] proposed Medusa, an efficient method that enhances LLM inference by adding extra decoding heads to predict multiple tokens in parallel, reducing decoding steps through parallel processing. [69] introduced EAGLE, a speculative sampling framework that addresses feature-level autoregression uncertainty by incorporating a token sequence advanced by one time step, enabling efficient second-to-top-layer feature prediction. Subsequently, [68] improved this with EAGLE-2, introducing a context-aware dynamic draft tree for more accurate draft modeling, leveraging well-calibrated draft models with closely approximated confidence scores.

To explore speculative decoding for LVLMs, [39] enhanced inference efficiency in LVLMs, focusing on the LLaVA 7B model. [49] address token selection ambiguity, where visual autoregressive models assign uniformly low probabilities to tokens, impairing speculative decoding. They propose LANTERN, a relaxed acceptance condition utilizing token interchangeability in latent space, restoring speculative decoding effectiveness while maintaining image quality and semantic coherence through a total variation distance bound. [132] introduced Speculative Jacobi Decoding (SJD) for autoregressive text-to-image generation, enabling the model to predict and accept multiple tokens per step, generating images more efficiently than traditional methods.

5. DECODING APPLICATIONS

In this section, we shift our focus to organizing these methods according to the applications in which they have been used. Decoding algorithms play a crucial role across a variety of applications, from enhancing model alignment to optimizing specific generation tasks. Understanding how these algorithms are adapted to different applications provides valuable insights into their versatility and effectiveness. Figure 4 illustrates the diverse applications of decoding methods.

5.1 Improve Model Alignment

Decoding strategies play a vital role in improving model alignment by mitigating hallucinations, enhancing safety, and strengthening reasoning capabilities. These strategies involve tailoring the output generation process during inference to achieve the desired outcomes. Unlike other methods of improving model alignment, decoding strategies provide a dynamic and adaptive approach, ensuring that the model consistently meets user expectations while adhering to ethical standards.

5.1.1 Mitigate Hallucination

Decoding methods have emerged as an effective inference-time tool for mitigating hallucinations [46], which refers to the generation of plausible but factually incorrect content by the model. Compared to prompt-based [41; 140; 175] and knowledge-editing [176; 64] approaches, decoding methods are model-agnostic and offer better interpretability. [40] introduced Decoding by Contrasting Retrieval Heads (DeCoRe), a decoding strategy that mitigates hallucinations by dynamically contrasting the outputs of a base and masked LLM, guided by conditional entropy. Similarly, [162] proposed a Comparator-driven Decoding-Time (CDT) framework, which generates hallucinatory and truthful comparators using multi-task fine-tuning and refines next-token predictions by contrasting logit differences between the target LLMs and these comparators.

In addition to mitigating hallucinations in LLMs, recent studies have demonstrated the effectiveness of contrastive decoding in addressing object hallucinations in LVLMs. [58] explores visual contrastive decoding techniques, such as image downsampling and editing, to reduce hallucinations. [98] proposed ConVis, which reconstructs images from hallucinated captions using a text-to-image model and compares probability distributions to capture visual contrastive signals that penalize hallucination generation. To counter hallucinations caused by strong language model priors suppressing visual input, [139] introduced DeCo, a dynamic correction decoding method that selectively integrates knowledge into the final layer to adjust output logits. [48] developed CT^2S, which preserves only the least important vision tokens after early decoder layers, enhancing vision-text associations during autoregressive decoding. Inspired by the Information Bottleneck theory, [51] proposed CATCH, which integrates complementary visual decoupling for information separation, non-visual screening for hallucination detection, and adaptive token-level contrastive decoding for mitigation. More recently, [14] has discovered an "attention hijacking" phenomenon, where interference from instruction tokens distorts visual perception, diverting attention to less discriminative regions and leading to hallucinations. Additionally, [135] proposed an attention reallocation mechanism that redistributes excess attention from output tokens to vi-

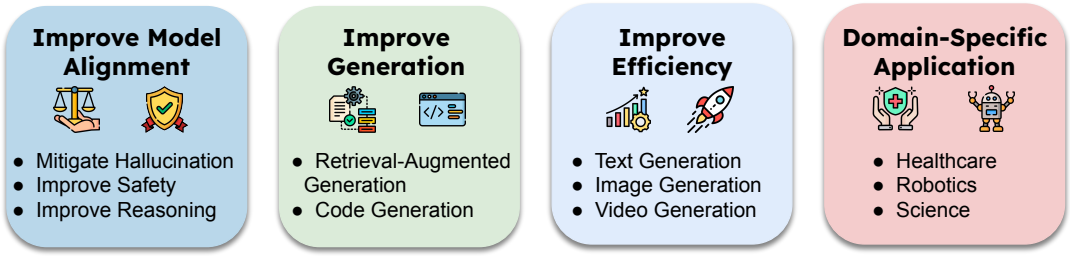

Figure 4: Decoding applications.

sual tokens, reducing LVLMs' reliance on language priors and strengthening visual input dependence to mitigate hallucinations.

5.1.2 *Improve Safety*

Besides mitigating hallucination, decoding methods are also used to enhance model safety. [2] introduced Safeinfer, a context-adaptive safety alignment strategy for generating safe responses using safety-guided decoding, which selects tokens based on the safety-optimized distributions to ensure ethical content. Similarly, [169] proposed the Root Defense Strategy, a decoder-oriented defense that corrects harmful queries directly, rather than rejecting them outright. Recognizing that LLMs may appear to block harmful queries but still harbor latent risks, [142] proposed Jailbreak Value Decoding (JVD). JVD evaluates the probability of generating harmful content in subsequent decoding steps from any given point.

Instead of manually selecting contrastive models or instruction templates, [179] proposed Adversarial Contrastive Decoding (ACD), an optimization-based framework that generates two opposing system prompts for prompt-based contrastive decoding. Using contrastive decoding, [94] successfully mitigates toxicity in a parameter-efficient manner. To improve alignment through decoding, [11] uses Q-learning to adjust the response distribution directly, maximizing a target reward without requiring model updates. Moreover, [3] identifies amplification bias and homogeneity issues in existing LLM decoding methods for recommendations. They propose Debiasing-Diversifying Decoding (D^3), which disables length normalization for ghost tokens to reduce amplification bias and uses a text-free assistant model to promote less frequent token generation, addressing recommendation homogeneity. To ensure the safety of conversational LLMs, [38] examines the effect of various decoding methods on the alignment between LLM-generated and human conversations. They show that fewer beams in beam search and lower P values in Nucleus sampling could lead to better alignment. PAD [143] adaptively injects calibrated Gaussian noise into token logits to mitigate privacy leakage of retrieved context in RAG.

5.1.3 *Improve Reasoning*

Reasoning [45] is a key aspect of human intelligence and an important trait for LLMs. Recent works have shown that reasoning chains are embedded in token selection, and decoding methods can enhance reasoning ability. [95] demonstrated that contrastive decoding improves reasoning in LLMs, outperforming existing methods by preventing abstract reasoning errors and avoiding simpler strategies like copying input sections during chain-of-thought reasoning. [101] proposed Distillation Contrastive Decoding (DCD), which enhances LLM reasoning at inference time by employing Contrastive Chain-of-Thought Prompting and advanced distillation techniques, including Dropout and Quantization, without requiring expert and amateur models. By investigating top-k alternative tokens, [149] found that CoT paths often emerge within these sequences and can be elicited from LLMs by modifying the decoding process.

Recently, to improve the performance of models that have been knowledge-edited for reasoning questions, [128] proposed Outdated Issue-Aware Decoding (DISCO), which captures the difference in the probability distribution between the original and edited models. To improve LLM reasoning in cross-domain settings, [112] proposed a method to teach multiple LLMs to collaborate by interleaving their generation at the token level. In parallel to syntactic decoding, such as auto-regressive decoding, [100] introduced semantic decoding, a perspective that views collaborative processes as optimization procedures within a semantic space, offering an abstraction for search and optimization directly in the space of semantic tokens.

In addition to improving reasoning performance, decoding methods can enhance inference efficiency for reasoning tasks. To reduce the cost of generating a full CoT reasoning chain, [77] introduced an auxiliary CoT model that generates and compresses the entire thought process into a compact token representation, semantically aligned with the original CoT output. To reduce the inference latency in tree-search-based reasoning methods, [151] introduced SEED, an efficient inference framework that accelerates reasoning tree construction using speculative scheduled execution, parallel drafting with speculative decoding, and a rounds-scheduled strategy to manage parallel drafts without verification conflicts. To reduce the cost of self-consistency decoding from the sampling process, [16] proposed self-para-consistency, where multiple paraphrases are generated for each test question.

5.2 Improve Generation Tasks

Decoding strategies enhance large generative models in tasks such as retrieval-augmented generation and code generation, improving both output quality and efficiency. These advancements focus on refining decoding methods, optimizing model architectures, and incorporating external resources to enhance performance.

5.2.1 *Retrieval Augmented Generation*

Decoding methods can effectively improve RAG at inference

time. [104] proposed entropy-based decoding to enhance truthfulness in retrieval-augmented LLMs, addressing challenges in ensuring faithful information retrieval. Similarly, adaptive contrastive decoding (ACD) [55] effectively incorporates contextual influence, improving the model's ability to generate contextually relevant outputs. To overcome the limitations of the generative retrieval model's fixed parametric capacity, [57] introduced Nonparametric Decoding (Np Decoding), which replaces standard embeddings with nonparametric contextualized vocab embeddings. Additionally, [168] proposed PAG, an optimization and decoding approach that guides the autoregressive generation of document identifiers in generative retrieval models through simultaneous decoding, streamlining the document retrieval process. More recently, to speed up language model generation with a retrieval-based approach, [42] proposed Retrieval-Based Speculative Decoding (REST). Unlike previous methods that rely on a draft language model, REST uses retrieval to generate draft tokens, leveraging the observation that text generation often follows common phrases and patterns.

5.2.2 *Code Generation*

Besides RAG, various decoding methods have been used for code generation with LLMs. [187] introduced Adaptive Temperature (AdapT) sampling, which adjusts the temperature during token decoding: higher for challenging tokens to explore diverse options, and lower for confident tokens to reduce tail randomness noise. [63] proposed Decoding Objectives for Code Execution (DOCE), a framework for execution-based evaluation. They focus on the effects of high-temperature sampling, execution-based reranking with high-quality unit tests, and self-debugging with multiple candidates. [102] introduced DocCGen, a two-step NL-to-code generation framework for structured domain-specific languages like YAML and JSON. It first detects relevant libraries using documentation to match the query, then constrains decoding with schema rules from these libraries.

To address security and correctness in code generation with Code LLMs, [37] investigates a defense approach using constrained decoding to generate secure code, proving more effective than prefix tuning without requiring a specialized training dataset. [146] presents Uncertainty-Aware Selective Contrastive Decoding (USCD), which improves one-pass code generation performance. It pre-judges noise in output distributions using standard deviation, then applies a "lame" prompt to reduce noise and enhance code quality. [92] proposed LEVER, which trains verifiers to assess program correctness using natural language input, the program, and its execution results. Programs are reranked by combining verification scores with LLM probabilities, marginalizing over identical execution results.

Additionally, decoding methods can help mitigate hallucination during code generation. [93] proposed DESEC, a two-stage method that uses token-level features to guide decoding. It builds an offline token scoring model with a proxy Code LLM and adjusts token likelihoods during decoding based on these scores. [134] introduced Selective Prompt Anchoring (SPA), which addresses self-attention dilution in LLMs for code generation. SPA strengthens the influence of selected parts of the initial prompt by adjusting the logit distribution based on the difference between anchored and non-anchored text.

5.3 Improve Generation Efficiency

Decoding methods are used to boost generation efficiency across multiple domains, such as text, image, and video.

5.3.1 *Text Generation*

Beyond speculative decoding, several works focus on enhancing text generation efficiency through innovative decoding strategies. Reward-Augmented Decoding (RAD) [26] used a lightweight unidirectional reward model to guide a language model in producing text with desired properties. [188] introduced Hierarchical Skip Decoding (HSD), an efficient autoregressive method that adaptively skips decoding layers based on sequence length, reducing computational overhead without requiring additional trainable components. [163] proposed Frustratingly Simple Decoding (FSD), which uses an anti-language model to penalize repetitive content. The anti-LM can be implemented as an n-gram model or a vectorized variant, adding no extra parameters and incurring minimal cost, making it as fast as greedy search.

More recently, [79] proposed ADED, a decoding methodology that enhances LLM efficiency without fine-tuning, utilizing an adaptive draft-verification process. [121] proposed Seesaw, which uses dynamic model resharding to enable reconfiguration of parallelization strategies during decoding. [32] introduced Position-Aware Depth Decay Decoding, which employs a power-law decay function to optimize the number of layers retained during token generation for more efficient performance.

5.3.2 *Image Generation*

To accelerate autoregressive text-to-image generation, [132] introduced Speculative Jacobi Decoding (SJD), a training-free probabilistic parallel decoding algorithm. [35] explores non-autoregressive text-to-image models that generate image tokens in parallel. They introduce an iterative mask-predict approach, enabling the model to refine its predictions using partially observed tokens, which enhances convergence speed and output quality. [130] introduced the Hybrid Autoregressive Transformer (HART), an autoregressive visual generation model that employs hybrid tokenization. This approach enables continuous feature decoding during generation, overcoming the limitations of finite VQ codebooks and improving overall generation quality.

5.3.3 *Video Generation*

By reformulating dense caption generation as a set of prediction tasks, [147] proposed PDVC, an end-to-end dense video captioning framework with parallel decoding. To ensure global coherence and local realism in video generation, GLOBER [126] uses a video decoder that processes global features and synthesizes video frames in a non-autoregressive manner. For enhanced flexibility, the video decoder incorporates normalized frame indexes to perceive temporal information, enabling the generation of arbitrary sub-video clips with predefined starting and ending frame indexes. Similarly, VideoGen [65] employs an advanced video decoder trained on unlabeled data to produce high-definition videos with strong temporal consistency and high frame fidelity.

5.4 Domain-Specific Applications

Decoding strategies can significantly enhance domain-specific

applications, such as healthcare and robotics. In the *healthcare* domain, [160] addresses hallucination in medical information extraction tasks with Alternate Contrastive Decoding (ALCD). They redefine the task as an identification-and-classification process, separating these steps by masking token optimization during fine-tuning. During inference, ALCD improves both identification and classification by contrasting output distributions from sub-task models, thereby minimizing interference from other LLM capabilities. Additionally, an adaptive constraint strategy refines the contrastive token scope, further enhancing performance. In the scientific domain, [180] implements cross-subject semantic decoding for video-stimulated fMRI, while [17] explores open-vocabulary auditory neural decoding using fMRI-prompted LLMs.

In *robotics*, [47] formulates the construction of action sequences as a probabilistic filtering problem, ensuring that the sequences are both probable according to the LM and feasible within grounded environmental models. Their approach demonstrates how grounded models can be derived from both simulation and real-world domains. By integrating knowledge from language models and grounded models, their decoding strategy effectively addresses complex, long-horizon embodiment tasks, enabling the generation of accurate and feasible action sequences. Similarly, [80] introduced Bidirectional Decoding (BID), an inference algorithm that combines action chunking with closed-loop operations. BID samples multiple predictions at each time step, optimizing for backward coherence (alignment with prior decisions) and forward contrast (high likelihood of future plans). This dual optimization approach ensures consistent decision-making while allowing the system to adapt to unexpected environmental changes.

6. DISCUSSIONS

Despite the significant potential of decoding methods across various tasks and applications, several challenges remain. In this section, we highlight the limitations of current decoding methods and discuss possible future research directions.

6.1 Exploring Dynamic and Universal Decoding Methods

Although decoding methods are effective for specific tasks, they often rely on manually crafted examples. As discussed earlier, token-wise and layer-wise contrastive decoding (§ 4.1) enhance control over text generation in LLMs and LVLMs. However, their effectiveness heavily depends on the selection of contrastive examples or layers. For instance, [66] compares outputs from smaller language models to those from larger ones, assuming that bigger models produce higher-quality text. However, this assumption does not always hold, as there are cases where the generation is worse with CD. Moreover, [110] even points out that smaller LLMs tend to be less sycophantic, meaning they are less likely to prioritize aligning with user beliefs over providing truthful responses. This highlights the complexity of crafting contrastive examples, as it requires balancing multiple qualities in the generated text. Similarly, in selecting contrasting layers, the optimal layers in DoLa [20] are sensitive across datasets, making it less versatile since it requires a task-specific validation set. This limitation presents an opportunity for future research to develop methods for constructing dynamic, universally applicable contrastive examples.

For guided decoding (§ 4.2), search methods like the one used in [174], rely on test cases and are constrained by a small search space. While [34] demonstrated that TS-LLMs perform well across reasoning, planning, alignment, and decision-making tasks on trees with a depth of 64, they still struggle to scale to larger scenarios due to the computational overhead introduced by node expansion and value evaluation. This highlights the need for more efficient strategies that can scale to larger problem spaces without introducing excessive computational costs.

6.2 Interpreting Decoding Methods

While some decoding methods, such as those explored by [149], offer insights into the intrinsic reasoning abilities of LLMs through the lens of decoding, more theoretical foundations are needed to understand why models behave in certain ways during the decoding process. For instance, [115] underscores the critical role of hyperparameter tuning in optimizing decoding methods. Their findings highlight that while some approaches can achieve impressive performance, they often require substantial effort to dial in the hyperparameters. In an initial effort, [13] theoretically demonstrates that contrastive decoding can be viewed as linearly extrapolating the next-token logits from a large, hypothetical language model. They find that this linear extrapolation may prevent contrastive decoding from producing the most obvious answers, as these are already assigned high probabilities by the base LM.

Moving forward, future research should focus on exploring the interpretability of these methods, which could provide valuable insights into how different decoding strategies align with human reasoning and decision-making processes. One potential approach is *mechanistic interpretability* [5; 71; 4], which seeks interpretability by reverse-engineering black-box models. For instance, [97] investigates the possibility of decoding multiple future tokens from a single token's hidden representation. Their causal intervention study reveals that certain layers can approximate the model's output with up to 48% accuracy from a single hidden state, providing significant insights into the model's prediction chain at each hidden state.

6.3 Combining Different Decoding Paradigms

As discussed in Section 4, contrastive decoding, guided decoding, and parallel decoding are versatile paradigms that have proven effective for a variety of tasks, such as controlling generation, improving quality, and enhancing efficiency. Given their individual successes, it is natural to explore the potential of combining these paradigms. For instance, [166] combined Speculative Decoding with contrastive decoding, achieving both improved generation quality and inference speedup. Additionally, these decoding paradigms can be integrated with other advanced generation techniques, such as RAG. [150] enhanced RAG by incorporating drafting with a set of specialized drafters, which provide diverse perspectives on the evidence while reducing the token count per draft. Future research could investigate further combinations of decoding paradigms to address the unique challenges posed by LLMs.

6.4 Expanding the Diversity of Decoding Objectives

Traditionally, decoding methods for LMs have primarily focused on text generation quality. However, recent works on LLMs and LVLMs have shifted toward improving the alignment of generated outputs. While several aspects of alignment have been explored, including toxicity [73], truthfulness [116; 20], and safety [161; 182], significant gaps remain in addressing other critical issues, such as *privacy, bias, and copyright concerns*. Additionally, while decoding methods have been applied across various domains, there are limited applications in high-stakes sectors like healthcare, law, and finance. Finally, for works focusing on improving the decoding efficiency through parallel decoding, there is a need to ensure the balance of decoding accuracy and efficiency. As noted by [158], there is still room for improvement to align the drafter with the target LLM for speculative decoding to scale up the drafter to improve decoding accuracy while maintaining its efficiency advantages. Future research could explore leveraging decoding methods to tackle these issues, offering a plug-and-play solution to enhance the trustworthiness of LLMs for important applications.

6.5 Improving Adversarial Robustness

Decoding methods are often overlooked from the security perspective. While much of the research focuses on enhancing the security of LLMs through techniques such as preventing data leakage and defending against jailbreaking attacks, the decoding mechanism itself is often neglected as a potential security risk. Decoding methods like SafeDecoding [161] and ROSE [182] are designed to generate safe responses from models, but they can also be exploited by attackers to generate malicious outputs. More concerningly, [91] demonstrated that adversaries with typical API access can steal the type and hyperparameters of a model's decoding algorithm at a low cost. This highlights a growing security concern regarding potential attacks that manipulate the underlying decoding method of a model. Moving forward, it would be valuable to explore strategies for defending against decoding-based attacks. As suggested by [91], watermarking [129] could serve as a potential countermeasure, adding noise to the final probability distribution and making it more difficult for attackers to extract hyperparameters from the target model.

7. CONCLUSION

This survey provides a comprehensive review of three primary decoding paradigms and their diverse applications in LLMs and LVLMs, showcasing their effectiveness and efficiency in tackling complex generation tasks. We believe that decoding methods offer a cost-effective way to enhance and extend LLM capabilities and hope our work sparks further discussion and research in this area.

Acknowledgments

This material is based upon work supported by NSF awards (SaTC-2241068, IIS-2506643, and POSE-2346158), a Cisco Research Award, and NSF NAIRR Pilot Award #240469. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the National Science Foundation. This work is supported in part by NSF under grants III-2106758.

8. REFERENCES

[1] L. A. Agrawal, A. Kanade, N. Goyal, S. Lahiri, and S. Rajamani. Monitor-guided decoding of code lms with static analysis of repository context. *Advances in Neural Information Processing Systems*, 36, 2024.

[2] S. Banerjee, S. Tripathy, S. Layek, S. Kumar, A. Mukherjee, and R. Hazra. Safeinfer: Context adaptive decoding time safety alignment for large language models. *arXiv preprint arXiv:2406.12274*, 2024.

[3] K. Bao, J. Zhang, Y. Zhang, X. Huo, C. Chen, and F. Feng. Decoding matters: Addressing amplification bias and homogeneity issue for llm-based recommendation. *arXiv preprint arXiv:2406.14900*, 2024.

[4] N. Belrose, Z. Furman, L. Smith, D. Halawi, I. Ostrovsky, L. McKinney, S. Biderman, and J. Steinhardt. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*, 2023.

[5] L. Bereska and E. Gavves. Mechanistic interpretability for ai safety–a review. *arXiv preprint arXiv:2404.14082*, 2024.

[6] L. Beurer-Kellner, M. Fischer, and M. Vechev. Guiding llms the right way: Fast, non-invasive constrained generation. *arXiv preprint arXiv:2403.06988*, 2024.

[7] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[8] T. Cai, Y. Li, Z. Geng, H. Peng, J. D. Lee, D. Chen, and T. Dao. Medusa: Simple llm inference acceleration framework with multiple decoding heads. *arXiv preprint arXiv:2401.10774*, 2024.

[9] Z. Cao, Y. Yang, and H. Zhao. Nothing in excess: Mitigating the exaggerated safety for llms via safety-conscious activation steering. *arXiv preprint arXiv:2408.11491*, 2024.

[10] S. Casper, L. Schulze, O. Patel, and D. Hadfield-Menell. Defending against unforeseen failure modes with latent adversarial training. *arXiv preprint arXiv:2403.05030*, 2024.

[11] S. Chakraborty, S. S. Ghosal, M. Yin, D. Manocha, M. Wang, A. S. Bedi, and F. Huang. Transfer q star: Principled decoding for llm alignment. *arXiv preprint arXiv:2405.20495*, 2024.

[12] A. Chan, A. Madani, B. Krause, and N. Naik. Deep extrapolation for attribute-enhanced generation. *Advances in Neural Information Processing Systems*, 34:14084–14096, 2021.

[13] H.-S. Chang, N. Peng, M. Bansal, A. Ramakrishna, and T. Chung. Explaining and improving contrastive decoding by extrapolating the probabilities of a huge and hypothetical lm. *arXiv preprint arXiv:2411.01610*, 2024.

[14] B. Chen, X. Lyu, L. Gao, J. Song, and H. T. Shen. Attention hijackers: Detect and disentangle attention hijacking in lvlms for hallucination mitigation. *arXiv preprint arXiv:2503.08216*, 2025.

[15] C. Chen, S. Borgeaud, G. Irving, J.-B. Lespiau, L. Sifre, and J. Jumper. Accelerating large language model decoding with speculative sampling. *arXiv preprint arXiv:2302.01318*, 2023.

[16] W. Chen, W. Wang, Z. Chu, K. Ren, Z. Zheng, and Z. Lu. Self-para-consistency: Improving reasoning tasks at low cost for large language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 14162–14167, 2024.

[17] X. Chen, C. Du, C. Liu, Y. Wang, and H. He. Open-vocabulary auditory neural decoding using fmri-prompted llm. *arXiv preprint arXiv:2405.07840*, 2024.

[18] J. Cheng, X. Liu, K. Zheng, P. Ke, H. Wang, Y. Dong, J. Tang, and M. Huang. Black-box prompt optimization: Aligning large language models without model training. *arXiv preprint arXiv:2311.04155*, 2023.

[19] Y. Cheng, X. Liang, Y. Gong, W. Xiao, S. Wang, Y. Zhang, W. Hou, K. Xu, W. Liu, W. Li, et al. Integrative decoding: Improve factuality via implicit self-consistency. *arXiv preprint arXiv:2410.01556*, 2024.

[20] Y.-S. Chuang, Y. Xie, H. Luo, Y. Kim, J. Glass, and P. He. Dola: Decoding by contrasting layers improves factuality in large language models. *arXiv preprint arXiv:2309.03883*, 2023.

[21] D. Dai, L. Dong, Y. Hao, Z. Sui, B. Chang, and F. Wei. Knowledge neurons in pretrained transformers. *arXiv preprint arXiv:2104.08696*, 2021.

[22] J. Dai, X. Pan, R. Sun, J. Ji, X. Xu, M. Liu, Y. Wang, and Y. Yang. Safe rlhf: Safe reinforcement learning from human feedback. *arXiv preprint arXiv:2310.12773*, 2023.

[23] S. Das, L. Jin, L. Song, H. Mi, B. Peng, and D. Yu. Entropy guided extrapolative decoding to improve factuality in large language models. *arXiv preprint arXiv:2404.09338*, 2024.

[24] S. Dathathri, A. Madotto, J. Lan, J. Hung, E. Frank, P. Molino, J. Yosinski, and R. Liu. Plug and play language models: A simple approach to controlled text generation. *arXiv preprint arXiv:1912.02164*, 2019.

[25] A. Deng, Z. Chen, and B. Hooi. Seeing is believing: Mitigating hallucination in large vision-language models via clip-guided decoding. *arXiv preprint arXiv:2402.15300*, 2024.

[26] H. Deng and C. Raffel. Reward-augmented decoding: Efficient controlled text generation with a unidirectional reward model. *arXiv preprint arXiv:2310.09520*, 2023.

[27] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[28] S. Dhuliawala, I. Kulikov, P. Yu, A. Celikyilmaz, J. Weston, S. Sukhbaatar, and J. Lanchantin. Adaptive decoding via latent preference optimization. *arXiv preprint arXiv:2411.09661*, 2024.

[29] C. Dong, Y. Li, H. Gong, M. Chen, J. Li, Y. Shen, and M. Yang. A survey of natural language generation. *ACM Computing Surveys*, 55(8):1–38, 2022.

[30] L. Dong, N. Yang, W. Wang, F. Wei, X. Liu, Y. Wang, J. Gao, M. Zhou, and H.-W. Hon. Unified language model pre-training for natural language understanding and generation. *Advances in neural information processing systems*, 32, 2019.

[31] A. Fan, M. Lewis, and Y. Dauphin. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833*, 2018.

[32] S. Fan, X. Fang, X. Xing, P. Han, S. Shang, and Y. Wang. Position-aware depth decay decoding: Boosting large language model inference efficiency. *arXiv preprint arXiv:2503.08524*, 2025.

[33] Y. Fang, Z. Yang, Z. Chen, Z. Zhao, and J. Zhou. From uncertainty to trust: Enhancing reliability in vision-language models with uncertainty-guided dropout decoding. *arXiv preprint arXiv:2412.06474*, 2024.

[34] X. Feng, Z. Wan, M. Wen, S. M. McAleer, Y. Wen, W. Zhang, and J. Wang. Alphazero-like tree-search can guide large language model decoding and training. *arXiv preprint arXiv:2309.17179*, 2023.

[35] Z. Feng, R. Hu, L. Liu, F. Zhang, D. Tang, Y. Dai, X. Feng, J. Li, B. Qin, and S. Shi. Emage: Non-autoregressive text-to-image generation. *arXiv preprint arXiv:2312.14988*, 2023.

[36] Y. Fu, P. Bailis, I. Stoica, and H. Zhang. Break the sequential dependency of llm inference using lookahead decoding. *arXiv preprint arXiv:2402.02057*, 2024.

[37] Y. Fu, E. Baker, Y. Ding, and Y. Chen. Constrained decoding for secure code generation. *arXiv preprint arXiv:2405.00218*, 2024.

[38] S. Furniturewala, K. Jaidka, and Y. Sharma. Impact of decoding methods on human alignment of conversational llms. *arXiv preprint arXiv:2407.19526*, 2024.

[39] M. Gagrani, R. Goel, W. Jeon, J. Park, M. Lee, and C. Lott. On speculative decoding for multimodal large language models. *arXiv preprint arXiv:2404.08856*, 2024.

[40] A. P. Gema, C. Jin, A. Abdulaal, T. Diethe, P. Teare, B. Alex, P. Minervini, and A. Saseendran. Decore: Decoding by contrasting retrieval heads to mitigate hallucinations. *arXiv preprint arXiv:2410.18860*, 2024.

[41] Z. Gou, Z. Shao, Y. Gong, Y. Shen, Y. Yang, N. Duan, and W. Chen. Critic: Large language models can self-correct with tool-interactive critiquing. *arXiv preprint arXiv:2305.11738*, 2023.

[42] Z. He, Z. Zhong, T. Cai, J. D. Lee, and D. He. Rest: Retrieval-based speculative decoding. *arXiv preprint arXiv:2311.08252*, 2023.

[43] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*, 2019.

[44] A. Holtzman, J. Buys, M. Forbes, A. Bosselut, D. Golub, and Y. Choi. Learning to write with cooperative discriminators. *arXiv preprint arXiv:1805.06087*, 2018.

[45] J. Huang and K. C.-C. Chang. Towards reasoning in large language models: A survey. *arXiv preprint arXiv:2212.10403*, 2022.

[46] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 2023.

[47] W. Huang, F. Xia, D. Shah, D. Driess, A. Zeng, Y. Lu, P. Florence, I. Mordatch, S. Levine, K. Hausman, et al. Grounded decoding: Guiding text generation with grounded models for embodied agents. *Advances in Neural Information Processing Systems*, 36, 2024.

[48] F. Huo, W. Xu, Z. Zhang, H. Wang, Z. Chen, and P. Zhao. Self-introspective decoding: Alleviating hallucinations for large vision-language models. *arXiv preprint arXiv:2408.02032*, 2024.

[49] D. Jang, S. Park, J. Y. Yang, Y. Jung, J. Yun, S. Kundu, S.-Y. Kim, and E. Yang. Lantern: Accelerating visual autoregressive models with relaxed speculative decoding. *arXiv preprint arXiv:2410.03355*, 2024.

[50] P. Kaliosis, J. Pavlopoulos, F. Charalampakos, G. Moschovis, and I. Androutsopoulos. A data-driven guided decoding mechanism for diagnostic captioning. *arXiv preprint arXiv:2406.14164*, 2024.

[51] Z. Kan, C. Zhang, Z. Liao, Y. Tian, W. Yang, J. Xiao, X. Li, D. Jiang, Y. Wang, and Q. Liao. Catch: Complementary adaptive token-level contrastive decoding to mitigate hallucinations in lvlms. *arXiv preprint arXiv:2411.12713*, 2024.

[52] M. Khalifa, H. Elsahar, and M. Dymetman. A distributional approach to controlled text generation. *arXiv preprint arXiv:2012.11635*, 2020.

[53] M. Kim, H. Lee, K. M. Yoo, J. Park, H. Lee, and K. Jung. Critic-guided decoding for controlled text generation. *arXiv preprint arXiv:2212.10938*, 2022.

[54] S. Kim, B. Cho, S. Bae, S. Ahn, and S.-Y. Yun. Vacode: Visual augmented contrastive decoding. *arXiv preprint arXiv:2408.05337*, 2024.

[55] Y. Kim, H. J. Kim, C. Park, C. Park, H. Cho, J. Kim, K. M. Yoo, S.-g. Lee, and T. Kim. Adaptive contrastive decoding in retrieval-augmented generation for handling noisy contexts. *arXiv preprint arXiv:2408.01084*, 2024.

[56] K. Konen, S. Jentzsch, D. Diallo, P. Schütt, O. Bensch, R. E. Baff, D. Opitz, and T. Hecking. Style vectors for steering generative large language model. *arXiv preprint arXiv:2402.01618*, 2024.

[57] H. Lee, J. Kim, H. Chang, H. Oh, S. Yang, V. Karpukhin, Y. Lu, and M. Seo. Nonparametric decoding for generative retrieval. *arXiv preprint arXiv:2210.02068*, 2022.

[58] Y.-L. Lee, Y.-H. Tsai, and W.-C. Chiu. Delve into visual contrastive decoding for hallucination mitigation of large vision-language models. *arXiv preprint arXiv:2412.06775*, 2024.

[59] S. Leng, H. Zhang, G. Chen, X. Li, S. Lu, C. Miao, and L. Bing. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13872–13882, 2024.

[60] B. Lester, R. Al-Rfou, and N. Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021.

[61] Y. Leviathan, M. Kalman, and Y. Matias. Fast inference from transformers via speculative decoding. In *International Conference on Machine Learning*, pages 19274–19286. PMLR, 2023.

[62] D. Li, J. Li, H. Li, J. C. Niebles, and S. C. Hoi. Align and prompt: Video-and-language pre-training with entity prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4953–4963, 2022.

[63] H.-S. Li, P. Fernandes, I. Gurevych, and A. F. Martins. Doce: Finding the sweet spot for execution-based code generation. *arXiv preprint arXiv:2408.13745*, 2024.

[64] K. Li, O. Patel, F. Viégas, H. Pfister, and M. Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36, 2024.

[65] X. Li, W. Chu, Y. Wu, W. Yuan, F. Liu, Q. Zhang, F. Li, H. Feng, E. Ding, and J. Wang. Videogen: A reference-guided latent diffusion approach for high definition text-to-video generation. *arXiv preprint arXiv:2309.00398*, 2023.

[66] X. L. Li, A. Holtzman, D. Fried, P. Liang, J. Eisner, T. Hashimoto, L. Zettlemoyer, and M. Lewis. Contrastive decoding: Open-ended text generation as optimization. *arXiv preprint arXiv:2210.15097*, 2022.

[67] X. L. Li and P. Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021.

[68] Y. Li, F. Wei, C. Zhang, and H. Zhang. Eagle-2: Faster inference of language models with dynamic draft trees. *arXiv preprint arXiv:2406.16858*, 2024.

[69] Y. Li, F. Wei, C. Zhang, and H. Zhang. Eagle: Speculative sampling requires rethinking feature uncertainty. *arXiv preprint arXiv:2401.15077*, 2024.

[70] X. Liang, H. Wang, Y. Wang, S. Song, J. Yang, S. Niu, J. Hu, D. Liu, S. Yao, F. Xiong, and Z. Li. Controllable text generation for large language models: A survey. *arXiv preprint arXiv:2408.12599*, 2024.

[71] Z. Lin, S. Basu, M. Beigi, V. Manjunatha, R. A. Rossi, Z. Wang, Y. Zhou, S. Balasubramanian, A. Zarei, K. Rezaei, et al. A survey on mechanistic interpretability for multi-modal foundation models. *arXiv preprint arXiv:2502.17516*, 2025.

[72] A. Liu, H. Bai, Z. Lu, X. Kong, S. Wang, J. Shan, M. Cao, and L. Wen. Direct large language model alignment through self-rewarding contrastive prompt distillation. *arXiv preprint arXiv:2402.11907*, 2024.

[73] A. Liu, M. Sap, X. Lu, S. Swayamdipta, C. Bhagavatula, N. A. Smith, and Y. Choi. Dexperts: Decoding-time controlled text generation with experts and anti-experts. *arXiv preprint arXiv:2105.03023*, 2021.

[74] J. Liu, A. Cohen, R. Pasunuru, Y. Choi, H. Hajishirzi, and A. Celikyilmaz. Making ppo even better: Value-guided monte-carlo tree search decoding. *arXiv preprint arXiv:2309.15028*, 2023.

[75] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35, 2023.

[76] S. Liu, H. Ye, L. Xing, and J. Zou. In-context vectors: Making in context learning more effective and controllable through latent space steering. *arXiv preprint arXiv:2311.06668*, 2023.

[77] T. Liu, Z. Chen, Z. Liu, M. Tian, and W. Luo. Expediting and elevating large language model reasoning via hidden chain-of-thought decoding. *arXiv preprint arXiv:2409.08561*, 2024.

[78] X. Liu, L. Hu, P. Bailis, A. Cheung, Z. Deng, I. Stoica, and H. Zhang. Online speculative decoding. *arXiv preprint arXiv:2310.07177*, 2023.

[79] X. Liu, B. Lei, R. Zhang, and D. Xu. Adaptive draft-verification for efficient large language model decoding. *arXiv preprint arXiv:2407.12021*, 2024.

[80] Y. Liu, J. I. Hamid, A. Xie, Y. Lee, M. Du, and C. Finn. Bidirectional decoding: Improving action chunking via closed-loop resampling. *arXiv preprint arXiv:2408.17355*, 2024.

[81] X. Lu, S. Welleck, P. West, L. Jiang, J. Kasai, D. Khashabi, R. L. Bras, L. Qin, Y. Yu, R. Zellers, et al. Neurologic a* esque decoding: Constrained text generation with lookahead heuristics. *arXiv preprint arXiv:2112.08726*, 2021.

[82] X. Lu, P. West, R. Zellers, R. L. Bras, C. Bhagavatula, and Y. Choi. Neurologic decoding:(un) supervised neural text generation with predicate logic constraints. *arXiv preprint arXiv:2010.12884*, 2020.

[83] Y. Lu, M. Bartolo, A. Moore, S. Riedel, and P. Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. *arXiv preprint arXiv:2104.08786*, 2021.

[84] W. Luo, F. Song, W. Li, G. Peng, S. Wei, and H. Wang. Odysseus navigates the sirens' song: Dynamic focus decoding for factual and diverse open-ended text generation. *arXiv preprint arXiv:2503.08057*, 2025.

[85] K. Meng, D. Bau, A. Andonian, and Y. Belinkov. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35:17359–17372, 2022.

[86] K. Meng, A. S. Sharma, A. Andonian, Y. Belinkov, and D. Bau. Mass-editing memory in a transformer. *arXiv preprint arXiv:2210.07229*, 2022.

[87] A. Meta. Introducing llama 3.1: Our most capable models to date. *Meta AI Blog*, 12, 2024.

[88] X. Miao, G. Oliaro, Z. Zhang, X. Cheng, Z. Wang, Z. Zhang, R. Y. Y. Wong, A. Zhu, L. Yang, X. Shi, et al. Specinfer: Accelerating generative large language model serving with tree-based speculative inference and verification. *arXiv preprint arXiv:2305.09781*, 2023.

[89] K. Min, M. Kim, K.-i. Lee, D. Lee, and K. Jung. Mitigating hallucinations in large vision-language models via summary-guided decoding. *arXiv preprint arXiv:2410.13321*, 2024.

[90] K. Murray and D. Chiang. Correcting length bias in neural machine translation. *arXiv preprint arXiv:1808.10006*, 2018.

[91] A. Naseh, K. Krishna, M. Iyyer, and A. Houmansadr. Stealing the decoding algorithms of language models. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 1835–1849, 2023.

[92] A. Ni, S. Iyer, D. Radev, V. Stoyanov, W.-t. Yih, S. Wang, and X. V. Lin. Lever: Learning to verify language-to-code generation with execution. In *International Conference on Machine Learning*, pages 26106–26128. PMLR, 2023.

[93] Y. Nie, C. Wang, K. Wang, G. Xu, G. Xu, and H. Wang. Decoding secret memorization in code llms through token-level characterization. *arXiv preprint arXiv:2410.08858*, 2024.

[94] T. Niu, C. Xiong, S. Yavuz, and Y. Zhou. Parameter-efficient detoxification with contrastive decoding. *arXiv preprint arXiv:2401.06947*, 2024.

[95] S. O'Brien and M. Lewis. Contrastive decoding improves reasoning in large language models. *arXiv preprint arXiv:2309.09117*, 2023.

[96] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[97] K. Pal, J. Sun, A. Yuan, B. C. Wallace, and D. Bau. Future lens: Anticipating subsequent tokens from a single hidden state. *arXiv preprint arXiv:2311.04897*, 2023.

[98] Y. Park, D. Lee, J. Choe, and B. Chang. Convis: Contrastive decoding with hallucination visualization for mitigating hallucinations in multimodal large language models. *arXiv preprint arXiv:2408.13906*, 2024.

[99] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. In M. Walker, H. Ji, and A. Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[100] M. Peyrard, M. Josifoski, and R. West. The era of semantic decoding. *arXiv preprint arXiv:2403.14562*, 2024.

[101] P. Phan, H. Tran, and L. Phan. Distillation contrastive decoding: Improving llms reasoning with contrastive decoding and distillation. *arXiv preprint arXiv:2402.14874*, 2024.

[102] S. Pimparkhede, M. Kammakomati, S. Tamilselvam, P. Kumar, A. P. Kumar, and P. Bhattacharyya. Doccgen: Document-based controlled code generation. *arXiv preprint arXiv:2406.11925*, 2024.

[103] C. Qian, J. Zhang, W. Yao, D. Liu, Z. Yin, Y. Qiao, Y. Liu, and J. Shao. Towards tracing trustworthiness dynamics: Revisiting pre-training period of large language models. *arXiv preprint arXiv:2402.19465*, 2024.

[104] Z. Qiu, Z. Ou, B. Wu, J. Li, A. Liu, and I. King. Entropy-based decoding for retrieval-augmented large language models. *arXiv preprint arXiv:2406.17519*, 2024.

[105] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. *Improving language understanding with unsupervised learning*, 2018.

[106] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[107] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

[108] A. Santilli, S. Severino, E. Postolache, V. Maiorca, M. Mancusi, R. Marin, and E. Rodolà. Accelerating transformer inference for translation via parallel decoding. *arXiv preprint arXiv:2305.10427*, 2023.

[109] M. Sclar, Y. Choi, Y. Tsvetkov, and A. Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. *arXiv preprint arXiv:2310.11324*, 2023.

[110] M. Sharma, M. Tong, T. Korbak, D. Duvenaud, A. Askell, S. R. Bowman, N. Cheng, E. Durmus, Z. Hatfield-Dodds, S. R. Johnston, et al. Towards understanding sycophancy in language models. *arXiv preprint arXiv:2310.13548*, 2023.

[111] E. Shen, A. Fan, S. M. Pratt, J. S. Park, M. Wallingford, S. M. Kakade, A. Holtzman, R. Krishna, A. Farhadi, and A. Kusupati. Superposed decoding: Multiple generations from a single autoregressive inference pass. *arXiv preprint arXiv:2405.18400*, 2024.

[112] S. Z. Shen, H. Lang, B. Wang, Y. Kim, and D. Sontag. Learning to decode collaboratively with multiple language models. *arXiv preprint arXiv:2403.03870*, 2024.

[113] T. Shen, R. Jin, Y. Huang, C. Liu, W. Dong, Z. Guo, X. Wu, Y. Liu, and D. Xiong. Large language model alignment: A survey. *arXiv preprint arXiv:2309.15025*, 2023.

[114] C. Shi, C. Yang, X. Zhu, J. Wang, T. Wu, S. Li, D. Cai, Y. Yang, and Y. Meng. Unchosen experts can contribute too: Unleashing moe models' power by self-contrast. *arXiv preprint arXiv:2405.14507*, 2024.

[115] C. Shi, H. Yang, D. Cai, Z. Zhang, Y. Wang, Y. Yang, and W. Lam. A thorough examination of decoding methods in the era of llms. *arXiv preprint arXiv:2402.06925*, 2024.

[116] W. Shi, X. Han, M. Lewis, Y. Tsvetkov, L. Zettlemoyer, and S. W.-t. Yih. Trusting your evidence: Hallucinate less with context-aware decoding. *arXiv preprint arXiv:2305.14739*, 2023.

[117] T. Shin, Y. Razeghi, R. L. Logan IV, E. Wallace, and S. Singh. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *arXiv preprint arXiv:2010.15980*, 2020.

[118] S. Smith, M. Patwary, B. Norick, P. LeGresley, S. Rajbhandari, J. Casper, Z. Liu, S. Prabhumoye, G. Zerveas, V. Korthikanti, et al. Using deepspeed and megatron to train megatron-turing nlg 530b, a large-scale generative language model. *arXiv preprint arXiv:2201.11990*, 2022.

[119] M. Stern, N. Shazeer, and J. Uszkoreit. Blockwise parallel decoding for deep autoregressive models. *Advances in Neural Information Processing Systems*, 31, 2018.

[120] N. Stiennon, L. Ouyang, J. Wu, D. Ziegler, R. Lowe, C. Voss, A. Radford, D. Amodei, and P. F. Christiano. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021, 2020.

[121] Q. Su, W. Zhao, X. Li, M. Andoorveedu, C. Jiang, Z. Zhu, K. Song, C. Giannoula, and G. Pekhimenko. Seesaw: High-throughput llm inference via model resharding. *arXiv preprint arXiv:2503.06433*, 2025.

[122] Y. Su and N. Collier. Contrastive search is what you need for neural text generation. *arXiv preprint arXiv:2210.14140*, 2022.

[123] Y. Su, T. Lan, Y. Wang, D. Yogatama, L. Kong, and N. Collier. A contrastive framework for neural text generation. *Advances in Neural Information Processing Systems*, 35:21548–21561, 2022.

[124] N. Subramani, N. Suresh, and M. E. Peters. Extracting latent steering vectors from pretrained language models. *arXiv preprint arXiv:2205.05124*, 2022.

[125] L. Sun, Y. Huang, H. Wang, S. Wu, Q. Zhang, C. Gao, Y. Huang, W. Lyu, Y. Zhang, X. Li, et al. Trustllm: Trustworthiness in large language models. *arXiv preprint arXiv:2401.05561*, 2024.

[126] M. Sun, W. Wang, Z. Qin, J. Sun, S. Chen, and J. Liu. Glober: coherent non-autoregressive video generation via global guided video decoder. *Advances in Neural Information Processing Systems*, 36, 2024.

[127] X. Sun, T. Ge, F. Wei, and H. Wang. Instantaneous grammatical error correction with shallow aggressive decoding. *arXiv preprint arXiv:2106.04970*, 2021.

[128] Z. Sun, Y. Liu, J. Wang, F. Meng, J. Xu, Y. Chen, and J. Zhou. Outdated issue aware decoding for factual knowledge editing. *arXiv preprint arXiv:2406.02882*, 2024.

[129] S. Szyller, B. G. Atli, S. Marchal, and N. Asokan. Dawn: Dynamic adversarial watermarking of neural networks. In *Proceedings of the 29th ACM international conference on multimedia*, pages 4417–4425, 2021.

[130] H. Tang, Y. Wu, S. Yang, E. Xie, J. Chen, J. Chen, Z. Zhang, H. Cai, Y. Lu, and S. Han. Hart: Efficient visual generation with hybrid autoregressive transformer. *arXiv preprint arXiv:2410.10812*, 2024.

[131] G. Team, R. Anil, S. Borgeaud, Y. Wu, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[132] Y. Teng, H. Shi, X. Liu, X. Ning, G. Dai, Y. Wang, Z. Li, and X. Liu. Accelerating auto-regressive text-to-image generation with training-free speculative jacobi decoding. *arXiv preprint arXiv:2410.01699*, 2024.

[133] I. Tenney. Bert rediscovers the classical nlp pipeline. *arXiv preprint arXiv:1905.05950*, 2019.

[134] Y. Tian and T. Zhang. Selective prompt anchoring for code generation. *arXiv preprint arXiv:2408.09121*, 2024.

[135] C. Tu, P. Ye, D. Zhou, L. Bai, G. Yu, T. Chen, and W. Ouyang. Attention reallocation: Towards zero-cost and controllable hallucination mitigation of mllms. *arXiv preprint arXiv:2503.08342*, 2025.

[136] A. M. Turner, L. Thiergart, G. Leech, D. Udell, J. J. Vazquez, U. Mini, and M. MacDiarmid. Activation addition: Steering language models without optimization. *arXiv e-prints*, pages arXiv–2308, 2023.

[137] B. Upadhyay, A. Sudhakar, and A. Maheswaran. Efficient reinforcement learning for unsupervised controlled text generation. *arXiv preprint arXiv:2204.07696*, 2022.

[138] A. K. Vijayakumar, M. Cogswell, R. R. Selvaraju, Q. Sun, S. Lee, D. Crandall, and D. Batra. Diverse beam search: Decoding diverse solutions from neural sequence models. *arXiv preprint arXiv:1610.02424*, 2016.

[139] C. Wang, X. Chen, N. Zhang, B. Tian, H. Xu, S. Deng, and H. Chen. Mllm can see? dynamic correction decoding for hallucination mitigation. *arXiv preprint arXiv:2410.11779*, 2024.

[140] H. Wang and K. Shu. Explainable claim verification via knowledge-grounded reasoning with large language models. *arXiv preprint arXiv:2310.05253*, 2023.

[141] H. Wang and K. Shu. Trojan activation attack: Red-teaming large language models using steering vectors for safety-alignment. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 2347–2357, 2024.

[142] H. Wang, B. Wu, Y. Bian, Y. Chang, X. Wang, and P. Zhao. Probing the safety response boundary of large language models via unsafe decoding path generation. *arXiv preprint arXiv:2408.10668*, 2024.

[143] H. Wang, X. Xu, B. Huang, and K. Shu. Privacy-aware decoding: Mitigating privacy leakage of large language models in retrieval-augmented generation. *arXiv preprint arXiv:2508.03098*, 2025.

[144] J. Wang, Y. Gao, and J. Sang. Valid: Mitigating the hallucination of large vision language models by visual layer fusion contrastive decoding. *arXiv preprint arXiv:2411.15839*, 2024.

[145] P. Wang, D. Zhang, L. Li, C. Tan, X. Wang, K. Ren, B. Jiang, and X. Qiu. Inferaligner: Inference-time alignment for harmlessness through cross-model guidance. *arXiv preprint arXiv:2401.11206*, 2024.

[146] S. Wang, L. Ding, L. Shen, Y. Luo, Z. He, W. Yu, and D. Tao. Uscd: Improving code generation of llms by uncertainty-aware selective contrastive decoding. *arXiv preprint arXiv:2409.05923*, 2024.

[147] T. Wang, R. Zhang, Z. Lu, F. Zheng, R. Cheng, and P. Luo. End-to-end dense video captioning with parallel decoding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6847–6857, 2021.

[148] X. Wang, J. Pan, L. Ding, and C. Biemann. Mitigating hallucinations in large vision-language models with instruction contrastive decoding. *arXiv preprint arXiv:2403.18715*, 2024.

[149] X. Wang and D. Zhou. Chain-of-thought reasoning without prompting. *arXiv preprint arXiv:2402.10200*, 2024.

[150] Z. Wang, Z. Wang, L. Le, H. S. Zheng, S. Mishra, V. Perot, Y. Zhang, A. Mattapalli, A. Taly, J. Shang, et al. Speculative rag: Enhancing retrieval augmented generation through drafting. *arXiv preprint arXiv:2407.08223*, 2024.

[151] Z. Wang, J. Wu, Y. Lai, C. Zhang, and D. Zhou. Seed: Accelerating reasoning tree construction via scheduled speculative decoding. *arXiv preprint arXiv:2406.18200*, 2024.

[152] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

[153] S. Welleck, J. Liu, X. Lu, H. Hajishirzi, and Y. Choi. Naturalprover: Grounded mathematical proof generation with language models. *Advances in Neural Information Processing Systems*, 35:4913–4927, 2022.

[154] G. Wiher, C. Meister, and R. Cotterell. On decoding strategies for neural text generators. *Transactions of the Association for Computational Linguistics*, 10:997–1012, 2022.

[155] H. Xia, T. Ge, P. Wang, S.-Q. Chen, F. Wei, and Z. Sui. Speculative decoding: Exploiting speculative execution for accelerating seq2seq generation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 3909–3925, 2023.

[156] H. Xia, Y. Li, J. Zhang, C. Du, and W. Li. Swift: On-the-fly self-speculative decoding for llm inference acceleration. *arXiv preprint arXiv:2410.06916*, 2024.

[157] H. Xia, Z. Yang, Q. Dong, P. Wang, Y. Li, T. Ge, T. Liu, W. Li, and Z. Sui. Unlocking efficiency in large language model inference: A comprehensive survey of speculative decoding. *arXiv preprint arXiv:2401.07851*, 2024.

[158] H. Xia, Z. Yang, Q. Dong, P. Wang, Y. Li, T. Ge, T. Liu, W. Li, and Z. Sui. Unlocking efficiency in large language model inference: A comprehensive survey of speculative decoding. In L.-W. Ku, A. Martins, and V. Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 7655–7671, Bangkok, Thailand, Aug. 2024. Association for Computational Linguistics.

[159] Y. Xie, K. Kawaguchi, Y. Zhao, X. Zhao, M.-Y. Kan, J. He, and Q. Xie. Self-evaluation guided beam search for reasoning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[160] D. Xu, Z. Zhang, Z. Zhu, Z. Lin, Q. Liu, X. Wu, T. Xu, X. Zhao, Y. Zheng, and E. Chen. Mitigating hallucinations of large language models in medical information extraction via contrastive decoding. *arXiv preprint arXiv:2410.15702*, 2024.

[161] Z. Xu, F. Jiang, L. Niu, J. Jia, B. Y. Lin, and R. Poovendran. Safedecoding: Defending against jailbreak attacks via safety-aware decoding. *arXiv preprint arXiv:2402.08983*, 2024.

[162] D. Yang, D. Xiao, J. Wei, M. Li, Z. Chen, K. Li, and L. Zhang. Improving factuality in large language models via decoding-time hallucinatory and truthful comparators. *arXiv preprint arXiv:2408.12325*, 2024.

[163] H. Yang, D. Cai, H. Li, W. Bi, W. Lam, and S. Shi. A frustratingly simple decoding method for neural text generation. *arXiv preprint arXiv:2305.12675*, 2023.

[164] K. Yang and D. Klein. Fudge: Controlled text generation with future discriminators. *arXiv preprint arXiv:2104.05218*, 2021.

[165] Y. Yang, L. Huang, and M. Ma. Breaking the beam search curse: A study of (re-) scoring methods and stopping criteria for neural machine translation. *arXiv preprint arXiv:1808.09582*, 2018.

[166] H. Yuan, K. Lu, F. Huang, Z. Yuan, and C. Zhou. Speculative contrastive decoding. *arXiv preprint arXiv:2311.08981*, 2023.

[167] Y. Zeldes, D. Padnos, O. Sharir, and B. Peleg. Technical report: Auxiliary tuning and its application to conditional text generation. *arXiv preprint arXiv:2006.16823*, 2020.

[168] H. Zeng, C. Luo, and H. Zamani. Planning ahead in generative retrieval: Guiding autoregressive generation through simultaneous decoding. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 469–480, 2024.

[169] X. Zeng, Y. Shang, Y. Zhu, J. Chen, and Y. Tian. Root defence strategies: Ensuring safety of llm at the decoding level. *arXiv preprint arXiv:2410.06809*, 2024.

[170] Y. Zeng, G. Liu, W. Ma, N. Yang, H. Zhang, and J. Wang. Token-level direct preference optimization. *arXiv preprint arXiv:2404.11999*, 2024.

[171] H. Zhang, S. Si, H. Wu, and D. Song. Controllable text generation with residual memory transformer. *arXiv preprint arXiv:2309.16231*, 2023.

[172] H. Zhang and D. Song. Discup: Discriminator cooperative unlikelihood prompt-tuning for controllable text generation. *arXiv preprint arXiv:2210.09551*, 2022.

[173] J. Zhang, J. Wang, H. Li, L. Shou, K. Chen, G. Chen, and S. Mehrotra. Draft & verify: Lossless large language model acceleration via self-speculative decoding. *arXiv preprint arXiv:2309.08168*, 2023.

[174] S. Zhang, Z. Chen, Y. Shen, M. Ding, J. B. Tenenbaum, and C. Gan. Planning with large language models for code generation. *arXiv preprint arXiv:2303.05510*, 2023.

[175] S. Zhang, L. Pan, J. Zhao, and W. Y. Wang. The knowledge alignment problem: Bridging human and external knowledge for large language models. *arXiv preprint arXiv:2305.13669*, 2023.

[176] S. Zhang, T. Yu, and Y. Feng. Truthx: Alleviating hallucinations by editing large language models in truthful space. *arXiv preprint arXiv:2402.17811*, 2024.

[177] X. Zhang and X. Wan. Mil-decoding: Detoxifying language models at token-level via multiple instance learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 190–202, 2023.

[178] Z. Zhao, E. Monti, J. Lehmann, and H. Assem. Enhancing contextual understanding in large language models through contrastive decoding. *arXiv preprint arXiv:2405.02750*, 2024.

[179] Z. Zhao, X. Zhang, K. Xu, X. Hu, R. Zhang, Z. Du, Q. Guo, and Y. Chen. Adversarial contrastive decoding: Boosting safety alignment of large language models via opposite prompt optimization. *arXiv preprint arXiv:2406.16743*, 2024.

[180] R. Zheng and L. Sun. Llm4brain: Training a large language model for brain video understanding. *arXiv preprint arXiv:2409.17987*, 2024.

[181] X. Zheng, H. Lin, X. Han, and L. Sun. Toward unified controllable text generation via regular expression instruction. *arXiv preprint arXiv:2309.10447*, 2023.

[182] Q. Zhong, L. Ding, J. Liu, B. Du, and D. Tao. Rose doesn't do that: Boosting the safety of instruction-tuned large language models with reverse prompt contrastive decoding. *arXiv preprint arXiv:2402.11889*, 2024.

[183] W. Zhou, Y. E. Jiang, E. Wilcox, R. Cotterell, and M. Sachan. Controlled text generation with natural language instructions. In *International Conference on Machine Learning*, pages 42602–42613. PMLR, 2023.

[184] Y. Zhou, K. Lyu, A. S. Rawat, A. K. Menon, A. Rostamizadeh, S. Kumar, J.-F. Kagy, and R. Agarwal. Distillspec: Improving speculative decoding via knowledge distillation. *arXiv preprint arXiv:2310.08461*, 2023.

[185] L. Zhu, D. Ji, T. Chen, P. Xu, J. Ye, and J. Liu. Ibd: Alleviating hallucinations in large vision-language models via image-biased decoding. *arXiv preprint arXiv:2402.18476*, 2024.

[186] T. Zhu, K. Zhang, J. Xie, and Y. Su. Deductive beam search: Decoding deducible rationale for chain-of-thought reasoning. *arXiv preprint arXiv:2401.17686*, 2024.

[187] Y. Zhu, J. A. Li, G. Li, Y. Zhao, J. Li, Z. Jin, and H. Mei. Improving code generation by dynamic temperature sampling. *arXiv preprint arXiv:2309.02772*, 2023.

[188] Y. Zhu, X. Yang, Y. Wu, and W. Zhang. Hierarchical skip decoding for efficient autoregressive text generation. *arXiv preprint arXiv:2403.14919*, 2024.

Context-Aware Counterfactual Data Augmentation for Gender Bias Mitigation in Language Models

Shweta Parihar
University of Illinois at Chicago
spari@uic.edu

Guangliang Liu
Michigan State University
liuguan5@msu.edu

Natalie Parde
University of Illinois at Chicago
parde@uic.edu

Lu Cheng
University of Illinois at Chicago
lucheng@uic.edu

ABSTRACT

A challenge in mitigating social bias in fine-tuned language models (LMs) is the potential reduction in language modeling capability, which can harm downstream performance. Counterfactual data augmentation (CDA), a widely used method for fine-tuning, highlights this issue by generating synthetic data that may align poorly with real-world distributions or creating overly simplistic counterfactuals that ignore the social context of altered sensitive attributes (e.g., gender) in the pretraining corpus. To address these limitations, we propose a simple yet effective context-augmented CDA method, *Context-CDA*, which uses large LMs to enhance the diversity and contextual relevance of the debiasing corpus. By minimizing discrepancies between the debiasing corpus and pretraining data through augmented context, this approach ensures better alignment, enhancing language modeling capability. We then employ uncertainty-based filtering to exclude generated counterfactuals considered low-quality by the target smaller LMs (i.e., LMs to be debiased), further improving the fine-tuning corpus quality. Experimental results on gender bias benchmarks demonstrate that *Context*-CDA effectively mitigates bias without sacrificing language modeling performance while offering insights into social biases by analyzing distribution shifts in next-token generation probabilities.

1. INTRODUCTION

Language models (LMs) have achieved remarkable success in generating human-like text across a wide range of applications, from chatbots [22] to translation [51]. However, these models often inherit and amplify biases present in their training data, which can lead to outputs that reinforce stereotypes or perpetuate harmful prejudices. These biases partly stem from the massive datasets used to train LMs, which frequently reflect societal imbalances, discriminatory language, and historical injustices.

Despite promising results in mitigating bias in LMs, a fundamental drawback of current debiasing methods is their potential to harm the core modeling abilities of LMs [8]. These methods often rely on strategies such as removing or altering biased data [49] or introducing controlled outputs [32], which can reduce the model's exposure to natural linguistic patterns. By filtering or distorting the underlying data, these techniques can hinder the model's ability to grasp the subtleties of language, leading to less fluent and contextually inaccurate text generation. This trade-off underscores the challenge of reducing bias while maintaining language proficiency.

Take Counterfactual Data Augmentation (CDA) [19; 52] for gender bias as an example. CDA works by altering gender attributes to generate counterfactual examples, which are then used during fine-tuning to reduce bias in model predictions. While CDA is effective in reducing bias, it often leads to a degradation in LMs' language modeling ability [8; 7; 29]. A major reason for this is the synthetic data generated by CDA may not align well with real-world data distributions and can create overly simplistic counterfactuals that fail to account for the social context of altered sensitive attributes (e.g., gender) embedded in the pre-training corpus.

To overcome these challenges, we propose a simple yet effective approach, *Context-CDA*, that leverages the generative capabilities of large LMs to augment the context used for debiasing. Particularly, larger LMs trained on vast and diverse corpora can generate context-rich counterfactual examples that are not only more diverse but also closely resemble natural language patterns (see an example in Figure 1). This context-aware data augmentation process helps avoid the severe distortions often introduced by traditional CDA techniques, where overly simplified examples fail to consider the social context of altered sensitive attributes. Our approach thus enables a more comprehensive debiasing process, enhancing fairness while also preserving the language modeling abilities of LMs.

However, the smaller LMs to be debiased often struggle to learn from text generated by larger LMs. Samples deemed challenging for the target LMs may hurt their language modeling capabilities [12]. To improve the performance of the model and reduce the impact of low-quality generations, we further propose using an uncertainty-based filtering strategy. Particularly, we leverage semantic entropy [15] to quantify the uncertainty of each augmented sample in our corpus. Semantic entropy advances other uncertainty estimation approaches (e.g., likelihoods) as it incorporates linguistic invariances created by shared meanings. High semantic entropy indicates that the target LM finds the sentence uncertain, ambiguous, or difficult to interpret due to multiple possible meanings or unpredictable word choices. We systematically filter out the counterfactual generations with the largest semantic entropy. Therefore, during fairness fine-tuning, the target LMs are trained on text that is clearer and less prone to misinterpretation. This filtering also ensures that the model can focus on learning meaningful and unambiguous patterns from the data, enhancing its ability to generalize and perform well in real-world applications.

Our contributions are as follows: **Method**: We develop *Context-CDA*, a new approach that produces context-rich, gender-flipped sentences. We enhance data quality for target LMs by using semantic entropy filtering to exclude ambiguous and irrelevant coun-

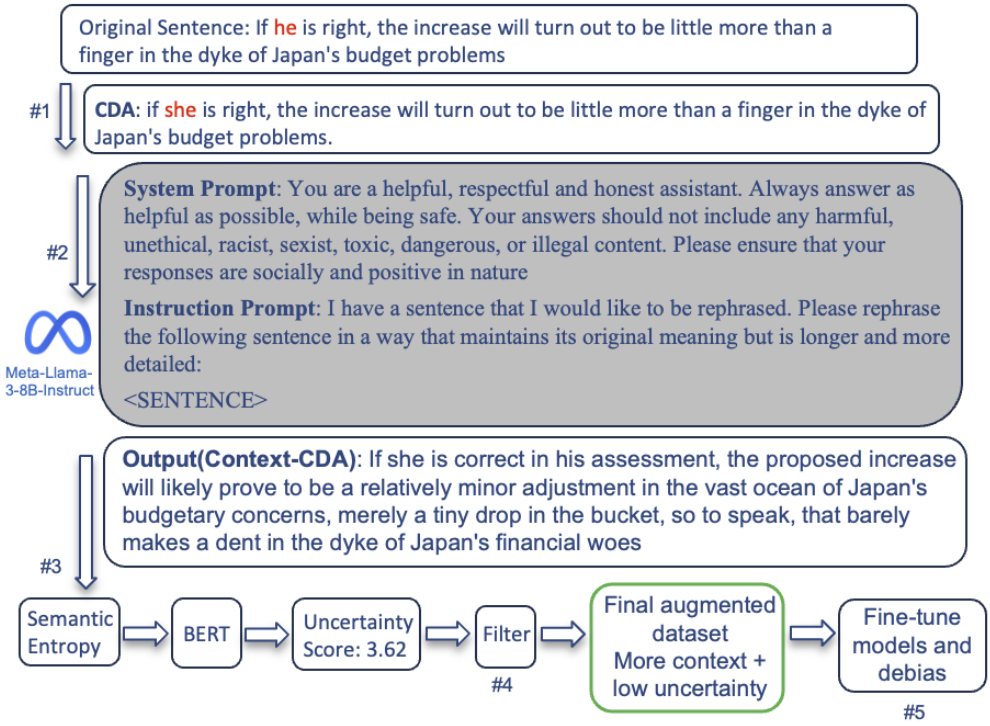

Figure 1: An illustration of the proposed *Context-CDA* pipeline using the StereoSet benchmark [23] with BERT [5]. Step 1: Flip the gender words. Step 2: Use a larger LM (e.g., Llama-3-8B-Instruct [9]) with the system and instruction prompt to get the augmented data. Step 3: Use the target small LM (e.g., BERT) to calculate the semantic entropy of augmented data. Step 4: Filter the counterfactuals based on the semantic entropy. Step 5: Debias the target small LM.

terfactuals for fine-tuning. **Experiments**: Our extensive evaluations show that *Context-CDA* preserves language modeling ability while reducing bias more effectively than traditional CDA methods (which may compromise on fluency). Semantic entropy-based filtering of low-quality counterfactuals further enhances debiasing effectiveness and maintains language fluency. Analysis of next-token distribution indicates a better gender balance and token diversity, leading to reduced skew towards male tokens, and increasing robustness. Comprehensive evaluation across five diverse model architectures, including BERT and DistilBERT (encoder-only models), T5 (encoder-decoder generative model), GPT-2, and Llama-3-1B (causal decoder-only generative model), demonstrates that these findings are consistent and robust across both discriminative and generative systems, validating the true model-agnostic nature of *Context-CDA*.

2. RELATED WORK

An LM may be deployed in a different setting than that for which it was intended, such as with or without a human intermediary for automated decision-making [38]. The primary causes of LM biases include inherent biases in the training data [1]. Bias mitigation techniques are categorized by the different stages of LM workflow. Pre-processing mitigation techniques focus on reducing bias and unfairness early in the dataset or model inputs. One of the early formalizations of this approach involves CDA, which has emerged as a prominent technique for mitigating biases in language models by introducing minimally perturbed examples that alter specific attributes while preserving the overall semantics. The foundational work in CDA by [49] demonstrated its effectiveness in reducing gender bias in coreference resolution by swapping gendered terms in training data. [19] extended this to a more general framework by applying CDA for neural NLP tasks by generating matched sentence pairs through gendered word interventions. [52] proposed a CDA method tailored for morphologically rich languages to generate grammatically correct gender-swapped sentences.

As described by [44], the CDA procedure can be applied in a one-sided manner (using only the counterfactual sentence for further training) or a two-sided manner (incorporating both the original and counterfactual sentences in the training data). [20] introduce Counterfactual Data Substitution (CDS), a variant of CDA, where gendered terms in the training data are probabilistically replaced with their counterfactual counterparts, rather than duplicating each instance with a gender-swapped version. More recently, [45] proposed Polyjuice, a controllable counterfactual generation system based on GPT-2, enabling diverse perturbation types for training and evaluation. [25] introduced a neural demographic perturber trained on a large human-annotated dataset (PANDA), and demonstrate that augmenting data via demographic perturbations improves model fairness with minimal performance trade-offs. Both works highlight the utility of automated, fine-grained counterfactual generation for robust and fair NLP.

Recent advances in CDA span diverse domains, reflecting its growing relevance in addressing data imbalance and spurious correlations. In NLP, [41] introduce FairFlow, a model-based CDA method

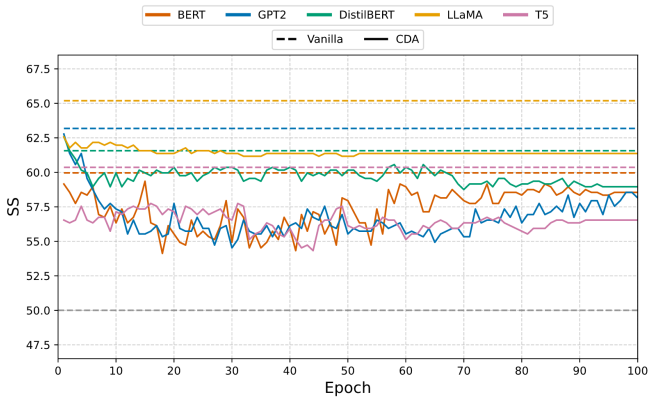

Figure 2: StereoSet bias score for 5 LLM Models - Vanilla v/s CDA (Intrinsic bias). 50 indicates no bias.

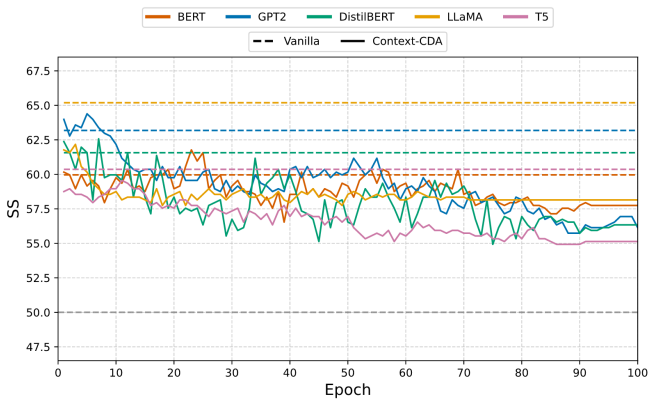

Figure 3: StereoSet bias score for 5 LLM Models - Vanilla v/s *Context-CDA*.

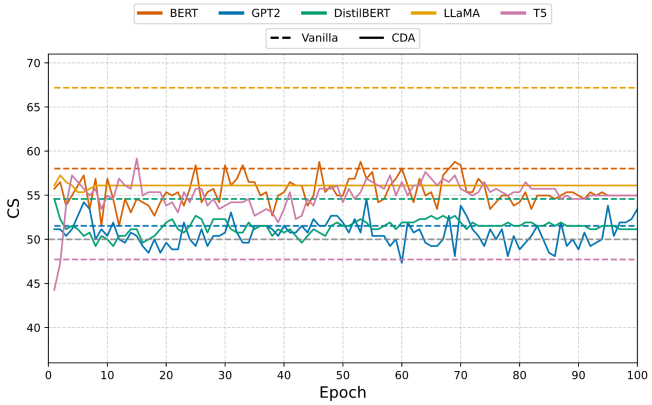

Figure 4: CrowS-Pairs bias score for 5 LLM Models - Vanilla v/s CDA (Intrinsic bias). 50 indicates no bias.

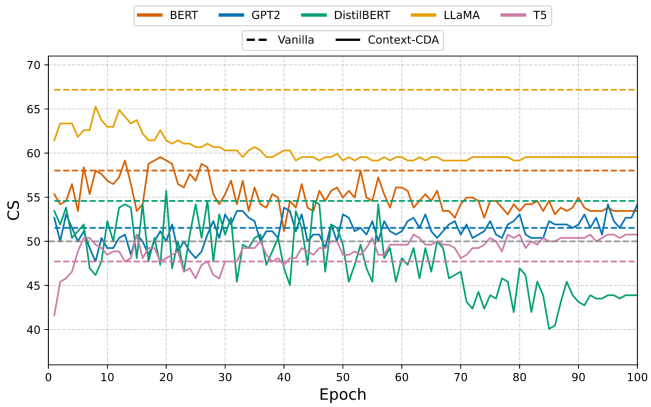

Figure 5: CrowS-Pairs bias score for 5 LLM Models - Vanilla v/s *Context-CDA*.

that generates parallel counterfactuals without manual intervention, improving fairness while preserving fluency. Similarly, [35] employ self-imitation reinforcement learning for CDA, emphasizing both fairness and robustness under contextual shifts. In sentiment analysis, [46] propose polarity-reversing augmentations using pre-trained transformers to enhance aspect-based sentiment classification. Beyond text, CDA has also seen traction in graph neural networks. For instance, CAGAD [47] utilizes diffusion models to generate counterfactual anomalies in graphs, improving anomaly detection in node representations. In reinforcement learning, ACAMDA [37] recovers temporal causal structures to simulate realistic hypothetical scenarios for improved data efficiency. CAIAC [42] swaps causally irrelevant state components to generate robust offline learning transitions. A parallel effort to improve interpretability can be seen in FCE-UTD [17], which generates factor-level counterfactuals for causal explanations in Point-of-Interest (POI) recommendation. Together, these works underscore the versatility of CDA across modalities and its emerging role in mitigating bias.

Additionally, CDA has been integrated with other debiasing methods such as adversarial training [28] and calibration strategies [40] to improve robustness and generalization. However, concerns remain around the semantic validity and distributional shift introduced by counterfactuals, leading to research on controllable generation (e.g., using LMs or paraphrasing systems) and evaluation metrics to ensure linguistic and contextual coherence. Our work builds upon this by extending the CDA framework to include context-rich, gender-flipped sentences generated by a large LM following the work of [10] for debiasing along with semantic entropy filtering to exclude ambiguous counterfactuals and improve corpus quality.

3. METHODS

The proposed method consists of three major steps: (1) Augmenting the context of CDA using large LMs, (2) Uncertainty-based filtering, and (3) Debiasing via fine-tuning on filtered counterfactuals. The overview of *Context-CDA* is illustrated in Figure 1.
CrowS-Pairs bias score for 5 LLM Models - Vanilla v/s CDA (Intrinsic bias). 50 indicates no bias.

3.1 Context-Aware CDA

For gender bias, vanilla CDA alters gender-related words, which can degrade language modeling ability while being effective at bias mitigation [8; 29]. One primary reason for this degradation is that the distribution of the corpus generated by vanilla CDA and then used for debiasing drifts from the distribution of the pre-trained corpus. When debiasing via fine-tuning the counterfactual generation, the n-gram distribution of the target LM is altered, resulting in the degradation of language modeling ability [7]. To address the limitation, we introduce *Context-CDA*, a prompting-based method that advances traditional CDA by prompting a larger LM to gener-

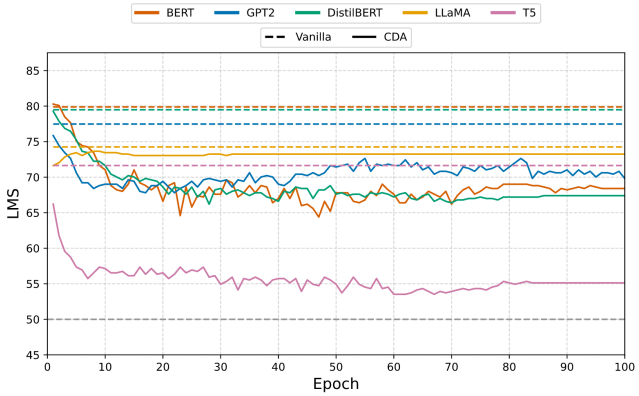

Figure 6: LMS score (↑) for 5 LLM Models - Vanilla v/s CDA.

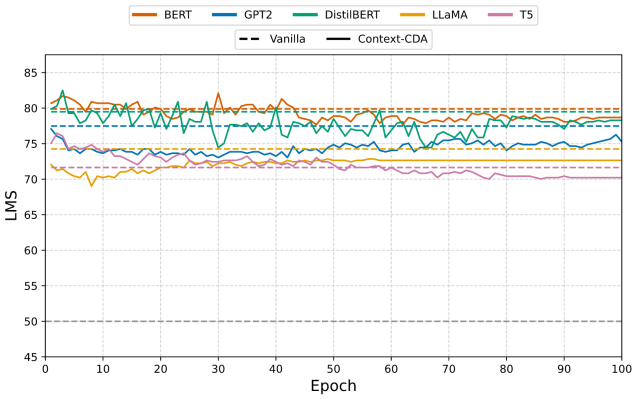

Figure 7: LMS score (↑) for 5 LLM Models - Vanilla v/s *Context-CDA*.

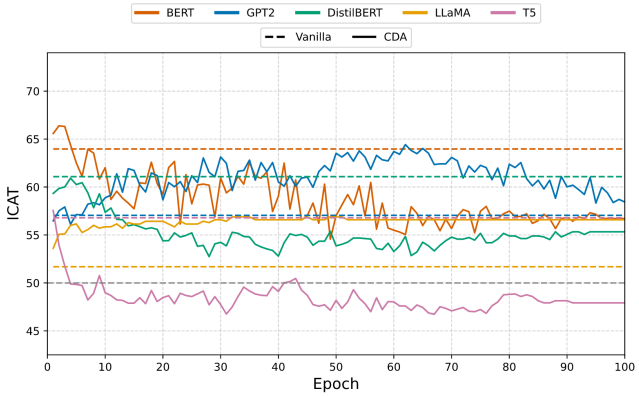

Figure 8: ICAT score (↑) for 5 LLM Models - Vanilla v/s CDA.

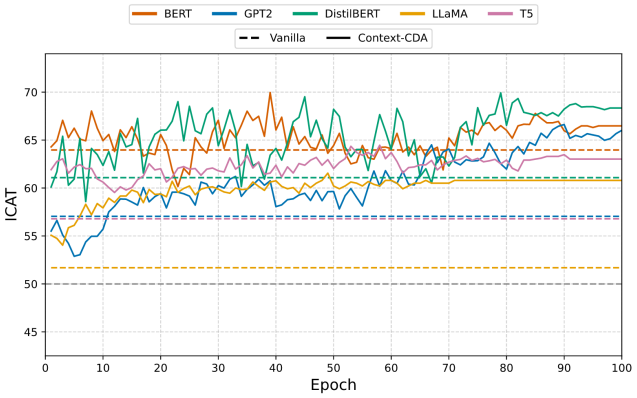

Figure 9: ICAT score (↑) for 5 LLM Models - Vanilla v/s *Context-CDA*.

ate contextually richer counterfactuals. Rather than merely flipping gender-related words, *Context-CDA* refines the sentences to preserve their original meaning while providing a more natural and diverse context aligned with the LM's pre-training distribution. This approach to context-aware data augmentation mitigates the severe distortions often caused by traditional CDA methods, where oversimplified examples fail to account for the social context of altered sensitive attributes. As a result, our method enables a more thorough debiasing process, promoting fairness while preserving the LMs' modeling capabilities. Specifically, after generating counterfactual examples using CDA $\tilde{x}_i = \text{CDA}(x_i)$, we design a system prompt and an instruction prompt (illustrated in Figure 1) to ask a Llama-3-8b-Instruct [9] model to rephrase $\tilde{x}_i$ such that it contains more context. The resulting generation is denoted as $\tilde{x}_i^c$.

3.2 Uncertainty-Based Filtering

With the generated counterfactuals, another challenge is that text generated by the larger LMs can be overly complex or noisy for the target smaller LMs to be debiased, hindering learning effectiveness. To further enhance the quality (e.g., mitigating hallucination) of the generated counterfactuals for the target LMs, we propose filtering out the generations that exhibit the greatest uncertainty by the target LMs. Uncertainty in the data has been shown to be an informative signal for algorithmic bias [39; 33] and hallucination in large LMs [6]. Specifically, we use a filtering process based on semantic entropy following the methodology in [15], which advances other uncertainty estimation approaches like likelihoods as it incorporates linguistic invariances created by shared meanings. For this process, we first sample multiple rephrased output sequences from a language model for each of our generated counterfactuals. These sequences are then clustered into semantic equivalence classes using a bidirectional entailment test following the methodology in [15], where two sequences are considered equivalent if they logically imply each other within context. Finally, the probabilities of sequences in each cluster are summed, and semantic entropy is calculated over these meaning-level probabilities to measure uncertainty over meanings rather than just surface forms. The semantic entropy (SE) for each counterfactual can be calculated as follows:

$$SE(\tilde{x}_i^c) \approx -|C|^{-1} \sum_{j=1}^{|C|} \log p(C_j \mid \tilde{x}_i^c), \tag{1}$$

where C denotes the number of samples generated by the larger LM. We calculate the semantic entropy for each generated counterfactual $\tilde{x}_i^c$ using the target LMs and filter out the top k-percent (e.g., 30%) of sentences with the highest entropy. This ensures that overly complex or noisy sentences, which the target LMs may struggle to learn, are removed from the corpus, leaving only the most useful examples for debiasing during the fine-tuning.

3.3 Debiasing via Fine-tuning on Filtered Context-CDA

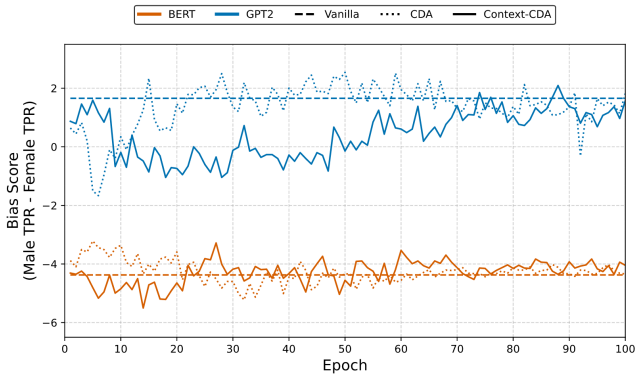

Figure 10: BiasBios score for BERT and GPT-2 (Extrinsic bias). 0 indicates no bias.

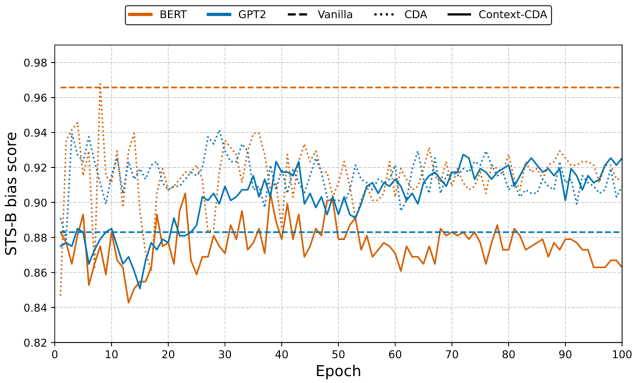

Figure 11: STS-B bias score for BERT and GPT-2 (Extrinsic bias). 0 indicates no bias.

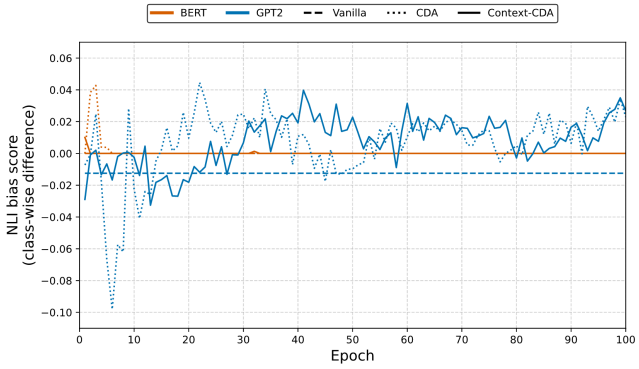

Figure 12: NLI-Bias score for BERT and GPT-2 (Extrinsic bias). 0 indicates no bias.

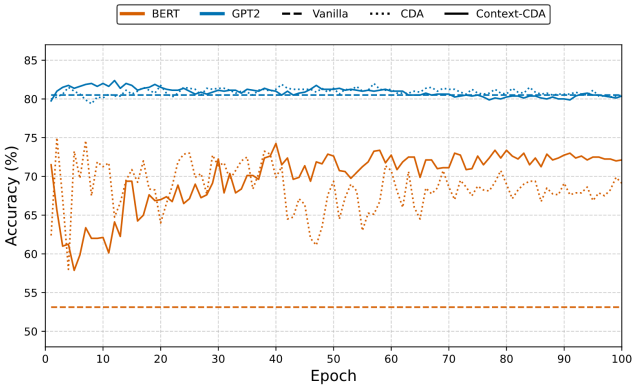

Figure 13: QNLI score (↑) for BERT and GPT-2 (Downstream task).

We use the contextually rich counterfactual data samples obtained after augmentation and filtering to debias the target small LMs [10] like BERT and GPT-2. The key idea is to introduce alternative versions of the input data where specific attributes (e.g., gender) are modified without changing the underlying meaning via augmented context. During fine-tuning, this augmentation forces the model to learn representations that are invariant to these modifications, which helps mitigate biased correlations in the data.

While our evaluation focuses on encoder models like BERT, this kind of representation debiasing is relevant to generative systems too. This is largely because BERT-like transformer encoders serve as foundational components in state-of-the-art encoder-decoder models (e.g., T5, BART, mBART) which are widely used for generation tasks such as machine translation and summarization. Moreover, contextualized encoder representations are commonly employed as frozen or fine-tuned backbones in modular and retrieval-augmented generation pipelines [16]. As a result, debiasing these shared representations can directly benefit a broad range of downstream generative applications.

Moreover, to demonstrate generalizability of *Context-CDA*, we evaluate across diverse architectures including encoder-only (BERT, DistilBERT), encoder-decoder (T5), and decoder-only (GPT-2, Llama-3.2-1B) models, as detailed in Section 4. In summary, our *Context-CDA* method aims to balance debiasing and language modeling performance by generating augmented sentences that maintain linguistic coherence while reducing bias via fine-tuning. This ensures that the target model learns from a corpus that is both diverse and aligned with natural language patterns.

4. EXPERIMENTS

We conduct the following experiments to validate the effectiveness of our proposed approach: (1) Evaluating the debiasing performance on intrinsic bias and language modeling capabilities, (2) Evaluating the debiasing performance on extrinsic bias, (3) Evaluating the debiasing performance on various downstream tasks, and (4) Analyzing next-token distribution to study predicted output token shifts after debiasing. All experiments are conducted across five diverse model architectures spanning encoder-only, encoder-decoder, and decoder-only designs to comprehensively validate *Context-CDA*'s robustness and generalizability.

4.1 Experimental Setup

Datasets: To implement gender bias mitigation, we use the news-commentary dataset [36] as our debiasing corpus following [14] with the gender words list from [50]. For **intrinsic bias**, we use gender samples from two benchmark datasets: (1) StereoSet [23] tests whether models can complete sentences without reinforcing harmful stereotypes, while still maintaining language fluency. (2) CrowS-Pairs [24] tests social biases in LMs by presenting sentence pairs, where one reflects a stereotype and the other does not. For **extrinsic bias**, we use datasets (1) BiasBios (Bias in Bios)

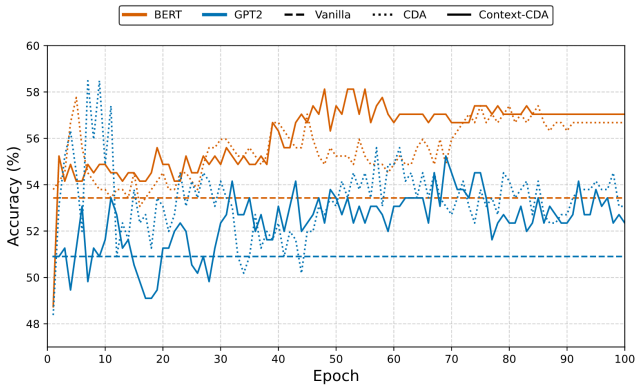

Figure 14: RTE score (↑) for BERT and GPT-2 (Downstream task).

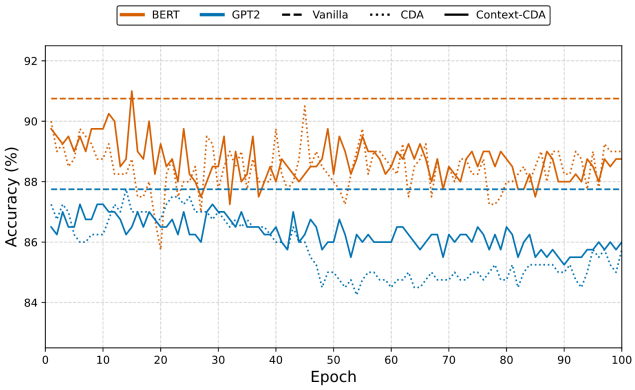

Figure 15: SST-2 score (↑) for BERT and GPT-2 (Downstream task).

[4], (2) STS-B (Semantic Textual Similarity Benchmark) [2] and (3) NLI-Bias (Natural Language Inference-Bias) [4]. For evaluating model performances on downstream tasks like sentiment analysis, entailment, and question-answering, we use the following datasets from the GLUE (General Language Understanding Evaluation) benchmark [43], (1) SST-2 (Stanford Sentiment Treebank-2) [34], (2) RTE (Recognizing Textual Entailment) [3], and (3) QNLI (Question-answering Natural Language Inference) [43] derived from the Stanford Question Answering Dataset (SQuAD).

Base LMs: Our evaluation covers five diverse LMs spanning different architectures. This includes two encoder-only models (BERT [5] and DistilBERT [30]), one encoder-decoder model (T5 [27]), and two causal decoder-only models (GPT-2 [26] and Llama-3.2-1B [13]) for a comprehensive evaluation. We evaluate BERT and GPT-2 on all intrinsic bias, extrinsic bias, and downstream task performance metrics. DistilBERT, T5, and Llama-3.2-1B are evaluated solely on intrinsic bias metrics. This selection enables us to validate that *Context-CDA* is model-agnostic and effective across encoder-only, encoder-decoder, and decoder-only architectures.

Baselines: We compare the Vanilla model, which is the pre-trained LM without debiasing, and LMs debiased using traditional CDA as the baseline for all models including BERT, GPT-2, DistilBERT, T5 and Llama-3.2-1B. Additionally, we consider several baselines for BERT and GPT-2, respectively. For BERT, we use the following methods as the baseline: (1) MABEL [11], an intermediate pre-training approach for mitigating gender bias in contextualized representation; (2) INLP [28], mitigating gender bias in word embeddings by removing information from neural representations; (3) SelfDebias [31], a decoding algorithm that, given only a textual description of the undesired behavior, reduces the probability of an LM producing problematic text; and (4) SENT-DEBIAS [18], which reduces gender bias in sentence-level representations. For GPT-2, we use the following methods as baselines: (1) SelfDebias [31], a decoding algorithm that, given only a textual description of the undesired behavior, reduces the probability of an LM producing problematic text; (2) SENT-DEBIAS [18], which reduces gender bias in sentence-level representations, and (3) wiki-debiased [48], a baseline with GPT-2 as the base LM that uses parameter-efficient methods to fine-tune GPT-2 using WikiText2 [21].

Metrics: To evaluate debiasing performance, we distinguish between intrinsic bias and extrinsic bias. Intrinsic bias refers to the stereotypical associations encoded directly within a language model's representations or probabilities, independent of downstream tasks. It is commonly measured through benchmarks such as StereoSet and CrowS-Pairs, which assess whether models favor stereotypical over anti-stereotypical continuations or sentence pairs. Extrinsic bias captures disparities that arise when models are applied to downstream tasks. It reflects whether representational biases affect task performance, for instance, differences in prediction accuracy or true positive rates across demographic groups. We measure extrinsic bias using datasets such as BiasBios, STS-B, and NLI-Bias.

For intrinsic bias, we report four standard metrics from the StereoSet [23] and CrowS-Pairs [24] benchmarks: (1) Stereotype Score (SS) calculates the percentage of examples in which a model prefers a stereotypical association over an anti-stereotypical association (a score of 50 indicates no bias and a score above and below 50 indicates preference for stereotypical or anti-stereotypical associations); (2) Language Modeling Score (LMS) calculates the percentage of instances in which an LM prefers meaningful over meaningless association (a higher score indicates better language performance); (3) Idealized CAT Score (ICAT) describes the comprehensive performance of the model by combining SS and LMS into one score (a higher score indicates better performance on debiasing while maintaining language fluency); and (4) CrowS-Pairs Score (CS) calculates the percentage of instances where the model assigns a higher probability to the stereotypical sentence over the anti-stereotypical one (a score closer to 50 indicates less bias, while scores above or below 50 suggest a preference toward biased associations).

For extrinsic bias, we report the following three metrics: (1) NLI-Bias measures the class-wise difference between male and female samples in the NLI-Bias dataset, with scores closer to 0 indicating less bias; (2) BiasBios measures the difference in true positive rate between male and female samples in BiasBios dataset; and (3) STS-B flips the gendered words in the sentence pairs and then calculates semantic similarity in original and gender-flipped sentence pairs to measure gender bias. The bias score is calculated as the difference in prediction accuracy between male and female groups, with scores closer to 0 indicating less bias. For performance on downstream tasks, we use accuracy as a metric for datasets (1) QNLI, (2) RTE, and (3) SST-2.

We set the uncertainty filtering threshold k to 30% as it gives the best performance based on our debiasing corpus as shown in the ablation study in Appendix A. For each language model, including BERT, GPT-2, DistilBERT, T5, and Llama-3.2-1B, we perform debiasing at each epoch to evaluate the intrinsic bias and lan-

	BERT				GPT-2			
Debiasing Technique	**SS**	**LMS (↑)**	**ICAT (↑)**	**CS**	**SS**	**LMS (↑)**	**ICAT (↑)**	**CS**
MABEL	47.28	51.65	48.84	<u>52.29</u>	-	-	-	-
INLP	<u>49.16</u>	50.25	49.41	55.73	-	-	-	-
wiki-debiased	-	-	-	-	60.40	<u>91.01</u>	72.08	56.49
SelfDebias	59.34	<u>84.20</u>	<u>68.47</u>	<u>52.29</u>	<u>56.05</u>	87.43	<u>73.18</u>	56.11
SENT-DEBIAS	59.37	84.09	68.33	<u>52.29</u>	60.84	89.07	69.76	56.11
Vanilla	59.95	79.87	63.96	58.01	63.17	77.46	57.04	51.52
CDA	58.55	68.41	56.71	54.96	57.34	69.61	59.39	<u>50.01</u>
Context-CDA	57.75	78.67	66.48	53.43	56.13	75.25	66.01	50.76

Table 1: Intrinsic bias evaluation for BERT and GPT-2 comparing baselines. Underlined values indicate the best performance.

guage modeling performance using the StereoSet and CrowS-Pairs benchmarks. Since our method fine-tunes models, we compare the vanilla, CDA, and *Context-CDA* models at each epoch of debiasing. For BERT and GPT-2, we additionally fine-tune models at each epoch to measure extrinsic bias and downstream task performance across datasets. This ensures a comprehensive assessment of debiasing effectiveness across both representation-level and task-level biases. This multi-faceted evaluation enables us to assess not only whether bias is reduced but also whether language modeling capability and downstream task performance are preserved.

4.2 Post Debiasing Performance on Intrinsic Bias Scores and Language Modeling

4.2.1 Multi-Model Evaluation: Demonstrating Robustness and Generalizability

To validate the robustness and model-agnostic nature of *Context-CDA*, we evaluate intrinsic bias on all five LMs.

Intrinsic Bias. To measure intrinsic bias, we conduct evaluations using the StereoSet (SS) and CrowS-Pairs (CS) benchmark at every debiasing epoch across all model architectures - BERT, DistilBERT, GPT-2, Llama-3.2-1B, and T5. We present detailed results comparing the performance of all models based on the SS score for CDA in Fig. 2 and *Context-CDA* in Fig. 3. We also present similar detailed results based on the CS score for CDA in Fig. 4 and *Context-CDA* in Fig. 5. Here, scores closer to 50 indicate less bias with a score of 50 indicating no bias. We observe that the vanilla BERT and DistilBERT generally start with the highest bias scores before debiasing begins. As debiasing progresses across epochs, both encoder models show consistent debiasing patterns and the SS and CS scores gradually approaches 50, indicating a reduction in bias in the models. We also observe that towards the end of training epochs, the bias scores stabilize and we conclude that further fine-tuning is not required. Towards epochs 75-85, *Context-CDA* starts outperforming CDA and achieves a stable bias score better than CDA. This consistency validates *Context-CDA* for encoder-only architectures. In GPT-2, CS bias score for *Context-CDA* and traditional CDA reaches around 50 (no bias) early on and oscillates about the same mean bias score as fine-tuning progresses. In Llama-3.2-1B, epochs 0 to 25 show a sustained decline in SS score reaching a plateau that indicates convergence without overfitting. The stable plateau throughout training demonstrates that further fine-tuning does not degrade or improve the debiasing performance. Both generative language models achieve comparable or superior results to their non-augmented baselines, supporting our claim that *Context-CDA* is effective for generative systems as well. T5, as an encoder-decoder model, shows particularly strong performance, achieving a balanced SS score and a significant improvement in CS score. Thus, compared to traditional CDA, *Context-CDA* shows consistent improvement across all intrinsic metrics.

Language Modeling Ability. To measure language modeling performance, we evaluate the Language Modeling Score (LMS) and Idealized CAT Score (ICAT) scores at every debiasing epoch across all model architectures - BERT, DistilBERT, GPT-2, Llama-3.2-1B, and T5. We present detailed results comparing the performance of all models based on the LMS score for CDA in Fig. 6 and *Context-CDA* in Fig. 7. We also present similar detailed results based on the ICAT score for CDA in Fig. 8 and *Context-CDA* in Fig. 9. Notably, across all models, *Context-CDA* consistently improves both LMS and ICAT scores compared to CDA. *Context-CDA* consistently outperforms CDA and achieves LMS scores as well as Vanilla models, while outperforming both Vanilla and CDA baselines in ICAT scores. This is a strong indicator of *Context-CDA's* superior language modeling performance compared to CDA, reinforcing that *Context-CDA* achieves better linguistic understanding and better preserves language modeling capability while reducing bias. We attribute this improvement to the greater diversity and naturalness of the *Context-CDA* corpus compared to traditional CDA, which enables models to retain stronger linguistic representations after debiasing. Overall, *Context-CDA* achieves effective bias mitigation while simultaneously improving language modeling performance.

4.2.2 Convergence Patterns

The per-epoch analysis reveals important training dynamics: (1) **Convergence Timing:** Across all five models, debiasing performance stabilizes at epochs 75-85, indicating the method reaches optimal performance earlier in training. (2) **Stability Without Overfitting:** The plateau in bias scores and sustained or improved scores demonstrate that models do not overfit to the debiasing corpus; instead, they learn stable, debiased representations. (3) **Model-Agnostic Pattern:** The identical convergence behavior across BERT, DistilBERT, GPT-2, Llama-3.2-1B, and T5 validates that this stability is not an artifact of a single architecture but a general property of *Context-CDA*. (4) **Training Efficiency:** The early stabilization suggests that practitioners can reduce training epochs, making the method computationally efficient. These insights strengthen confidence in the method's robustness and generalizability.

4.2.3 Comparison with Prior Debiasing Methods

We also compare the bias scores of fine-tuned BERT and GPT-2 against other debiasing baselines in Table 1 for an overall comparison. While methods such as SelfDebias and SENT-DEBIAS achieve higher scores on isolated metrics (e.g., ICAT), they often rely on inference-time interventions or post-hoc representation manipulation, which: (1) are model-specific or task-specific, limiting generalizability; (2) do not demonstrate consistent performance

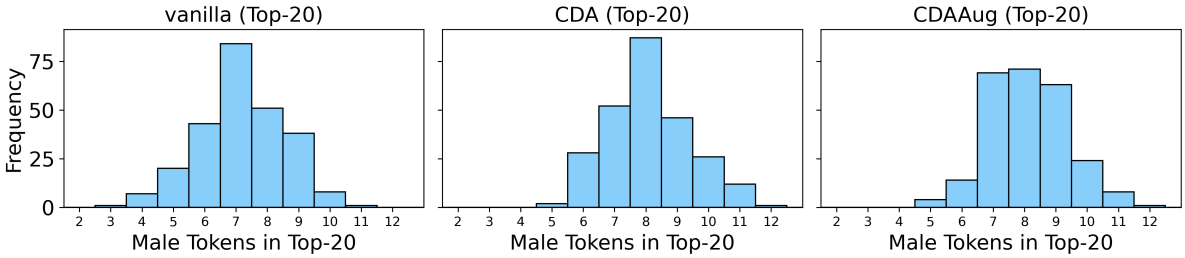

Figure 16: Top-20 male token distributions for GPT2.

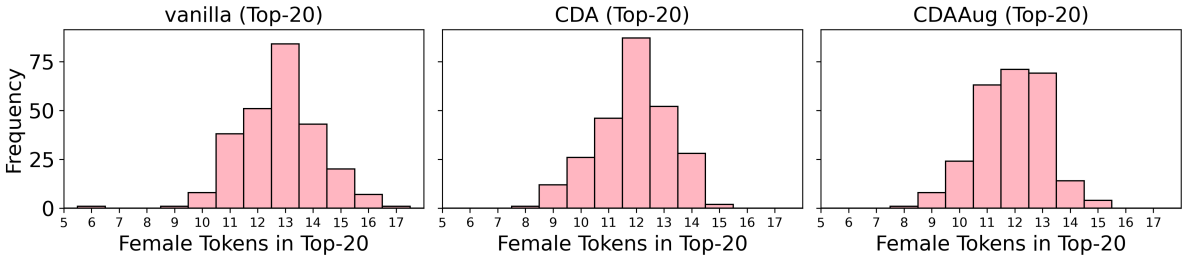

Figure 17: Top-20 female token distributions for GPT2.

across both intrinsic and extrinsic metrics; and (3) lack validation on diverse model architectures. In contrast, our extended evaluation demonstrates that *Context-CDA* achieves balanced, consistent performance across five distinct architectures, maintaining stable debiasing while preserving language modeling capability across both encoder-only and generative models. Wiki-debiased, in contrast, leverages curated Wikipedia data, making it domain-limited, whereas *Context-CDA* can be applied to arbitrary corpora. Consequently, *Context-CDA* is model-agnostic, requires no architecture modifications, and integrates seamlessly into the fine-tuning pipeline, making it a practical and effective debiasing solution across model architectures. The consistency of results across five model types provides strong empirical evidence of robustness, addressing concerns about limited generalization from single-model evaluation. It strikes a strong balance by offering the best trade-off between bias mitigation and language modeling performance, outperforming standard CDA across all intrinsic bias metrics and achieving comparable or superior results to other baselines, all without sacrificing fluency.

4.3 Post Debiasing Performance on Extrinsic Bias and Downstream Tasks

For evaluating performance on extrinsic bias metrics, we further fine-tune our models - BERT and GPT-2 on STS-B, NLI-Bias and BiasBios tasks and then evaluate the extrinsic bias scores on the fine-tuned models. Here, scores closer to 0 indicate low bias. For BERT, we find that for STS-B (Fig. 11), *Context-CDA* performs better than CDA, whereas for NLI-Bias and BiasBios (Fig. 12 and Fig. 10 respectively), *Context-CDA* performs as good as CDA indicating that *Context-CDA* is robust for even extrinsic bias evaluation metrics. For GPT-2, *Context-CDA* performs as well as CDA for BiasBios, STS-B, and NLI-Bias (Figs. 10, 11, and 12 respectively), further validating its effectiveness in reducing extrinsic bias across model architectures. For evaluating performance on various downstream language understanding tasks, we further fine-tune our debiased models on QNLI, RTE, and SST-2 and then evaluate the fine-tuned models for accuracy. Higher score indicates higher accuracy. For QNLI (Fig. 13), *Context-CDA* performs better than CDA whereas for RTE and SST-2 (Fig. 14 and Fig. 15), *Context-CDA* performs slightly better or as good as CDA, indicating that models fine-tuned with *Context-CDA* can perform better than CDA even in various downstream tasks.

4.4 Next-Token Distribution

To gain deeper insights into the debiasing performance, we compare the vanilla LM, CDA, and *Context-CDA* by examining next-token distributions in gender-related contexts. We focus on the GPT-2 model with the Stereotype Score (SS) closest to 50 after CDA debiasing, allowing us to investigate how the debiasing process affects token distribution and explore its impact on model predictions. We use sentences from StereoSet that specifically evaluate gender bias. Each sentence is split into two parts: the portion before the BLANK, referred to as the context, and the portion after it. The context is fed into GPT-2, and we compute the logits for the next token, extracting scores for tokens in a predefined male–female mapping set (approximately 200 words) [50]. After applying softmax, we obtain probabilities for each word and identify the top-20 tokens. We then count how many tokens are male-related and how many are female-related.

Figures 16 and 17 illustrate the frequency of male and female tokens, respectively, among the top-20 predictions. For male tokens, the vanilla LM peaks at 7 male-related tokens, while CDA and *Context-CDA* shift the peak to 8. Importantly, *Context-CDA* yields a more balanced distribution across contexts, reducing skew toward male-associated tokens. Similarly, for female tokens, the vanilla LM peaks at 13 female-related tokens, while CDA and *Context-CDA* shift the peak toward 12. Again, *Context-CDA* produces a more even distribution, avoiding over-concentration on specific female-

related tokens. Together, these results demonstrate that both CDA and *Context-CDA* adjust the token distribution away from the stronger bias seen in the Vanilla model. However, *Context-CDA* is more effective at spreading token probabilities evenly across male and female categories. This not only mitigates over-reliance on gender-specific terms but also promotes linguistic diversity, thereby improving robustness in predictions and reflecting a more balanced language modeling ability.

5. LIMITATIONS

While our method *Context-CDA* demonstrates improvements over traditional CDA in mitigating gender bias without compromising language modeling performance, several limitations remain. First, using large LMs for generating context-rich augmentations may introduce computational and environmental costs, which may hinder scalability and accessibility in resource-constrained settings. Second, although semantic entropy filtering improves corpus quality by excluding uncertain examples, it may also remove valuable complex counterfactuals; future work could explore adaptive or multi-criteria filtering strategies that balance quality, diversity, and training stability. Third, our study focuses primarily on binary gender counterfactuals. Extending this framework to non-binary and intersectional identities, as well as other sensitive attributes (e.g., race, religion, or profession), is an important next step. Fourth, because large LMs and target smaller models may differ in distributional characteristics, alignment mechanisms such as domain-adaptive filtering, grounding mechanisms such as fact-checking modules, and human-in-the-loop validation could further enhance contextual reliability. Finally, expanding this framework to include domain-specific bias detection or multilingual or multimodal extensions would enhance robustness and fairness across broader applications.

6. CONCLUSION

This work presents an effective gender debiasing method that maintains competitive performance in downstream tasks while reducing bias in LMs. Building on classic CDA, which effectively mitigates bias but often weakens language modeling capabilities, our proposed method, *Context-CDA*, enhances the debiasing corpus by leveraging large LMs to generate enriched context. This augmentation minimizes the discrepancy between the debiasing corpus and the original pre-training data, ensuring better alignment and also preserving linguistic fluency. Furthermore, we incorporate semantic entropy filtering to remove uncertain content, improving the overall quality of the generated corpus. Comprehensive evaluation across five diverse model architectures demonstrates that *Context-CDA* is truly model-agnostic, achieving robust and consistent debiasing performance across both discriminative and generative systems. Our method not only mitigates bias effectively but also enhances language modeling performance. As LMs continue to evolve, integrating more sophisticated debiasing techniques will be crucial for building more equitable and more robust AI systems.

7. REFERENCES

[1] Baumann et al. Bias on demand: a modelling framework that generates synthetic data with bias. In *Proceedings of the 2023 ACM Conference on FAccT*, pages 1002–1013, 2023.

[2] Cer et al. Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. *arXiv preprint arXiv:1708.00055*, 2017.

[3] Dagan et al. The pascal recognising textual entailment challenge. In *ML challenges workshop*. Springer, 2005.

[4] De-Arteaga et al. Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *proceedings of the Conference on FAcCT*, pages 120–128, 2019.

[5] Devlin et al. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of ACL*, 2019.

[6] Farquhar et al. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017), 2024.

[7] Fatemi et al. Improving gender fairness of pre-trained language models without catastrophic forgetting. 2021.

[8] Gallegos et al. Bias and fairness in large language models: A survey. *Computational Linguistics*, 50(3):1097–1179, 2024.

[9] Grattafiori et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[10] Han et al. Chatgpt based data augmentation for improved parameter-efficient debiasing of llms. 2024.

[11] He et al. Mabel: Attenuating gender bias using textual entailment data. *arXiv preprint arXiv:2210.14975*, 2022.

[12] Huang et al. Counterfactually-augmented snli training data does not yield better generalization than unaugmented data. *arXiv preprint arXiv:2010.04762*, 2020.

[13] Hugging Face. meta-llama/llama-3.2-1b. `https://huggingface.co/meta-llama/Llama-3.2-1B`, 2024.

[14] Kaneko et al. Debiasing isn't enough!–on the effectiveness of debiasing mlms and their social biases in downstream tasks. *arXiv preprint arXiv:2210.02938*, 2022.

[15] Kuhnet et al. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*, 2023.

[16] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. tau Yih, T. Rocktäschel, S. Riedel, and D. Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks, 2021.

[17] Li et al. Beyond relevance: Factor-level causal explanation for user travel decisions with counterfactual data augmentation. *ACM Transactions on Information Systems*, 2024.

[18] Liang et al. Towards debiasing sentence representations. In *Proceedings of 58th Annual Meeting of the ACL*, July 2020.

[19] Lu et al. Gender bias in neural natural language processing. *Logic, language, and security: essays dedicated to Andre Scedrov on the occasion of his 65th birthday*, 2020.

[20] Maudslay et al. It's all in the name: mitigating gender bias with name-based counterfactual data substitution. *arXiv preprint arXiv:1909.00871*, 2019.

[21] Merity et al. Pointer sentinel mixture models, 2016.

[22] Miller et al. Parlai: A dialog research software platform. *arXiv preprint arXiv:1705.06476*, 2017.

[23] Nadeem et al. Stereoset: Measuring stereotypical bias in pretrained language models. *Preprint arXiv:2004.09456*, 2020.

[24] Nangia et al. Crows-pairs: A challenge dataset for measuring social biases in masked language models. *arXiv preprint arXiv:2010.00133*, 2020.

[25] Qian et al. Perturbation augmentation for fairer nlp. *arXiv preprint arXiv:2205.12586*, 2022.

[26] Radford et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[27] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023.

[28] Ravfogel et al. Null it out: Guarding protected attributes by iterative nullspace projection. *arXiv preprint arXiv:2004.07667*, 2020.

[29] Raza et al. Mbias: Mitigating bias in large language models while retaining context. *Preprint arXiv:2405.11290*, 2024.

[30] V. Sanh, L. Debut, J. Chaumond, and T. Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, 2020.

[31] Schick et al. Self-diagnosis and self-debiasing: A proposal for reducing corpus-based bias in nlp, 2021.

[32] Sheng et al. The woman worked as a babysitter: On biases in language generation. *arXiv preprint arXiv:1909.01326*, 2019.

[33] Singh et al. Fairness in ranking under uncertainty. *Advances in Neural Information Processing Systems*, 2021.

[34] Socher et al. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

[35] K. Sreedhar, T. Kavya, J. Prasad, and V. Varshini. A novel metric-based counterfactual data augmentation with self-imitation reinforcement learning (sil). *International Journal of Advanced Computer Science & Applications*, 16(1), 2025.

[36] Statmt. Statistical and neural machine translation, 2023.

[37] Sun et al. Acamda: improving data efficiency in reinforcement learning through guided counterfactual data augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 15193–15201, 2024.

[38] Suresh et al. A framework for understanding sources of harm throughout the machine learning life cycle. In *Proceedings of the 1st ACM Conference on Equity and Access in Algorithms, Mechanisms, and Optimization*, pages 1–9, 2021.

[39] Tahir et al. Fairness through aleatoric uncertainty. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 2372–2381, 2023.

[40] Tan et al. Assessing social and intersectional biases in contextualized word representations. *Advances in neural information processing systems*, 32, 2019.

[41] Tokpo et al. Fairflow: An automated approach to model-based counterfactual data augmentation for nlp. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 160–176. Springer, 2024.

[42] Urpí et al. Causal action influence aware counterfactual data augmentation. *arXiv preprint arXiv:2405.18917*, 2024.

[43] Wanget al. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

[44] Webster et al. Measuring and reducing gendered correlations in pre-trained models. *Preprint arXiv:2010.06032*, 2020.

[45] Wu et al. Polyjuice: Generating counterfactuals for explaining, evaluating, and improving models. *arXiv preprint arXiv:2101.00288*, 2021.

[46] Wu et al. A novel counterfactual data augmentation method for aspect-based sentiment analysis. In *Asian Conference on Machine Learning*, pages 1479–1493. PMLR, 2024.

[47] Xiao et al. Counterfactual data augmentation with denoising diffusion for graph anomaly detection. *IEEE Transactions on Computational Social Systems*, 2024.

[48] Xie et al. An empirical analysis of parameter-efficient methods for debiasing pre-trained language models, 2023.

[49] Zhao et al. Gender bias in coreference resolution: Evaluation and debiasing methods. *Preprint arXiv:1804.06876*, 2018.

[50] Zhao et al. Learning gender-neutral word embeddings. *arXiv preprint arXiv:1809.01496*, 2018.

[51] Zhu et al. Multilingual machine translation with llms: Empirical results and analysis. *arXiv preprint arXiv:2304.04675*, 2023.

[52] Zmigrod et al. Counterfactual data augmentation for mitigating gender stereotypes in languages with rich morphology. *arXiv preprint arXiv:1906.04571*, 2019.

APPENDIX

A. ABLATION STUDY ON UNCERTAINTY THRESHOLD

To validate the robustness of our proposed *Context-CDA* approach, we conduct an ablation study analyzing the effect of varying the semantic entropy filtering threshold on debiasing and language modeling performance. In our main experiments, we used a default threshold of 30%, removing the top 30% of counterfactuals with the highest semantic entropy. This study evaluates how different thresholds (20% and 40%) influence the effectiveness of the model.

Impact on debiasing performance: Figures 18–21 illustrate the effect of semantic entropy thresholds on intrinsic bias metrics such as Stereotype Score (SS) and CrowS-Pairs Score (CS) for both BERT and GPT-2. At a 20% threshold (i.e., more lenient filtering), the model retains more counterfactuals, including those with moderate uncertainty. This sometimes leads to insufficient bias removal, particularly visible in the slightly higher CS values in Fig. 19. Conversely, at a 40% threshold (i.e., more aggressive filtering), while bias mitigation improves initially due to the exclusion of noisier samples, over-filtering may reduce the diversity of the counterfactual corpus, potentially limiting the coverage of gender-related variations and reducing generalization. Overall, the 30% threshold achieves the best balance: it significantly reduces gender bias and outperforms CDA while avoiding the potential drawbacks of both under- and over-filtering as observed in Fig. 19, 20 and 21.

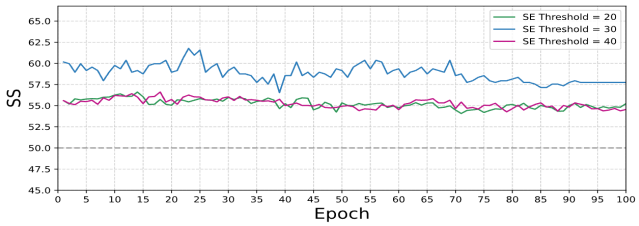

Figure 18: BERT SS Score for SE thresholds 20, 30 and 40.

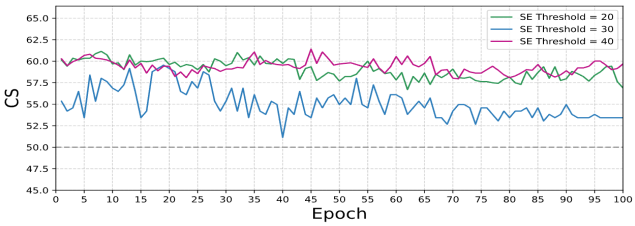

Figure 19: BERT CS Score for SE thresholds 20, 30 and 40.

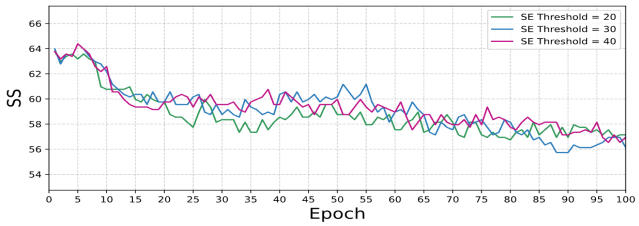

Figure 20: GPT-2 SS Score SE thresholds 20, 30 and 40.

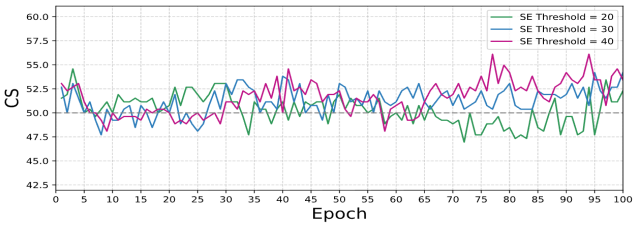

Figure 21: GPT-2 CS Score for SE thresholds 20, 30 and 40.

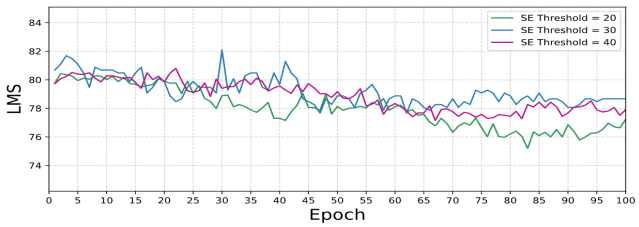

Figure 22: BERT LMS Score for SE thresholds 20, 30 and 40.

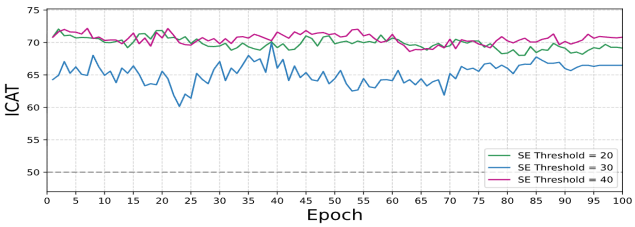

Figure 23: BERT ICAT Score for SE thresholds 20, 30 and 40.

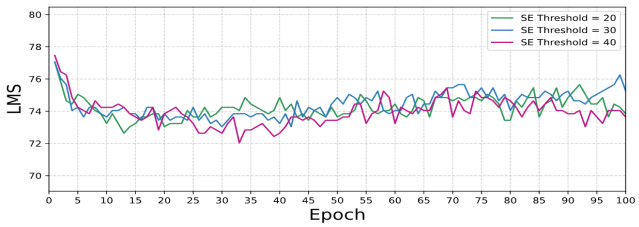

Figure 24: GPT-2 LMS Score for SE thresholds 20, 30 and 40.

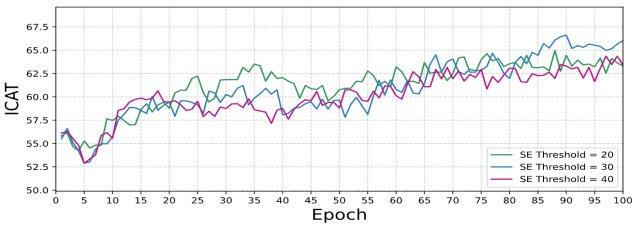

Figure 25: GPT-2 ICAT Score for SE thresholds 20, 30 and 40.

Hence, we choose 30% as the optimal setting for entropy-based filtering in the main experiments. However, the higher SS scores in Fig. 18 indicate the need for a more fine-tuned evaluation of this threshold to arrive at a balanced and optimal value.

Impact on Language Modeling Performance: Figures 22–25 report the Language Modeling Score (LMS) and ICAT score under different entropy thresholds-20, 30 and 40. For both BERT and GPT-2, a 20% threshold tends to retain too many noisy samples, leading to slightly reduced fluency and coherence, as seen in lower LMS scores in Fig. 22. On the other hand, a 40% threshold, while helping eliminate uncertain samples, may discard too many useful and contextually rich augmentations. In contrast, the 30% threshold yields higher LMS and ICAT scores, indicating better preservation of language fluency and semantic understanding. These results reinforce that filtering at this level helps strike an optimal trade-off between reducing uncertainty and retaining the linguistic richness of the training corpus. However, these thresholds can be further refined to reach a more balanced threshold.

On Membership Inference Attacks in Knowledge Distillation*

Ziyao Cui†
Duke University
308 Research Drive
Durham, North Carolina 27708
richard.cui@duke.edu

Minxing Zhang†
Duke University
308 Research Drive
Durham, North Carolina 27708
minxing.zhang@duke.edu

Jian Pei
Duke University
308 Research Drive
Durham, North Carolina 27708
j.pei@duke.edu

ABSTRACT

Large language models (LLMs) are trained on massive corpora that may contain sensitive information, creating privacy risks under membership inference attacks (MIAs). Knowledge distillation is widely used to compress LLMs into smaller student models, but its privacy implications are poorly understood. We systematically evaluate how distillation affects MIA vulnerability across six teacher-student model pairs and six attack methods. We find that distilled student models do not consistently exhibit lower MIA success than their teacher models, and in some cases demonstrate substantially higher member-specific attack success, challenging the assumption that knowledge distillation inherently improves privacy. We attribute this to mixed supervision in distillation: for vulnerable training data points, teacher predictions often align with ground-truth labels, causing student models to learn overly confident predictions that amplify the separability between members and non-members; conversely, for non-vulnerable points, teacher predictions and ground truth frequently diverge, providing inconsistent learning signals. To mitigate this, we propose three practical interventions – restricting distillation to non-vulnerable points, adding a low-dimensional **Bottleneck Projection**, and a normalization variant (**NoNorm**). Experiments show these methods reduce both aggregate and member-specific MIA success while preserving model utility, improving privacy-utility trade-offs for distilled LLMs.[1]

1. INTRODUCTION

Large Language Models (LLMs) have achieved remarkable success due to the scale and diversity of their pretraining data [23, 46]. Moreover, as AI models become increasingly prevalent, privacy concerns have gained significant attention, with researchers investigating vulnerabilities and defenses across model architectures [7, 4, 10]. For LLMs, massive pretraining data introduces serious risks of training data exposure and privacy breaches [43, 41]. More specifically, the massive and heterogeneous nature of these datasets makes it infeasible to fully remove sensitive content, including copyrighted materials [24, 11] and personally identifiable information [26, 37]. Consequently, LLMs may memorize privacy-sensitive data, enabling attackers to infer training membership through Membership Inference Attacks (MIAs) [31].

Prior work has extensively studied the detection of pretraining data in LLMs using MIAs [43, 45, 7, 4]. However, these studies largely analyze models in isolation. In parallel, model compression techniques such as knowledge distillation have been widely adopted to reduce the size and computational cost of modern LLMs, including Llama, Gemma, and BERT [28, 20, 21, 34, 39, 35]. Because distilled student models have fewer parameters and lower capacity, it is commonly assumed that distillation reduces memorization and improves privacy. This assumption, however, has not been systematically evaluated across diverse teacher-student pairs and MIA methods, leaving the privacy effects of distillation insufficiently understood.

To address this gap, as the first contribution in this paper, we evaluate six teacher-student model pairs across multiple architectures using six representative MIAs. We report three key findings. First, distilled student models do not consistently exhibit lower aggregate MIA accuracy than their teacher models. Second, student models can exhibit higher member-specific attack accuracy even when their overall accuracy is lower, which represents a greater practical privacy risk [4]. Third, we provide an explanation: the mixed supervision in distillation – combining ground-truth labels and teacher predictions – can reinforce memorization on vulnerable training data points due to the alignment between teacher predictions and ground-truth, leading the student model to produce overly confident outputs that amplify the separability between members and non-members; on the other hand, for non-vulnerable data points, teacher predictions diverge from ground-truth, making the supervision inconsistent and failing to provide clear privacy benefits.

Motivated by this insight, as the second contribution, we propose and evaluate three targeted interventions to reduce membership leakage in distilled models. First, we introduce a data-selection strategy that restricts distillation to *non-vulnerable* training data points, which reduces exposure to memorized points but may increase student perplexity due to reduced training data. To mitigate this utility loss, we propose two lightweight architectural modifications. The first is a low-dimensional **Bottleneck Projection** that limits representational capacity and discourages memorization. The second replaces layer normalization with **NoNorm**, a parameterized element-wise linear transformation. Across experiments, these interventions consistently reduce MIA success, and the architectural modifications in particular lower attack success without degrading model utility, yielding improved privacy-utility trade-offs compared to naïve distillation.

Outline.

Section 2 formalizes membership inference and introduces the evaluation metrics and protocol. Section 3 reviews related work on

*This research is supported in part by the NSF Project MSPA-2434666. All opinions, findings, conclusions, and recommendations in this article are those of the authors and do not necessarily reflect the views of the funding agencies.

[1]Our implementation and evaluation code are available at `https://github.com/richardcui18/mia-in-kd`.

†Both authors contributed equally to this research.

membership inference, knowledge distillation, and privacy-aware model compression. Section 4 presents a systematic empirical analysis of membership inference vulnerability in teacher and student models and introduces diagnostics for privacy leakage in student models. Section 5 describes the proposed privacy-preserving distillation methods. Section 6 reports experimental results, privacy-utility trade-offs, and additional analyses. Finally, Section 7 discusses limitations and directions for future work; supplementary material and extended ablations are provided in the Appendix.

2. PROBLEM DEFINITION AND EVALUATION METRICS

2.1 Membership Inference Attacks

Consider a LLM $\mathcal{N}$ and a dataset $\mathcal{D}$ used to train $\mathcal{N}$. A **MIA method** [45] M takes a target data point d as input and aims to determine whether $d \in \mathcal{D}$. Denote by $M(\mathcal{N}, d)$ the attacker's prediction, where $M(\mathcal{N}, d) = 1$ if M predicts $d \in \mathcal{N}$ and 0 otherwise. In practice, M may compute a confidence score $M'(\mathcal{N}, d)$ and use a threshold τ to predict the membership of d, that is, $M_\tau(\mathcal{N}, d) = \mathbf{1}[M'(\mathcal{N}, d) > \tau]$, where $\mathbf{1}$ is the indicator function.

2.2 Knowledge Distillation

In knowledge distillation, a teacher model $\mathcal{T}$ is used as a guiding framework to transfer knowledge to a student model $\mathcal{S}$, where the student model $\mathcal{S}$ is learned to mimic the performance of the teacher $\mathcal{T}$ [31, 44].

Let $\mathcal{D}_\mathcal{T}$ and $\mathcal{D}_\mathcal{S}$ denote the training datasets of the teacher model $\mathcal{T}$ and student model $\mathcal{S}$, respectively. We use the standard mixed-supervision distillation objective [17], in which the student model is trained to minimize a weighted sum of the supervised loss on ground-truth labels and a distillation loss that encourages the student model to match the teacher model's predictive distribution. Specifically, for training data points $(x, y) \sim \mathcal{D}_\mathcal{S}$, the objective is

$$\mathcal{L}_{\text{distill}} = \mathbb{E}_{(x,y)\sim\mathcal{D}_\mathcal{S}} \Big[\mathcal{L}_{\text{CE}}\big(y, p_\mathcal{S}(\cdot \mid x)\big) + \lambda \, \text{KL}\big(p_\mathcal{T}(\cdot \mid x) \,\|\, p_\mathcal{S}(\cdot \mid x)\big) \Big]$$

where $p_\mathcal{T}(\cdot \mid x)$ and $p_\mathcal{S}(\cdot \mid x)$ are the predictive distributions of the teacher and student models for training data (prefix) x, $\mathcal{L}_{\text{CE}}$ denotes cross-entropy loss, $\text{KL}(\cdot\|\cdot)$ is the Kullback-Leibler divergence, and $\lambda \geq 0$ controls the trade-off between the two terms.

2.3 Problem Definition

Consider a model $\mathcal{T}$ trained on dataset $\mathcal{D}_\mathcal{T}$. Let $\mathcal{D} \subseteq \mathcal{D}_\mathcal{T}$ be a set of member data points and let $\mathcal{D}'$ be a non-member set such that $\mathcal{D}' \cap \mathcal{D}_\mathcal{T} = \emptyset$. Given a MIA method M_τ with threshold τ, we define the **MIA true positive rate** as

$$TPR(\mathcal{T}) = \frac{1}{|\mathcal{D}|} \sum_{x \in \mathcal{D}} \mathbf{1}[M_\tau(\mathcal{T}, x) = 1],$$

and the **MIA true negative rate** as

$$TNR(\mathcal{T}) = \frac{1}{|\mathcal{D}'|} \sum_{x' \in \mathcal{D}'} \mathbf{1}[M_\tau(\mathcal{T}, x') = 0].$$

We define the **MIA accuracy** on model $\mathcal{T}$ as the average of these two quantities:

$$A(\mathcal{T}) = \frac{1}{2}\big[TPR(\mathcal{T}) + TNR(\mathcal{T})\big]. \quad (1)$$

The central question studied in this paper is whether distilled student models $\mathcal{S}$ are more vulnerable to MIAs than their teacher models $\mathcal{T}$. We assess vulnerability using both aggregate MIA accuracy, $A(\mathcal{T})$ versus $A(\mathcal{S})$, and member-specific MIA accuracy, $TPR(\mathcal{T})$ versus $TPR(\mathcal{S})$, and investigate how distillation can be modified to improve student robustness to membership inference.

3. RELATED WORK

3.1 MIA Methods

MIAs, introduced by Shokri et al. [31], aim to determine whether a data point was used to train a model. While MIAs have been studied in settings such as diffusion models [5] and multi-layer perceptrons [42], applying MIAs to LLMs presents distinct challenges. LLM training data are typically not public [40, 1, 16], complicating evaluation due to missing ground-truth membership labels, and modern LLMs are often trained for a single epoch over massive corpora, reducing classical memorization signals [6, 30]. Despite these challenges, several MIA methods have been developed for LLMs, including loss-based attacks [45], zlib-normalized loss [7], reference-model attacks using likelihood ratio tests [4], and ReCaLL, which leverages relative conditional log-likelihoods [43].

However, existing studies primarily analyze MIAs on individual models in isolation and do not consider how vulnerability changes across related models, such as teacher-student pairs produced by knowledge distillation. In particular, prior MIA methods [45, 7] were not designed to assess privacy trade-offs introduced by distillation. *Our work addresses this gap by systematically evaluating MIA behavior across distilled teacher-student model pairs and using these insights to develop privacy-preserving distillation methods.*

3.2 Knowledge Distillation

Knowledge distillation [17, 3] is a model compression technique in which a smaller student model is trained to mimic a larger teacher model using a combination of supervised loss and a distillation loss that aligns predictions by student and teacher models [14, 44]. Distillation has been widely applied to LLMs, including DistilBERT [28] and subsequent methods that focus on matching output distributions [32, 22, 47]. Other approaches exploit intermediate representations as training signals, enabling the student model to imitate the teacher model's hidden states through layer-wise distillation [27, 33, 20].

Prior work on knowledge distillation has largely emphasized efficiency and performance, with limited attention to privacy. In contrast, *our work studies knowledge distillation through the lens of MIAs, analyzing how the distillation process alters privacy risk across teacher-student model pairs*. Rather than treating distillation as inherently privacy-preserving, we leverage the teacher model's MIA vulnerability as a signal to understand and mitigate privacy leakage in the student model. As distillation becomes a standard component of LLM deployment, incorporating privacy-aware objectives is essential for responsible model compression.

3.3 Privacy in Knowledge Distillation

A growing body of work has studied the privacy implications of knowledge transfer mechanisms such as knowledge distillation. Several approaches propose distillation-based defenses against membership leakage, including ensemble and self-distillation schemes and cross-distillation protocols [36, 9, 29, 48]. These methods, however, focus on specific engineered variants rather than the canonical mixed-supervision distillation pipeline commonly used in practice.

Model	Min-K% (k)	Min-K%++ (k)	ReCaLL Prefix
Pythia	0.10	0.50	7
DistilPythia	0.50	0.30	7
Gemma 2 27B	0.15	0.25	5
Gemma 2 9B	0.20	0.90	30
Gemma 2 2B	0.20	0.10	28
Gemma 2 2B Distilled	0.40	0.20	28
Llama 3.1 8B	0.70	0.10	20
Llama 3.2 1B	0.90	0.10	22
Llama 3.2 3B	0.90	0.10	22

Table 1: Optimal hyperparameters selected via cross-validation for Min-K%, Min-K%++, and ReCaLL across all evaluated target models.

Teacher Model	Student Model	ReCall	Loss	Zlib	Mink	Mink++	Ref Model
Pythia	DistilPythia	**0.555** / 0.565	**0.442** / 0.444	**0.613** / 0.635	**0.316** / 0.414	0.501 / 0.501	0.648 / **0.579**
Gemma 2 27B	Gemma 2 2B	0.667 / 0.667	**0.494** / 0.525	0.556 / **0.481**	**0.543** / 0.556	0.537 / **0.451**	0.494 / **0.420**
	Gemma 2 2B Distilled	0.667 / **0.494**	**0.494** / 0.537	0.556 / 0.556	**0.543** / 0.580	0.537 / **0.432**	0.494 / **0.426**
	Gemma 2 9B	**0.667** / 0.704	0.494 / **0.475**	0.556 / **0.531**	0.543 / 0.543	0.537 / **0.481**	0.494 / **0.426**
Llama 3.1 8B	Llama 3.2 1B	0.702 / **0.682**	**0.303** / 0.311	0.552 / **0.532**	**0.300** / 0.309	**0.311** / 0.335	0.441 / **0.349**
	Llama 3.2 3B	**0.702** / 0.806	**0.303** / 0.314	0.552 / **0.538**	**0.300** / 0.311	0.311 / **0.205**	0.441 / **0.381**

Table 2: Comparison of aggregate MIA accuracy for teacher-student model pairs. Each cell reports teacher / student MIA accuracy, with **bold** indicating the lower (more privacy-preserving) value for each pair and attack method.

In contrast, *our work systematically evaluates this standard pipeline across diverse LLM families and identifies conditions under which distillation amplifies membership leakage*.

Related empirical studies have also questioned the privacy benefits of distillation. For example, Jagielski et al. [19] show that student models may replicate teacher model behaviors under certain conditions, yielding limited privacy gains. Their setting, however, considers distillation solely from teacher model outputs, without access to ground-truth labels, whereas *our study analyzes the more prevalent mixed-supervision objective [17] that combines teacher model predictions and labeled data*. Earlier work by Jagannatha et al. [18] reports lower privacy leakage in DistilBERT compared to BERT, but is limited to a single model pair and a narrow clinical dataset. *Our work expands beyond these settings by examining multiple teacher-student model pairs and datasets, providing a broader understanding of how distillation affects privacy and how it can be modified to improve robustness against membership inference*.

4. MIA IN DISTILLED LLMS

We present a systematic empirical analysis of membership inference vulnerability under knowledge distillation, evaluating six teacher-student pairs across multiple architectures using six representative attacks and the metrics defined in Section 2.3.

4.1 Experimental Setup

4.1.1 Models and Datasets

We evaluate three model families with known training data and publicly available distilled variants. **Pythia** [2] is trained on the ArXiv subset of the Pile [13], with DistilPythia [15] as the student model. **Gemma 2 27B** [38] is trained on the WikiMIA dataset [30], with Gemma 2 9B, Gemma 2 2B, and Gemma 2 2B distilled [38, 35] as student models. **Llama 3.1 8B** [12] is trained on the ArXiv subset of the Pile [13], with Llama 3.2 3B and Llama 3.2 1B [12] as student models.

For non-member evaluation, we use the built-in non-member split of WikiMIA for Gemma models [30]. For all other models, we use the WebInstructSub-prometheus dataset [8], released in May 2024, after the model release dates, ensuring it was not used during training – a method widely adopted by existing works [25, 43, 41].

4.1.2 MIA Methods

We evaluate membership inference using six representative attacks. **ReCaLL** [43] measures changes in conditional log-likelihood when prefixing inputs with non-member context. **Loss** [45] uses the per-example loss as a membership score, while **Zlib** [7] normalizes this loss by zlib compression entropy. The **Reference-model** attack extends the loss-based approach by training shadow models with and without the target data point and performing a likelihood-ratio test. **Min-K%** [30] computes membership scores from the average log-likelihood of the lowest-probability $k\%$ tokens, and **Min-K%++** further calibrates these scores using the mean and standard deviation over the vocabulary.

We leverage cross-validation to select hyperparameters for these attacks: for each dataset and target model, we tune attack-specific hyperparameters – such as the prefix length for ReCaLL and the value of k for Min-K% and Min-K%++ – to maximize average attack performance. We report the optimal hyperparameters in Table 1.

As described in Section 2, each method outputs a confidence score per data point, which is converted to a binary prediction using a threshold selected via ROC analysis to maximize MIA accuracy while ensuring balanced predicted classes. All experiments are conducted on NVIDIA A5000/A6000 GPUs.

4.2 Aggregate MIA Accuracy

Table 2 reports aggregate MIA accuracy for teacher-student model pairs. Across six model pairs and six attack methods, student models frequently achieve MIA accuracy comparable to, and in several cases exceeding, that of their teachers. Specifically, the teacher model

Teacher Model	Student Model	ReCall	Loss	Zlib	Mink	Mink++	Ref Model
Pythia	DistilPythia	**0.658** / 0.673	0.631 / **0.630**	**0.234** / 0.290	**0.444** / 0.683	0.993 / 0.993	**0.620** / 0.636
Gemma 2 27B	Gemma 2 2B	**0.615** / 0.670	**0.606** / 0.661	0.156 / **0.037**	0.761 / **0.495**	0.688 / **0.404**	0.780 / **0.376**
	Gemma 2 2B Distilled	0.615 / **0.532**	**0.606** / 0.716	0.156 / **0.101**	0.761 / **0.376**	**0.688** / 0.771	0.780 / **0.358**
	Gemma 2 9B	0.615 / **0.459**	0.606 / **0.505**	**0.156** / 0.165	0.761 / **0.633**	0.688 / **0.450**	0.780 / **0.339**
Llama 3.1 8B	Llama 3.2 1B	0.968 / **0.937**	**0.451** / 0.513	**0.497** / 0.567	**0.447** / 0.508	**0.486** / 0.604	0.563 / 0.563
	Llama 3.2 3B	**0.968** / 0.974	**0.451** / 0.510	**0.497** / 0.515	**0.447** / 0.502	0.486 / **0.320**	**0.563** / 0.563

Table 3: Comparison of member-specific MIA accuracy for teacher-student model pairs. Each cell reports teacher / student member-specific MIA accuracy, with **bold** indicating the lower (more privacy-preserving) value for each pair and attack method.

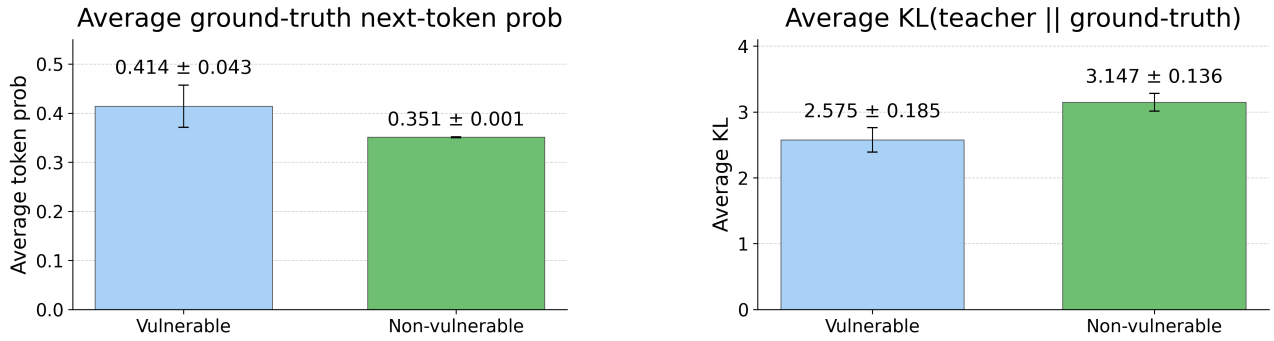

Figure 1: Teacher & ground-truth alignment stratified by membership inference vulnerability. **Left:** Average probability assigned by the teacher model to the ground-truth next token. **Right:** KL divergence between the teacher model's predictive distribution and the ground-truth distribution.

exhibits lower accuracy in 15 cases, the student model in 17 cases, and both models tie in 4 cases.

We further test whether teacher models consistently exhibit higher MIA accuracy than their distilled student models using a one-sided sign test, with the null hypothesis $H_0 : P(A(\mathcal{T}) > A(\mathcal{S})) \leq 0.5$ and alternative hypothesis $H_1 : P(A(\mathcal{T}) > A(\mathcal{S})) > 0.5$, where $A(\mathcal{T})$ and $A(\mathcal{S})$ denote the teacher and student MIA accuracies, respectively. Across all comparisons, the teacher model exhibits higher MIA accuracy in 17 cases, yielding a p-value of 0.43. We therefore fail to reject H_0, indicating that *knowledge distillation does **not** consistently reduce membership inference risk*. This result challenges the common assumption that model compression alone improves privacy.

4.3 Member-Specific MIA Accuracy

Since member-specific privacy risks are more practically significant than aggregate metrics [4], we explore the member-specific MIA accuracy for teacher-student model pairs.

As illustrated in Table 3, the teacher model exhibits lower accuracy in 18 cases, the student model in 16 cases, and both models tie in 2 cases. For Pythia and Llama families, the student model member-specific MIA accuracy increases by 9.81% and 2.29%, respectively, compared to the teacher model. Notably, under the reference-model attack for the Pythia family, the student model achieves lower aggregate accuracy than the teacher model (0.579 vs. 0.648; Table 2) while exhibiting higher member-specific accuracy (0.636 vs. 0.620; Table 3). This demonstrates that aggregate MIA accuracy can obscure increased member-specific vulnerability and that lower overall attack success does not necessarily imply stronger privacy protection.

We also conduct the one-sided sign test with a similar setting as Section 4.2. Across all comparisons, the teacher model exhibits higher MIA accuracy in 16 cases, giving a p-value of 0.432. We again fail to reject H_0, indicating that knowledge distillation also does not consistently reduce member-specific risk.

4.4 Why May Student Models Exhibit Greater Privacy Leakage?

Teacher and student models differ fundamentally in their training signals. Teacher models are trained solely with ground-truth supervision, whereas student models optimize a mixed objective that combines the ground-truth loss with a distillation loss, typically the KL divergence between teacher model and student model predictions. Consequently, student models learn from two supervision signals: ground-truth labels and the teacher model's soft predictions. These signals are not equally aligned across training data points. For training data points that are already vulnerable to MIAs (i.e., MIA on this point is successful) on the teacher model, the teacher model assigns high probability to the ground-truth token and exhibits low divergence from the ground-truth distribution. In this case, distillation reinforces supervised learning, allowing the student model to fit these data points especially well. In contrast, for less vulnerable training data points (i.e., MIA is unsuccessful), teacher model predictions are less confident and deviate more from the ground-truth, introducing noise into the distillation signal and failing to provide clear privacy benefits.

We quantify this effect using two metrics: the teacher model's probability assigned to the ground-truth next token and the KL divergence between the teacher model's predictive distribution and the ground-truth distribution. As shown in Figure 1, attack-vulnerable training data points exhibit higher next-token probabilities and lower KL divergence, indicating strong alignment between supervision signals, whereas less vulnerable data points show the opposite trend. This asymmetry leads student models to preferentially learn and

Model	ReCaLL	Loss	Zlib	Min-K	Min-K++	Ref
Evaluated on Vulnerable ($\mathcal{D}_v$)						
Non-Vulnerable	**0.250**	**0.205**	**0.114**	**0.364**	**0.318**	**0.114**
Full	1.000	1.000	0.977	0.818	0.864	1.000
Evaluated on Non-vulnerable ($\mathcal{D}_{nv}$)						
Non-Vulnerable	1.000	0.976	0.976	0.952	0.929	1.000
Full	1.000	**0.952**	**0.452**	**0.833**	**0.905**	1.000
Evaluated on Member ($\mathcal{D}$)						
Non-Vulnerable	**0.593**	**0.556**	**0.432**	**0.630**	**0.593**	**0.519**
Full	1.000	0.975	0.556	0.827	0.889	1.000
Evaluated on Non-member ($\mathcal{D}'$)						
Non-Vulnerable	0.963	0.988	**0.988**	**0.877**	0.914	1.000
Full	0.963	**0.975**	1.000	0.914	**0.877**	1.000

Table 4: MIA accuracies for DistilPythia student models trained using only non-vulnerable data points vs. trained on the full dataset. **Bold** entries indicate lower MIA accuracy (stronger privacy).

overfit already vulnerable points, amplifying the confidence gap between members and non-members and strengthening MIA decision boundaries.

5. PRIVACY-PRESERVING DISTILLATION METHODS

In this section, we develop methods for reducing membership inference risk in distilled models. We first consider a simple data-selection method that restricts distillation to *non-vulnerable* training data points. Although this approach directly reduces exposure to memorized data points, it also weakens the training signal and can degrade model utility. Motivated by this trade-off, we introduce two lightweight architectural modifications – **bottleneck projection** and **NoNorm** – that can be applied during distillation to reduce memorization while largely preserving utility.

5.1 Distillation on Non-Vulnerable Data

We consider a simple data-selection strategy that uses privacy signals from the teacher model to limit student model exposure to memorized content. We apply a MIA to the teacher model $\mathcal{T}$ on its training dataset $\mathcal{D}$ and partition $\mathcal{D}$ into two disjoint subsets: a *vulnerable* set $\mathcal{D}_v = \{x \in \mathcal{D} \mid M(\mathcal{T}, x) = 1\}$, containing data points identified as members by the attack, and a *non-vulnerable* set $\mathcal{D}_{nv} = \{x \in \mathcal{D} \mid M(\mathcal{T}, x) = 0\}$. By construction, $\mathcal{D}_v \cup \mathcal{D}_{nv} = \mathcal{D}$ and $\mathcal{D}_v \cap \mathcal{D}_{nv} = \emptyset$.

This partition captures privacy-relevant structure in the teacher model: data points in $\mathcal{D}_v$ exhibit behaviors that are easily distinguishable from non-members, whereas those in $\mathcal{D}_{nv}$ do not. Accordingly, a natural approach is to restrict distillation to $\mathcal{D}_{nv}$, which directly reduces exposure to highly memorized data points and limits the transfer of membership signals from teacher to student.

5.1.1 Empirical Evidence of Effectiveness

We evaluate the non-vulnerable-only distillation strategy using DistilPythia derived from a Pythia teacher. Table 4 reports MIA accuracies for six attacks (ReCaLL, Loss, Zlib, Min-K%, Min-K%++, and Reference-model), stratified by four subsets: *Vulnerable* ($\mathcal{D}_v$), *Non-vulnerable* ($\mathcal{D}_{nv}$), *Member* ($\mathcal{D}$), and *Non-member* ($\mathcal{D}'$). We compare two student model variants: distillation using only $\mathcal{D}_{nv}$ ("Non-Vulnerable") and distillation using the full dataset $\mathcal{D}$ ("Full").

Model	Vulnerable	Non-vulnerable	Members
Non-Vulnerable	972.96	188.06	436.01
Full	**94.45**	**149.54**	**118.21**

Table 5: Perplexity of student models evaluated on the Vulnerable, Non-vulnerable, and Member subsets. **Bold** entries indicate lower perplexity (better model utility).

Restricting distillation to non-vulnerable data points substantially reduces attack success on both vulnerable and member subsets. Averaged across attacks, the non-vulnerable-only student reduces MIA accuracy by 75.02% on $\mathcal{D}_v$ and 35.20% on $\mathcal{D}$, while achieving comparable accuracy on non-member data. These results confirm the privacy benefit of excluding highly memorized training data points during distillation.

However, this approach incurs clear utility costs. Table 5 reports perplexity on vulnerable, non-vulnerable, and member subsets, showing that the non-vulnerable-only student model consistently exhibits higher perplexity, reflecting degraded language-model quality due to reduced training data.

Overall, while simple and effective at reducing membership leakage, non-vulnerable-only distillation substantially weakens the training signal by reducing the effective training dataset size. With this, it demonstrates that privacy leakage can be mitigated through data selection, while also highlighting the need for complementary methods that reduce leakage without incurring significant utility loss. This motivates the architectural interventions introduced next.

5.2 Bottleneck Projection

Motivated by evidence that larger models are more prone to memorization and MIAs [7], we introduce a simple architectural modification that limits representational capacity: a low-dimensional **Bottleneck Projection** in the feed-forward network. Instead of the standard single projection from hidden dimension H to intermediate size I (typically $I \approx 4H$), we first project hidden states into a compact bottleneck space of dimension $B \ll H$, followed by expansion to I. This two-step projection has parameter cost $O(HB + BI)$, compared to $O(HI)$ for the standard design, yielding substantial parameter and computational savings when $B \ll I$.

Beyond efficiency, the **Bottleneck Projection** constrains intermediate representations, limiting the model's ability to encode fine-grained, training data-specific signals. This restriction reduces memorization and weakens features exploited by MIAs. The **Bottleneck Projection** method is a lightweight modification that integrates seamlessly into existing transformer blocks and can be applied only to student models, preserving the capacity of the teacher.

5.3 NoNorm

Motivated by Sun et al. [34], who showed that replacing layer normalization can simplify model architecture and reduce inference latency[2], we introduce **NoNorm** as a privacy-preserving technique for knowledge distillation in LLMs. More specifically, the mean and variance computations in layer normalization may inadvertently encode information about the training data, posing a potential privacy risk. To address this concern, we replace layer normalization with a simpler element-wise affine transformation:

$$\text{NoNorm}(\mathbf{h}) = \boldsymbol{\gamma} \circ \mathbf{h} + \boldsymbol{\beta},$$

where $\boldsymbol{\gamma}, \boldsymbol{\beta} \in \mathbb{R}^n$, n is the number of channels, $\mathbf{h}$ denotes the hidden state, and $\circ$ is the Hadamard product.

This modification has two key advantages. First, **NoNorm** avoids computing training data statistics, eliminating a potential channel through which training data information could be memorized and exploited by MIAs. Second, the transformation in **NoNorm** improves inference efficiency, yielding a simpler model that is less prone to memorization and thus potentially more robust to MIAs.

6. EXPERIMENTAL EVALUATION OF PRIVACY-PRESERVING DISTILLATION METHODS

We evaluate the effectiveness of the proposed architectural interventions – **Bottleneck Projection** and **NoNorm** – in reducing MIA vulnerability in distilled models.

6.1 Setup and Baselines

All experiments use DistilPythia student models trained for 30 epochs.[3] We compare four student model variants derived from the same teacher: (i) **None**, a model trained with standard distillation with no privacy protection; (ii) **Bottleneck Projection**, which incorporates a low-dimensional **Bottleneck Projection** in the feed-forward layers; (iii) **NoNorm**, which replaces layer normalization with an element-wise affine transformation; and (iv) **All**, which combines **Bottleneck Projection** and **NoNorm**. Each variant is evaluated using six representative MIAs, with results reported separately for member and non-member data.

6.1.1 Hyperparameter Selection: Bottleneck Projection Dimensionality

To identify the optimal value for the bottleneck projection dimensionality B, we first fix all other training hyperparameters and train DistilPythia student variants for 30 epochs while varying $B \in \{48, 96, 192, 384, 768\}$. We report MIA accuracies for six representative attacks (ReCaLL, Loss, Zlib, Min-K%, Min-K%++, and Reference-model) and perplexity measured on the member subset.

[2]Although NoNorm has been proposed primarily to reduce model latency, it has not been studied as a defense mechanism against MIAs for privacy preservation in the LLM knowledge distillation scenario.

[3]We focus on the Pythia family due to computational constraints.

B	ReCaLL	Loss	Zlib	Min-K	Min-K++	Ref
Evaluated on Member ($\mathcal{D}$)						
48	0.877	0.963	**0.654**	**0.864**	0.877	0.988
96	0.864	**0.914**	**0.654**	0.938	**0.864**	**0.938**
192	0.926	0.975	0.679	0.889	0.914	0.988
384	**0.802**	0.951	0.667	0.926	0.901	0.975
768	0.926	0.975	0.704	0.889	0.938	1.000
Evaluated on Non-member ($\mathcal{D}'$)						
48	0.889	**0.951**	1.000	0.951	0.938	0.988
96	**0.852**	0.975	1.000	**0.852**	0.938	1.000
192	0.926	**0.951**	1.000	0.951	**0.914**	0.988
384	0.926	0.975	1.000	0.951	0.926	1.000
768	0.938	0.963	1.000	0.963	0.951	0.988

Table 6: MIA accuracies under different **Bottleneck Projection** dimension B. Within each column, **bold** indicates the lowest accuracy (best for privacy) and underline indicates the highest accuracy (worst for privacy) across different B values.

The privacy attack results are shown in Table 6. For the member side, smaller bottleneck projection dimensions often reduce MIA accuracy. In particular, $B = 768$ attains the highest MIA accuracy across 5 of the 6 MIA methods, and there is a general decreasing trend in MIA accuracy as B decreases. For the non-member case, we also observe that $B = 768$ achieves the highest MIA accuracy across 4 of the 6 MIA methods, and a similar decreasing trend as before. Therefore, we conclude that a bottleneck indeed has the ability to limit memorization and thus vulnerability to MIA.

B	Perplexity (Member)
48	200.17
96	218.67
192	161.56
384	138.44
768	**134.99**

Table 7: Perplexity on member subset for different **Bottleneck Projection** dimensions B. Underlined values indicate highest perplexity (worst utility); **bold** values indicate lowest perplexity (best utility).

The perplexity results are shown in Table 7. Perplexity on members generally improves as B increases; the largest B tested ($B = 768$) attains the lowest perplexity (best utility, 134.99), whereas $B = 96$ produced the worst perplexity (218.67). Intermediate values ($B = 192$ and $B = 384$) provide a compromise between privacy and utility. Thus, we use $B = 384$ for the **Bottleneck Projection** and **All** models in the following experiments, given its favorable privacy-utility trade-off.

6.2 MIA Accuracies

Table 8 reports MIA accuracy for all student model variants. On member data, the model with no privacy protection (**None**) exhibits the highest attack success across all methods. Introducing **Bottleneck Projection**, **NoNorm**, and their combination (**All**) reduces aggregate member-side attack accuracy by 9.45%, 4.71%, and 4.49%, respectively, indicating that both interventions effectively suppress memorization signals exploited by MIAs.

On non-member data, attack accuracies remain broadly comparable

Model	ReCaLL	Loss	Zlib	Min-K	Min-K++	Ref
Evaluated on Member ($\mathcal{D}$)						
None	1.000	1.000	0.802	1.000	0.951	1.000
Bottleneck Projection	**0.802**	**0.951**	**0.667**	**0.926**	0.901	**0.975**
NoNorm	0.988	1.000	0.728	**0.926**	**0.852**	1.000
All	1.000	1.000	**0.667**	0.938	0.914	1.000
Evaluated on Non-member ($\mathcal{D}'$)						
None	0.963	0.988	1.000	**0.926**	0.963	1.000
Bottleneck Projection	**0.926**	**0.975**	1.000	0.951	0.926	1.000
NoNorm	1.000	0.988	1.000	0.988	0.988	1.000
All	1.000	0.988	1.000	0.963	**0.914**	1.000

Table 8: MIA accuracies for DistilPythia student models under proposed architectural improvements. **Bolded** entries denote the lower MIA accuracy (better privacy).

Model	Perplexity (Member)
None	68.96
Bottleneck Projection	138.44
NoNorm	52.99
All	**51.68**

Table 9: Perplexity on held-out training data points for DistilPythia student models under proposed architectural improvements. **Bolded** entries denote lower perplexity (better language-model utility).

across variants. Relative to the model with no privacy protection, **Bottleneck Projection** slightly reduces non-member accuracy by 1.05%, while **NoNorm** and the combined model (**All**) increase it by 2.19% and 0.46%, respectively. These changes are small compared to the gains on member data. Since member identification constitutes the primary privacy risk in membership inference [4], improvements on the member subset are of primary importance.

Overall, the results show that the proposed architectural interventions substantially reduce member-specific vulnerability – the dominant privacy failure mode – while largely preserving privacy on non-member data.

6.3 Model Utility

We evaluate model utility using perplexity on the member subset, as shown in Table 9. **Bottleneck Projection** exhibits higher perplexity, reflecting the expected cost of reduced representational capacity. In contrast, **NoNorm** improves perplexity relative to the model with no privacy protection (**None**), and the combined model (**All**) achieves the lowest perplexity among all variants, demonstrating that **NoNorm** effectively mitigates the utility loss introduced by **Bottleneck Projection**.

Overall, the results demonstrate a favorable privacy-utility trade-off: while **Bottleneck Projection** alone reduces membership leakage (as illustrated in Table 8) at some cost to utility (as illustrated in Table 9), combining **Bottleneck Projection** with **NoNorm** recovers model utility while maintaining a substantial reduction in membership inference success.

7. CONCLUSION AND FUTURE WORK

In this paper, we study how knowledge distillation affects membership inference vulnerability in large language models. Across six teacher-student model pairs and six MIAs, we find that distillation does not reliably improve privacy: student models can match or exceed their teacher models in aggregate and member-specific attack success. This arises from mixed supervision during distillation, where alignment between teacher model predictions and ground-truth labels on vulnerable training data points induces overconfident student model behaviors.

We propose three mitigation strategies: restricting distillation to non-vulnerable data, introducing a low-dimensional **Bottleneck Projection**, and replacing layer normalization with **NoNorm**. While data restriction increases perplexity, the architectural modifications reduce member-specific MIA success without degrading utility.

Future work includes theoretical analyses of memorization under mixed supervision, adaptive privacy-utility trade-offs, integration with formal privacy mechanisms, and evaluation across broader model families and tasks.

References

[1] A. Anthropic. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 1:1, 2024.

[2] S. Biderman, H. Schoelkopf, Q. G. Anthony, H. Bradley, K. O'Brien, E. Hallahan, M. A. Khan, S. Purohit, U. S. Prashanth, E. Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR, 2023.

[3] C. Buciluǎ, R. Caruana, and A. Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541, 2006.

[4] N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramer. Membership inference attacks from first principles. In *2022 IEEE symposium on security and privacy (SP)*, pages 1897–1914. IEEE, 2022.

[5] N. Carlini, J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramer, B. Balle, D. Ippolito, and E. Wallace. Extracting training data from diffusion models. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 5253–5270, 2023.

[6] N. Carlini, D. Ippolito, M. Jagielski, K. Lee, F. Tramèr, and C. Zhang. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning*

Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023. OpenReview.net, 2023.

[7] N. Carlini, F. Tramer, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. Brown, D. Song, U. Erlingsson, et al. Extracting training data from large language models. In *30th USENIX security symposium (USENIX Security 21)*, pages 2633–2650, 2021.

[8] chargoddard. Webinstructsub-prometheus, 2024.

[9] R. Chourasia, B. Enkhtaivan, K. Ito, J. Mori, I. Teranishi, and H. Tsuchida. Knowledge cross-distillation for membership privacy. *Proceedings on Privacy Enhancing Technologies*, 2022(2):362–382, 2022.

[10] Z. Cui, M. Zhang, and J. Pei. Learning to attack: Uncovering privacy risks in sequential data releases. *arXiv preprint arXiv:2510.24807*, 2025.

[11] A. V. Duarte, X. Zhao, A. L. Oliveira, and L. Li. DE-COP: Detecting copyrighted content in language models training data. In R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, and F. Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 11940–11956. PMLR, 21–27 Jul 2024.

[12] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan, A. Goyal, A. Hartshorn, A. Yang, A. Mitra, A. Sravankumar, A. Korenev, A. Hinsvark, A. Rao, A. Zhang, A. Rodriguez, A. Gregerson, A. Spataru, B. Rozière, B. Biron, B. Tang, B. Chern, C. Caucheteux, C. Nayak, C. Bi, C. Marra, C. McConnell, C. Keller, C. Touret, C. Wu, C. Wong, C. C. Ferrer, C. Nikolaidis, D. Allonsius, D. Song, D. Pintz, D. Livshits, D. Esiobu, D. Choudhary, D. Mahajan, D. Garcia-Olano, D. Perino, D. Hupkes, E. Lakomkin, E. AlBadawy, E. Lobanova, E. Dinan, E. M. Smith, F. Radenovic, F. Zhang, G. Synnaeve, G. Lee, G. L. Anderson, G. Nail, G. Mialon, G. Pang, G. Cucurell, H. Nguyen, H. Korevaar, H. Xu, H. Touvron, I. Zarov, I. A. Ibarra, I. M. Kloumann, I. Misra, I. Evtimov, J. Copet, J. Lee, J. Geffert, J. Vranes, J. Park, J. Mahadeokar, J. Shah, J. van der Linde, J. Billock, J. Hong, J. Lee, J. Fu, J. Chi, J. Huang, J. Liu, J. Wang, J. Yu, J. Bitton, J. Spisak, J. Park, J. Rocca, J. Johnstun, J. Saxe, J. Jia, K. V. Alwala, K. Upasani, K. Plawiak, K. Li, K. Heafield, K. Stone, and et al. The llama 3 herd of models. *CoRR*, abs/2407.21783, 2024.

[13] L. Gao, S. Biderman, S. Black, L. Golding, T. Hoppe, C. Foster, J. Phang, H. He, A. Thite, N. Nabeshima, S. Presser, and C. Leahy. The Pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

[14] J. Gou, B. Yu, S. J. Maybank, and D. Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021.

[15] GPT-4 and Crumb. Distilpythia, 2023.

[16] D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[17] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[18] A. Jagannatha, B. P. S. Rawat, and H. Yu. Membership inference attack susceptibility of clinical language models. *arXiv preprint arXiv:2104.08305*, 2021.

[19] M. Jagielski, M. Nasr, K. Lee, C. A. Choquette-Choo, N. Carlini, and F. Tramer. Students parrot their teachers: Membership inference on model distillation. In *Advances in Neural Information Processing Systems (NeurIPS 2023)*, 2023.

[20] X. Jiao, Y. Yin, L. Shang, X. Jiang, X. Chen, L. Li, F. Wang, and Q. Liu. TinyBERT: Distilling BERT for natural language understanding. In T. Cohn, Y. He, and Y. Liu, editors, *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4163–4174, Online, Nov. 2020. Association for Computational Linguistics.

[21] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*, 2020.

[22] K. Liang, W. Hao, D. Shen, Y. Zhou, W. Chen, C. Chen, and L. Carin. Mixkd: Towards efficient distillation of large-scale language models. *Arxiv preprint*, Nov. 2020.

[23] Y. Liu, J. Cao, C. Liu, K. Ding, and L. Jin. Datasets for large language models: A comprehensive survey. *Artificial Intelligence Review*, 58(12):403, 2025.

[24] M. Meeus, S. Jain, M. Rei, and Y.-A. de Montjoye. Did the neurons read your book? document-level membership inference for large language models. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 2369–2385, 2024.

[25] M. Meeus, I. Shilov, S. Jain, M. Faysse, M. Rei, and Y.-A. de Montjoye. Sok: Membership inference attacks on llms are rushing nowhere (and how to fix it). In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 385–401. IEEE, 2025.

[26] M. Mozes, X. He, B. Kleinberg, and L. D. Griffin. Use of llms for illicit purposes: Threats, prevention measures, and vulnerabilities. *arXiv preprint arXiv:2308.12833*, 2023.

[27] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio. Fitnets: Hints for thin deep nets. In Y. Bengio and Y. LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[28] V. Sanh, L. Debut, J. Chaumond, and T. Wolf. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019.

[29] V. Shejwalkar and A. Houmansadr. Membership privacy for machine learning models through knowledge transfer. In *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI 2021)*, 2021.

[30] W. Shi, A. Ajith, M. Xia, Y. Huang, D. Liu, T. Blevins, D. Chen, and L. Zettlemoyer. Detecting pretraining data from large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[31] R. Shokri, M. Stronati, C. Song, and V. Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pages 3–18. IEEE, 2017.

[32] K. Song, H. Sun, X. Tan, T. Qin, J. Lu, H. Liu, and T.-Y. Liu. Lightpaff: A two-stage distillation framework for pre-training and fine-tuning. *arXiv preprint arXiv:2004.12817*, 2020.

[33] S. Sun, Y. Cheng, Z. Gan, and J. Liu. Patient knowledge distillation for bert model compression. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 4323–4332, 2019.

[34] Z. Sun, H. Yu, X. Song, R. Liu, Y. Yang, and D. Zhou. MobileBERT: a compact task-agnostic BERT for resource-limited devices. In D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2158–2170, Online, July 2020. Association for Computational Linguistics.

[35] Syed-Hasan-8503. Gemma-2-2b-it-distilled, 2024.

[36] X. Tang, S. Mahloujifar, L. Song, V. Shejwalkar, M. Nasr, A. Houmansadr, and P. Mittal. Mitigating membership inference attacks by self-distillation. In *Proceedings of the 31st USENIX Security Symposium (USENIX Security 2022)*, 2022.

[37] X. Tang, R. Shin, H. A. Inan, A. Manoel, F. Mireshghallah, Z. Lin, S. Gopi, J. Kulkarni, and R. Sim. Privacy-preserving in-context learning with differentially private few-shot generation. *arXiv preprint arXiv:2309.11765*, 2023.

[38] G. Team, M. Riviere, S. Pathak, P. G. Sessa, C. Hardin, S. Bhupatiraju, L. Hussenot, T. Mesnard, B. Shahriari, A. Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

[39] I. Timiryasov and J.-L. Tastet. Baby llama: knowledge distillation from an ensemble of teachers trained on a small dataset with no performance penalty. In A. Warstadt, A. Mueller, L. Choshen, E. Wilcox, C. Zhuang, J. Ciro, R. Mosquera, B. Paranjabe, A. Williams, T. Linzen, and R. Cotterell, editors, *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages 279–289, Singapore, Dec. 2023. Association for Computational Linguistics.

[40] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[41] C. Wang, Y. Wang, B. Hooi, Y. Cai, N. Peng, and K.-W. Chang. Con-recall: Detecting pre-training data in llms via contrastive decoding. *arXiv preprint arXiv:2409.03363*, 2024.

[42] L. Watson, C. Guo, G. Cormode, and A. Sablayrolles. On the importance of difficulty calibration in membership inference attacks. In *International Conference on Learning Representations*, 2022.

[43] R. Xie, J. Wang, R. Huang, M. Zhang, R. Ge, J. Pei, N. Z. Gong, and B. Dhingra. ReCaLL: Membership inference via relative conditional log-likelihoods. In Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8671–8689, Miami, Florida, USA, Nov. 2024. Association for Computational Linguistics.

[44] X. Xu, M. Li, C. Tao, T. Shen, R. Cheng, J. Li, C. Xu, D. Tao, and T. Zhou. A survey on knowledge distillation of large language models. *arXiv preprint arXiv:2402.13116*, 2024.

[45] S. Yeom, I. Giacomelli, M. Fredrikson, and S. Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*, pages 268–282. IEEE, 2018.

[46] M. Zhang, Y. Yang, R. Xie, B. Dhingra, S. Zhou, and J. Pei. Generalizability of large language model-based agents: A comprehensive survey. *arXiv preprint arXiv:2509.16330*, 2025.

[47] R. Zhang, J. Shen, T. Liu, J.-L. Liu, M. Bendersky, M. Najork, and C. Zhang. Do not blindly imitate the teacher: Using perturbed loss for knowledge distillation. *arXiv preprint arXiv:2305.05010*, 2023.

[48] J. Zheng, Y. Cao, and H. Wang. Resisting membership inference attacks through knowledge distillation. *Neurocomputing*, 452:114–126, 2021.

Classification with Uncertainty-Aware Multimodal Deep Learning: A Survey

Grigor Bezirganyan
Aix Marseille Univ
CNRS, LIS
Marseille, France
gbezirganyan@gmail.com

Sana Sellami
Aix Marseille Univ
CNRS, LIS
Marseille, France
sana.sellami@univ-amu.fr

Laure Berti-Équille
IRD, ESPACE-DEV
Montpellier, France
laure.berti@ird.fr

Sébastien Fournier
Aix Marseille Univ
CNRS, LIS
Marseille, France
sebastien.fournier@univ-amu.fr

ABSTRACT

Multimodal deep learning has achieved remarkable progress by leveraging complementary information across heterogeneous data sources such as texts, images, audios, and structured signals. While increasingly powerful encoders and fusion mechanisms have improved predictive performance, the reliability of multimodal systems remains a critical challenge. In particular, modality disagreement, distribution shifts, and noisy inputs can lead to overconfident yet incorrect predictions.

Distinct from existing surveys on uncertainty in deep learning [45] or on multimodal learning [8; 135], this survey jointly covers three aspects: (i) the structural foundations of multimodal classification examined through the lens of the uncertainty challenges each design choice introduces; (ii) uncertainty quantification in both unimodal and multimodal settings; and (iii) set-valued classification as a decision-level strategy for cautious multimodal prediction. We first review foundational aspects of multimodal representation learning and fusion strategies, highlighting their structural limitations in modeling inter-modal dependence and the uncertainty challenges each stage introduces. We then examine uncertainty quantification methods in deep learning, including both probabilistic and evidence-theoretic approaches, and analyze how these techniques extend to multimodal settings. Special attention is given to conflict-aware fusion mechanisms and to decision-level strategies such as set-valued classification, which enable more cautious and informative predictions.

Beyond reviewing existing methods, we identify key open challenges, including the modeling of partial dependence between modalities, the need for systematic benchmarking of multimodal uncertainty, and the integration of uncertainty into decision-making pipelines. Finally, we discuss how these reliability challenges extend to emerging multimodal agentic systems. By synthesizing advances across multimodal learning and uncertainty modeling, this survey aims to provide a unified perspective and to outline recent research directions toward more reliable multimodal AI systems.

1. Introduction

In recent years, multimodal deep learning (MDL) has seen increasing adoption in various domains, where fusing information from different data sources, such as images, texts, and audios, can improve predictive performance [129; 34; 115; 74]. Combining information from diverse data sources requires different design choices, including the methods used to extract and encode relevant information from each modality, the choice of when and how to fuse the information, and how to make the final prediction. Consequently, there has been a growing interest in developing multimodal learning architectures that do not only improve prediction accuracy but also reflect the reliability of the fused decision and effectively estimate the uncertainty.

Indeed, these heterogeneous data streams inherently possess fluctuating degrees of noise, occlusion, and missing information. By explicitly modeling both aleatoric uncertainty (data-inherent noise) and epistemic uncertainty (model ignorance), multimodal systems can dynamically calibrate their fusion mechanisms. This allows the model to intelligently down-weight corrupted or ambiguous modalities while actively leaning on more reliable signals, thereby preventing the propagation of silent errors during representation alignment. Furthermore, as these complex models are increasingly deployed in high-stakes environments like clinical diagnostics and autonomous navigation, robust uncertainty estimates are paramount. They provide the vital mechanisms needed for safe out-of-distribution (OOD) detection, mitigating catastrophic multimodal wrong prediction, and ultimately bridging the gap between raw predictive performance and trustworthy AI.

This paper discusses the evolution of multimodal deep learning and multimodal uncertainty quantification methods, highlighting both foundational work and state-of-the-art approaches. It also outlines the key challenges in this domain, including the complexity of existing approaches, the difficulty of han-

dling conflicting or uninformative modalities, and the need for efficient, actionable, and trustworthy predictions from multimodal inputs.

Several surveys address related but distinct problems. [45] provides a comprehensive treatment of uncertainty quantification (UQ) in deep learning, focusing primarily on unimodal architectures. Multimodal learning surveys [8; 135] cover fusion and representation challenges but do not address uncertainty modeling in depth. To our knowledge, no existing survey jointly covers (i) the structural foundations of multimodal classification with an explicit focus on the uncertainty challenges introduced at each stage, (ii) UQ methods in both unimodal and multimodal settings, and (iii) decision-level strategies such as set-valued classification for cautious multimodal prediction. This survey fills this gap by providing a unified perspective connecting all three dimensions, and by showing how limitations in fusion architecture directly motivate uncertainty-aware design.

The remainder of this paper is structured as follows: Section 2 reviews the recent key developments and the current trends in multimodal deep learning. Sections 3 and 4 present uncertainty quantification approaches in unimodal and multimodal settings, respectively. Section 5 discusses the challenges of evaluating multimodal UQ methods. Finally, Section 6 summarizes the main insights and outlines future research directions.

2. Multimodal Deep Learning Foundations

Multimodal (or multi-view) deep learning (MDL) aims to leverage and combine information from diverse data types, such as images, audios, and texts, and other types of signals, that could not previously be jointly integrated, in order to improve the performance of machine learning models. Although there is no universally accepted distinction between the terms multimodal and multi-view, they are often used interchangeably in the literature, with broadly similar definitions [140; 171; 135]. In the same line, we will use the terms multimodal and multi-view interchangeably throughout this paper. We define multimodal data as data that is represented in different forms or collected from different sources, but that describes the same underlying entity or concept [171].

Different modalities often contain complementary information, and combining them can lead to a more comprehensive representation of the underlying real-world entities or phenomena [93]. [8] identify five key challenges in MDL: data representation, translation, modality alignment, fusion, and co-learning described as follows:

- *Representation*: How to represent different modalities in a way that captures both their complementarity and redundancy.
- *Translation*: How to translate information between modalities (e.g., from image to text, or from text to audio). Since we focus on multimodal classification, inter-modal translation will not be discussed further in this article.
- *Alignment*: Ensuring that data from different modalities, such as text, images, and audio, refer to the same underlying concept or instance, whether at the data, feature, or semantic level. For example, in video captioning, alignment ensures that the text accurately describes the visual content. At the semantic level, models may associate concepts across modalities (e.g., linking an image of a cat with the spoken word "cat") despite differences in representation. While some methods enforce alignment explicitly, many models learn it implicitly through supervision on the main task (e.g., classification or captioning) [92].
- *Fusion*: Combining information from multiple modalities to perform a prediction. Different modalities can provide varying levels of information, and may contain noise, inaccuracies, or disagreements.
- *Co-Learning*: Transferring knowledge between modalities. Co-learning mainly explores how models can be benefited from other models trained on different modalities.

Beyond these challenges, we also highlight an additional issue that is highly relevant for multimodal classification: the modality imbalance, closely related to representation learning, when learning rates are different depending on the modality or when one modality dominates the training process, often leading to suboptimal joint representations [162].

In this paper, our primary focus is on multimodal classification, and in this section we review approaches and challenges in representation learning, multimodal fusion, and learning objectives, and modality imbalance. These three aspects are deeply interconnected: the quality of representation learning determines the informativeness of unimodal features; fusion governs how these features are combined into a joint decision; and learning objectives influence both representation and fusion by shaping how modalities are weighted during training. A recurring challenge across all three is modality imbalance, where some modalities dominate while others are underutilized, leading to suboptimal joint models. Addressing these issues is central to designing efficient and reliable multimodal classifiers.

These challenges have direct implications for uncertainty modeling that motivate the content of later sections. The quality of representation learning determines whether per-modality uncertainty can be estimated reliably: noisy or poorly calibrated representations propagate errors into any downstream uncertainty estimate. Fusion design governs how uncertainty is combined across modalities: strategies that assume equal modality reliability cannot detect nor propagate uncertainty arising from inter-modal conflict. Modality imbalance introduces systematic overconfidence when dominant modalities suppress the uncertainty signals of weaker sources. We highlight these implications throughout this section to build a principled motivation for the uncertainty quantification frameworks reviewed in Sections 3 and 4.

Other classical challenges of multimodal learning are less relevant in our scope. The challenge of translation pertains to tasks involving inter-modal generation, which falls outside the focus of this work. Similarly, co-learning typically addresses transfer learning across modalities in different settings. While explicit alignment can sometimes improve classification, it is often not essential, since semantic alignment is usually learned implicitly through supervision.

Scope note. The subsections below review representation learning, fusion strategies, and learning objectives as they

bear on uncertainty in multimodal classification. Rather than providing a comprehensive taxonomy of all MDL architectures, this section deliberately emphasizes the design choices that introduce calibration or conflict challenges, providing principled motivation for the UQ frameworks reviewed in Sections 3 and 4.

2.1 Multimodal Classification Steps

There are various taxonomies for multimodal classification, with one of the most widespread being based on the stage of the pipeline at which fusion occurs. Based on this, multimodal networks are categorized into three types: *early fusion*, *intermediate fusion*, *late fusion* [121]. Some architectures that use a combination of these taxonomies are also often referred as *hybrid fusion*. However, [135] argue that classifying architectures solely by fusion stage is not sufficiently specific to capture the diversity of current multimodal pipelines. Hence, they propose a more detailed taxonomy based on five stages: 1) Preprocessing, 2) Feature extraction, 3) Data fusion, 4) Primary learning, and 5) Final classification. In this paper, we follow the taxonomy provided by [135], while merging the *feature extraction* and *primary learning* stages into a single category referred to as *representation and primary learning*. We thus categorize the stages of multimodal classification pipelines into: 1) Preprocessing, 2) Representation / primary learning, 3) Fusion, and 4) Final classifier (Figure 1).

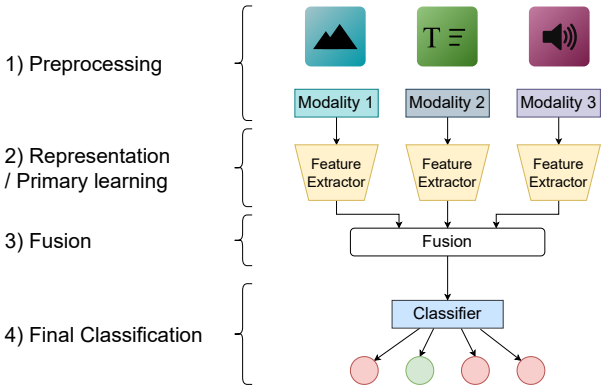

Figure 1: An example of a multimodal classifier divided into the proposed stages.

In many architectures, several of these stages can be shared and performed by the same neural network.

In the following subsections, we will review the main architectures in multimodal deep learning and their advances, focusing on representation learning and data fusion stages.

2.1.1 Preprocessing & Multimodal Representation Learning

This section presents the first two stages of the pipeline, with a particular focus on approaches including multimodal representation learning.

The preprocessing stage can include data cleaning, normalization, addressing missing values, and typically varies depending on the modality. For instance, text preprocessing may include normalization and tokenization; audio preprocessing often involves converting waveforms to spectrograms; and image preprocessing may include cropping, resizing, or normalization. Often, deep learning architectures may omit the preprocessing steps, and perform the learning from raw data.

An important consideration in multimodal classification is how each modality is represented before fusion, since the quality of learned features sets the foundation for cross-modal integration. We therefore review approaches for representation learning, tracing their evolution from handcrafted features to deep encoders and foundation models. A brief overview of representation learning approaches is illustrated in Figure 2.

In the early days of multimodal learning, feature extraction relied primarily on hand-crafted techniques, such as SIFT [98] for images and bag-of-words [53] for text. Fusion was often performed using linear methods such as Canonical Correlation Analysis (CCA) [57], which maximized cross-modal correlations. Extensions such as discriminative CCA [28; 76; 180] incorporated label information, yet these approaches remained limited to modeling linear relationships, motivating the development of more expressive non-linear models.

Neural network-based approaches began addressing these limitations even prior to the deep learning era [32]. With the rise of deep learning, joint training of modality-specific encoders became standard. Early works such as multimodal autoencoders [112] and deep Boltzmann machines [137] demonstrated the benefit of learning shared representations end-to-end, while Deep CCA [4] introduced non-linear cross-modal alignment.

As unimodal deep architectures matured, multimodal systems increasingly leveraged strong modality-specific encoders [38; 63; 6; 182]. Convolutional networks such as AlexNet [81], VGG [134], and ResNet [54] replaced handcrafted visual features, while distributed word embeddings like Word2Vec [108] and GloVe [118] became standard in text. These advances enabled richer latent representations suitable for cross-modal alignment and fusion.

A major paradigm shift occurred with the introduction of transformers [153], whose attention mechanisms allowed scalable modeling across modalities. Originally proposed for NLP, transformers were extended to vision [30] and audio [47], and became central to multimodal architectures. Models such as ViLBERT [99], LXMERT [141], and VisualBERT [91] demonstrated both dual-encoder and shared-encoder strategies for vision–language integration. While powerful, transformers incur quadratic complexity in sequence length. Alternatives such as MLP-Mixer [146] explored attention-free architectures with improved computational efficiency.

More recently, large-scale pre-trained multimodal models have further advanced representation learning. CLIP [120] introduced contrastive pre-training of image and text encoders in a shared embedding space, achieving strong zero-shot performance. Subsequent models such as ALIGN [66] and Flamingo [3] expanded this paradigm. The emergence of *foundation models* extended multimodal learning to broader modality sets; for example, ImageBind [46] aligned images, text, audio, depth, and other signals into a unified embedding space without requiring fully paired data across all modalities. In general, the representation learning stage in multimodal deep learning is an active area of research, since having a good representation of the data is crucial for the performance of the model. The trade-off between the complexity of the model and the quality of the representa-

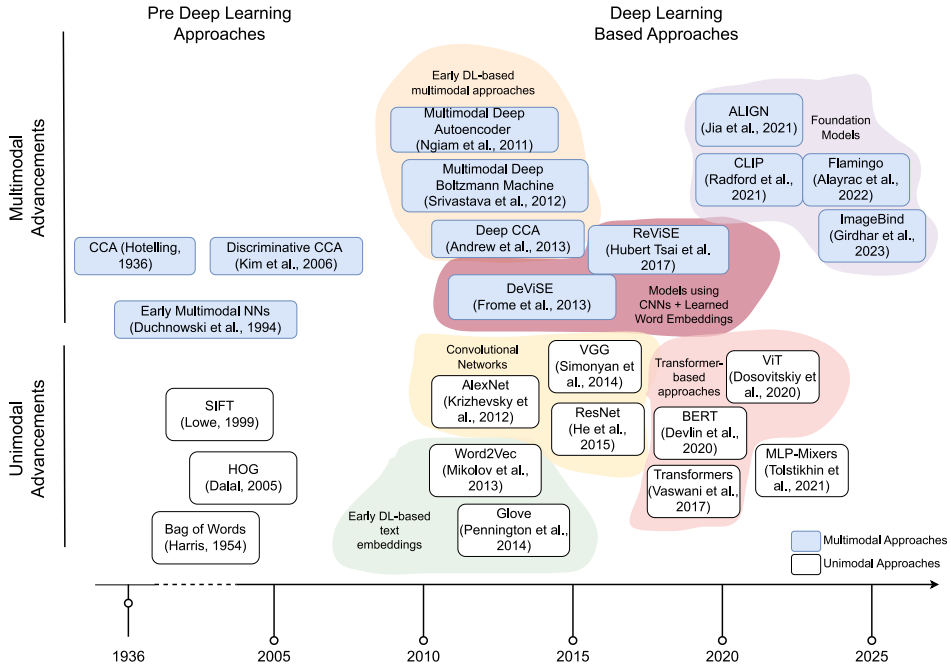

Figure 2: A brief overview of key approaches in unimodal and multimodal representation learning.

tion is an important consideration, as while more complex models can learn better representations, they also require more data and computational resources to train. The learning stage corresponds to the primary learning phase of the pipeline. Depending on the architecture, this learning may occur at a single point, as in early fusion, or at multiple points, as in cross-modality and late fusion strategies.

Uncertainty implication. Representation quality directly bounds the reliability of per-modality uncertainty estimates: a poorly calibrated encoder produces feature distributions whose uncertainty cannot be trusted by downstream fusion mechanisms. Large pre-trained models and foundation models offer better-calibrated representations but introduce epistemic uncertainty about feature-space transferability to the target domain [45].

Having reviewed representation learning, we next discuss multimodal fusion and final classification. Fusion is essential both for integrating features during representation learning and for aggregating predictions in the final decision stage.

2.1.2 Multimodal Fusion & Final Classification

Once suitable modality-specific representations are available, the next design choice concerns how to integrate them. Fusion is the core mechanism by which multimodal systems combine complementary or redundant information, and different fusion stages lead to different trade-offs in flexibility, computational cost, and robustness. Finally, the combined representation or aggregated predictions can be passed through a final classifier, which produces the system's output.

In this section, we will review the main approaches for fusion and final classification in multimodal deep learning. Fusion is usually categorized into *early, intermediate* and *late* fusion strategies. In early fusion, modalities are combined at the input data level and passed through a shared representation learner. In intermediate fusion, each modality is first processed by a separate feature extractor, and their features are then fused for downstream tasks. In late fusion, modality-specific classifiers make independent predictions, which are then aggregated at the decision level. In practice, these strategies are not mutually exclusive, many architectures combine them at different points in the pipeline, which is often referred to as hybrid fusion. Examples of early, intermediate and late fusion strategies are illustrated in Figure 3.

Early (raw data) fusion combines the raw inputs from multiple modalities and processes them with a unified encoder for feature learning and prediction. Because fusion occurs at such an early stage, it is challenging to directly integrate heterogeneous modalities (e.g., text with images, or audio with text). However, when modalities share similar raw representations (e.g., multiple image modalities, audio spectrograms with images, or images with depth maps), early fusion becomes straightforward to implement [138]. In such cases, it can also be computationally efficient, as only a single network is required to process the fused input.

The intermediate (feature) fusion combines information at the intermediate representation (feature) level, rather than at the raw data or classifier output level. This allows for interaction between modalities at a higher level, which is not possible in the early fusion stage. Compared to late fusion strategies, which mostly model modality information independently from one another, intermediate fusion also better models cross-modal correlations.

In the multimodal architectures utilizing intermediate fusion, fusion operation can happen at various points in the architecture. [49] categorized the approaches of intermediate fusion strategies based on when they are fused into *sudden, gradual,* and *multi-flow* fusion. Examples of these strategies can be found in Figure 4. In the *sudden* fusion architectures, all modalities are fused together at the same time in one fu-

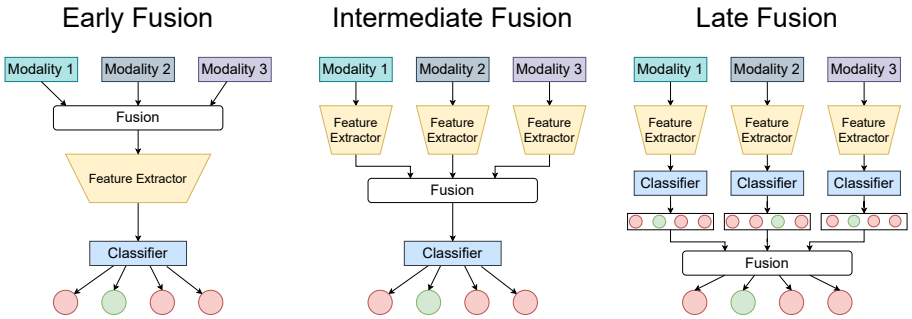

Figure 3: Examples of early, intermediate and late fusion strategies.

sion function. In the *gradual* fusion, a subset of modalities can be fused initially, and then additional modalities can be fused progressively. The gradual fusion approach allows for a hierarchical processing of the modalities. Finally, the *multi-flow* fusion, fuses modalities with different independent fusion functions, which then are fused with each other into a single representation by another function.

The late (decision) fusion combines modality-specific predictions at the decision level. Compared to intermediate fusion, late fusion sacrifices some capacity to model fine-grained cross-modal interactions, but offers greater modularity, robustness to missing modalities, and a natural interface for uncertainty quantification. We organize the existing late fusion approaches into four categories: *simple*, *meta-learning-based*, *optimization-based*, and *uncertainty-aware*, acknowledging that these categories are not exhaustive and that some methods may span multiple groups. A high level overview of the approaches is given in Figure 5.

a) Simple approaches: Simple approaches, as discussed by [78], combine classifier outputs using aggregation functions such as product, averaging, maximum, minimum, and majority voting. These methods typically operate on class-conditional probabilities and include a normalization step to preserve a valid probability distribution. In product fusion, probabilities from each classifier are multiplied and normalized. Averaging computes the mean probability across classifiers, with a common extension being weighted averaging, where modality weights are assigned or learned from a validation set [5; 139; 94]. Maximum and minimum fusion select the highest or lowest probability for each class, respectively, while majority voting assigns the class with the most votes as the final prediction. These simple approaches remain popular because they are easy to implement, but they treat all modalities equally or rely on fixed weights, which may not be optimal for every instance and can perform poorly when modalities are missing or degraded.

b) Meta-learning: To address the limitations of simple approaches, meta-learning-based approaches train a separate model on unimodal score functions to predict instance-specific fusion weights or learn more complex combinations. In the deep learning context, [133] combined two convolutional networks for action recognition via averaging and a multi-class SVM, finding the SVM fusion superior. Other works use logistic regression [152], decision trees, random forests [65], neural networks [62; 122], or adaptive weighting and gating mechanisms [170] to dynamically select or weight modalities. While meta-learning fusion can capture cross-modal dependencies and adapt to varying modality relevance, it requires labeled training data and may not generalize well under distribution shifts.

c) Optimization-based: Optimization-based fusion approaches remove the need for supervised fusion training by formulating fusion as an unsupervised problem, seeking an optimal representation that satisfies predefined structural or statistical constraints. These methods aim to recover a consensus prediction that agrees with all modalities while accounting for noise and outliers. Examples include low-rank matrix recovery [175; 116] and hard-rank-constrained matrix factorization with consistency preservation [29], which leverage structural assumptions in the prediction space. Other strategies estimate modality reliabilities and latent labels jointly using spectral formulations [117], or determine instance-specific fusion weights by optimizing unsupervised criteria such as clarity-index maximization [82]. While these methods avoid the need for supervised fusion training, their performance depends on the validity of their assumptions, and they can be computationally expensive.

d) Uncertainty-aware: Finally, uncertainty-aware approaches explicitly quantify the confidence of each modality and use this information in the fusion function to prioritize more trustworthy and informative sources. For example, [160] estimated uncertainty using deep ensembles and fused predictions with weights derived from uncertainty estimates and modality correlations. A prominent line of work [113; 148; 90; 59] applies the Dempster–Shafer (DS) theory of belief functions [24; 127], a well-established framework for decision-making under uncertainty in which evidence is represented as mass functions and combined using Dempster's rule of combination or its variants [119; 107; 59]. A related family of methods leverages subjective logic [69], which extends DS theory by introducing prior beliefs, subjective opinions, and additional fusion operators [70]. Several recent works [52; 130; 173; 96; 168; 12] have successfully applied subjective logic for multimodal uncertainty quantification, as discussed in Section 4. The uncertainty-aware late fusion category thereby forms the conceptual bridge between fusion architecture design and the principled evidence-theoretic UQ methods reviewed in later sections.

In summary, as outlined in Table 1, late fusion offers flexibility in using modality-specific architectures, handles missing modalities well, and allows straightforward integration of new ones. Its main drawback is weaker modeling of cross-modal interactions and the common assumption that all modalities are equally reliable, which can lead to overcon-

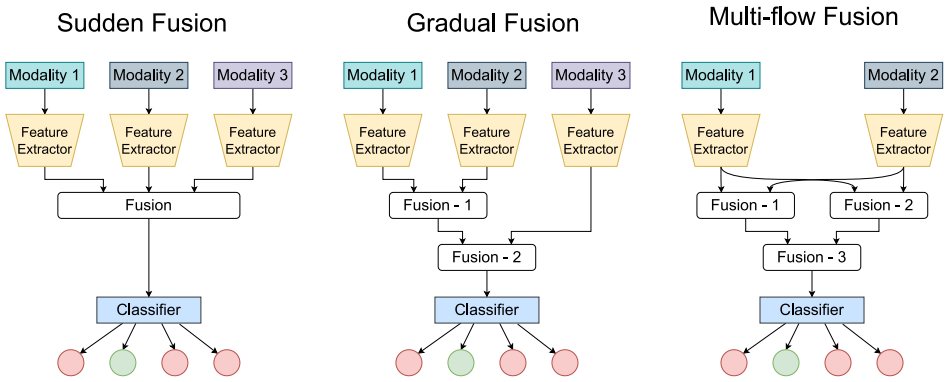

Figure 4: Examples of *sudden, gradual* and *multi-flow* intermediate fusion architectures.

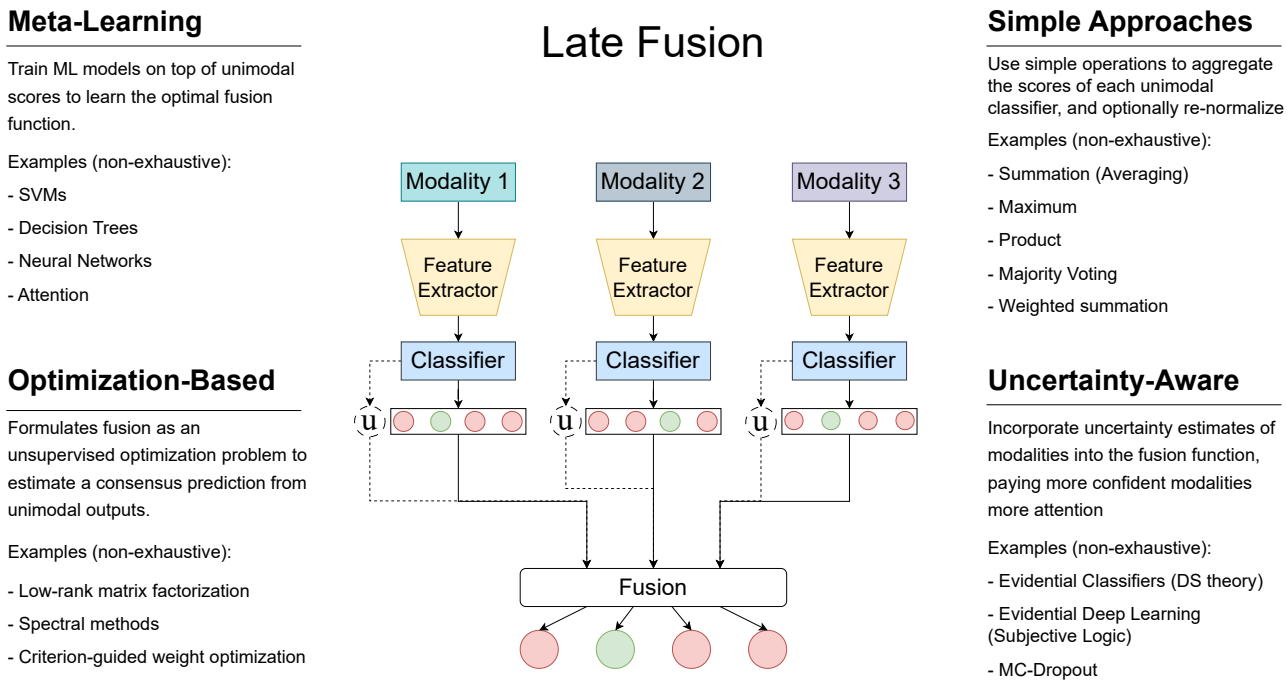

Figure 5: Overview of common multimodal late fusion strategies

fident or inaccurate predictions. Uncertainty-aware methods, such as those based on Dempster–Shafer theory, subjective logic, and other estimation frameworks, address this by weighting modalities according to their estimated reliability and handling potential conflicts.

The final classification stage is the endpoint of the multimodal pipeline, where the fused information is turned into the model's prediction. Depending on the setup, this input may be a joint representation from intermediate fusion or aggregated outputs from late fusion. The final classifier itself can be very simple, such as a linear layer, or more complex, such as a small neural network when richer decision boundaries are needed. The choice of classifier depends mainly on the task, the type of fused input, and the balance between simplicity and accuracy. In late fusion architectures, the fusion function often combines unimodal classifier outputs directly, so the fused result may serve as the final prediction, though some models still add a classifier on top to refine it.

Uncertainty implication. Fusion design governs how per-modality uncertainty propagates to the joint decision. Early and intermediate fusion collapse modality-specific confidence signals into a shared representation, making post-hoc per-modality uncertainty extraction difficult. Late fusion preserves per-modality uncertainty but requires an explicit combination rule; without one, differing modality confidences are silently equalized. Uncertainty-aware fusion strategies that address this limitation are reviewed in Section 4.

While fusion functions govern how information from different modalities is combined, and the final classifier generates

Table 1: Comparison of Fusion Strategies in Multimodal Learning

Fusion Strategy	Advantages	Limitations
Early Fusion	• Simple and effective for homogeneous modalities • Captures low-level cross-modal relationships	• Difficult to apply to heterogeneous modalities • May miss higher-level interactions • Sensitive to alignment and noise • Can struggle with missing modalities
Intermediate Fusion	• Allows modality-specific feature learning • Captures complex high-level interactions • Flexible fusion strategies (e.g., attention, gating)	• More complex design and training • May still struggle with missing modalities
Late Fusion	• Modular and easy to implement • Handles missing or unreliable modalities well • Allows using any architecture per modality	• Weak at modeling cross-modal interactions • Can produce overconfident outputs if modality reliability is not considered

the final prediction, the effectiveness of this combination ultimately depends on how the model is trained. In practice, training dynamics often cause certain modalities to dominate, leading to imbalance problems that undermine the benefits of fusion. We therefore turn next to learning objectives and modality imbalance.

2.2 Learning objectives and modality imbalance

In machine learning, and particularly in deep learning, the loss function plays a central role in shaping the behavior of the model. It determines not only what the model learns, but also how it balances generalization and overfitting, and whether it accounts for uncertainty in its predictions. The loss function encodes the objectives and constraints of the learning process, guiding optimization toward task-aligned solutions.

Beyond influencing output probabilities, the loss function also shapes the internal feature representations learned throughout the network. While in unimodal classification tasks the choice of loss is often straightforward, in multimodal settings it becomes more complex, as it affects both modality-specific learning and cross-modal integration. The optimization strategy has both direct and indirect effects on latent representations before and after fusion. A direct impact arises when loss functions explicitly govern feature representations and their interactions [49]. For example, [67] use contrastive learning with triplet loss to learn modality specific and shared representations. An indirect impact occurs when the loss is applied only at the model's output, while gradients still influence learned features.

Ideally, if all components of the multimodal architecture are trained optimally, the multimodal model should outperform, or at worst behave similarly to, the best unimodal model. However, multimodal models can underperform compared to unimodal models [143; 162]. [162] attribute this partly to overfitting, as multimodal models typically have more parameters and are more susceptible to it. Moreover, different modalities may learn at different rates, allowing faster-learning modalities to dominate training.

To mitigate this issue, [162] proposed optimizing modality-specific and multimodal losses jointly using adaptive weighting based on the overfitting-to-generalization ratio. Although Gradient Blending [162] can improve performance, it is computationally expensive and does not always find optimal weights, potentially leading to suboptimal results [11]. [166] refer to this phenomenon as *greedy learning*, where models rely excessively on easily optimized modalities. They propose estimating each modality's utilization rate from gradient norms and rebalancing training accordingly. Another approach, UMT [31], uses teacher unimodal networks to distill pre-trained unimodal features into the multimodal late fusion architecture. [174] integrate an unsupervised contrastive loss with supervised multimodal classification to address imbalance.

Although effective, these methods increase computational complexity through additional objectives or auxiliary networks. [11] show that such complexity does not necessarily ensure balanced modality learning, and propose simple deterministic weighting as a more efficient alternative. Designing simpler and more efficient modality balancing techniques remains an important research direction.

Uncertainty implication. Taken together, the limitations discussed in this section—unreliable cross-modal representations, fusion strategies that assume equal modality reliability, and training objectives that can amplify modality imbalance—highlight the need for principled uncertainty quantification in multimodal systems. Sections 3 and 4 address this

need, reviewing how uncertainty can be estimated, propagated, and leveraged for more reliable multimodal classification.

3. Uncertainty in Deep Learning

Deep learning models are increasingly being deployed across a wide range of domains, including safety-critical applications such as autonomous driving, medical diagnosis, and financial forecasting. In these contexts, incorrect predictions can lead to serious consequences, such as traffic accidents, misdiagnoses, or substantial financial losses. As a result, understanding the confidence of a model's predictions is essential for ensuring their trustworthiness and reliability. However, modern deep learning models are known to be poorly calibrated and often exhibit overconfidence, even when their predictions are incorrect [50]. To address this issue and improve the trustworthiness of such systems, it is crucial to develop methods that can accurately quantify the uncertainties in the prediction. Based on these uncertainty estimates, a model can either abstain from making a prediction under high uncertainty, or provide several plausible options in the form of a *set-valued classification*, where the true label is expected to be contained within the predicted set.

Although the methods reviewed in this section were primarily developed for unimodal settings, they form the essential building blocks for multimodal uncertainty quantification. In multimodal contexts, per-modality uncertainty estimates serve as inputs to uncertainty-aware fusion mechanisms (Section 4): a modality's reliability score is often derived directly from its predictive uncertainty, and the fusion operator must then combine these estimates in a way that accounts for inter-modal conflict. Understanding the assumptions and limitations of unimodal UQ methods is therefore directly relevant to assessing their suitability for multimodal deployment.

3.1 Types of Uncertainty

In literature, uncertainty is commonly categorized into two main types: *aleatoric* and *epistemic* uncertainty [79]. Figure 6 illustrates the difference between them.

Aleatoric uncertainty arises from inherent randomness in data, such as measurement noise or variability in the underlying process. It is often referred to as *irreducible uncertainty* [1; 55], since it cannot be eliminated by collecting more data or improving the model. However, some works argue that aleatoric uncertainty may be reduced by incorporating additional information, such as more features [64] or more modalities [56].

Epistemic uncertainty, in contrast, stems from a lack of knowledge about the model or data-generating process. It is commonly described as *reducible uncertainty* [1; 55], since it can be decreased with more data or improved modeling. [64] further distinguish between *model uncertainty*, related to the choice of model class, and *approximation uncertainty*, related to training data quality and quantity. [104] additionally introduce *distributional uncertainty*, which arises under distribution shift and is typically high for out-of-distribution (OOD) samples.

Distinguishing between aleatoric and epistemic uncertainty is useful for understanding uncertainty sources and designing appropriate quantification methods. Nevertheless, the distinction remains debated. For example, [110] showed that current methods struggle to disentangle the two types in practice, observing high correlation between them. [165] reported inconsistencies in entropy- and mutual-information-based decompositions of total uncertainty. Alternative definitions have been proposed [48; 124; 123], but no clear consensus has emerged. Therefore, while we acknowledge this distinction and refer to it when relevant, we do not rely heavily on it in the remainder of the paper.

3.2 Uncertainty Quantification Methods

Having defined the aleatoric and epistemic uncertainties in Section 3.1, we can now discuss the methods for quantifying them. Not all methods are able to quantify both types of uncertainty, and some methods are more suitable for certain types than others. In general, uncertainty quantification methods can be broadly categorized into four main categories: (1) Bayesian methods, (2) ensemble methods, (3) single network deterministic methods, and (4) test-time augmentation methods [45]. A high-level summary of these categories is provided in Figure 7.

Bayesian methods [84; 68; 125] apply Bayes' theorem to update prior beliefs $p(h)$ over hypotheses $h \in \mathcal{H}$ given data $\mathcal{D}$:

$$p(h|\mathcal{D}) = \frac{p(\mathcal{D}|h)\,p(h)}{p(\mathcal{D})}, \tag{1}$$

where $p(\mathcal{D}|h)$ is the likelihood and $p(\mathcal{D})$ is the evidence. The posterior $p(h|\mathcal{D})$ reflects the updated belief and captures *epistemic uncertainty*.

In *Bayesian Neural Networks* (BNNs), the hypotheses correspond to weight configurations θ. Instead of a single estimate, BNNs learn a distribution $p(\theta|\mathcal{D})$ and compute predictions via Bayesian model averaging:

$$p(y|x,\mathcal{D}) = \int p(y|x,\theta)\,p(\theta|\mathcal{D})\,d\theta. \tag{2}$$

This is generally intractable for modern networks, so *variational inference* (VI) approximates the posterior by $q_\phi(\theta)$, minimizing the KL-divergence $\mathrm{KL}(q_\phi \,\|\, p(\theta|\mathcal{D}))$. Since $p(\theta|\mathcal{D})$ is unknown, variational inference instead maximizes a loss called evidence lower bound (ELBO), which is equivalent to minimizing the KL-divergence loss up to a constant. *Bayes-by-Backprop* [15] enables training via the reparameterization trick.

Monte Carlo Dropout (MC-Dropout) [39] offers a lightweight approximation by training with dropout and sampling predictions at inference by keeping dropout active. The variance (or entropy) of these predictions estimates uncertainty. MC-Dropout is easy to implement, but studies [154] show its estimates are sensitive to dropout rate, model size, and target magnitude, and may not decrease with more data, limiting its reliability for epistemic UQ.

Gaussian Processes (GPs) [125] are a non-parametric Bayesian approach that models a distribution over functions rather than over finite-dimensional network weights. Formally, a GP is a collection of random variables such that any finite subset has a joint Gaussian distribution:

$$f(x) \sim \mathcal{GP}\big(m(x), k(x,x')\big), \tag{3}$$

where $m(x)$ is the mean function and $k(x,x')$ is a positive-definite kernel encoding correlations between inputs. Starting from a GP prior, conditioning on observed data yields a GP posterior for predictions and uncertainty estimation.

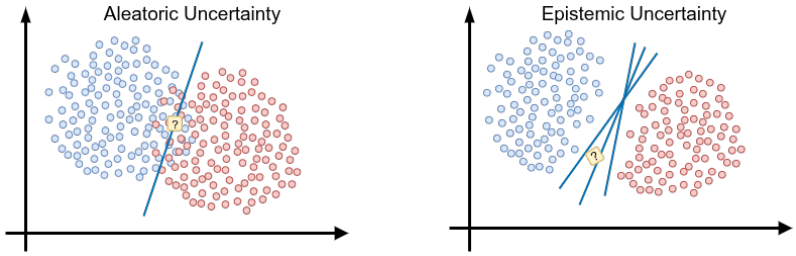

Figure 6: Examples of aleatoric and epistemic uncertainties. In the case of aleatoric uncertainty, even with infinite data and a perfect model, the sample marked with "?" cannot be confidently classified due to inherent overlap between classes. In the case of epistemic uncertainty, the sample cannot be confidently classified because multiple plausible decision boundaries exist, leading to different possible labels. Figure inspired by [64].

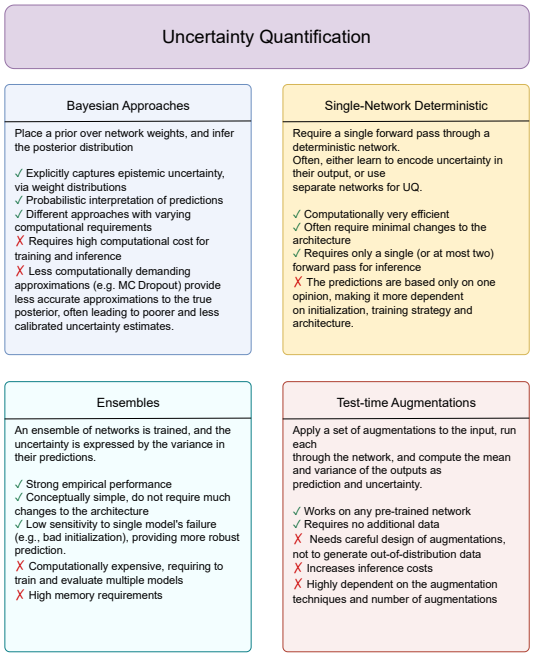

Figure 7: High-level summary of the main uncertainty-quantification paradigms in deep learning. Each quadrant lists general pros (✓) and cons (✗); individual methods may vary in their exact strengths and weaknesses.

While GPs provide well-calibrated uncertainty, their standard form scales as $\mathcal{O}(N^3)$ in the number of training points N. Sparse GPs [136; 145] reduce this to $\mathcal{O}(M^2N)$ using $M \ll N$ inducing points. Performance also depends strongly on kernel choice, which is challenging for structured data such as images. *Deep kernel learning* [164] addresses this by learning feature transformations with deep networks before applying the kernel, and *Deep Gaussian Processes* [22] stack multiple GPs to capture more complex, hierarchical functions.

Neural Processes (NPs) [43; 44] combine neural networks with features of GPs to learn distributions over functions in an end-to-end manner. They divide data into a *context set*, used to condition the model, and a *target set* for prediction. An encoder maps each context pair (x, y) to a latent representation, which is aggregated into a global latent variable. A decoder then combines this variable with target inputs to produce predictions.

Unlike GPs, NPs scale linearly as $\mathcal{O}(N + M)$ for N context and M target samples, making them suitable for large datasets. However, inference quality depends on the chosen context set. Extensions include *Convolutional Neural Processes* [37] for spatial data and *Attentive Neural Processes* [75] for improved context–target interactions.

In the multimodal setting, Bayesian methods and Gaussian Processes have been used to estimate per-modality epistemic uncertainty that is then passed as a reliability weight to uncertainty-aware fusion mechanisms, as discussed in Section 4.

The second category identified by [45] are **Ensembling methods**, which combine predictions from multiple models to improve accuracy and robustness [40]. They are related to Bayesian approaches through the idea of Bayesian model averaging [64]. [83] introduced *deep ensembles* as a practical alternative to Bayesian neural networks: networks trained with different random initializations learn diverse weight configurations, and their predictions are averaged. Uncertainty is estimated from the variance or entropy across ensemble members.

Deep ensembles are easy to implement and often highly performant, but incur high training and inference costs due to multiple full models. To reduce overhead, variants include *Snapshot Ensembles* [58], *Multi-head networks* [87], and *Ensemble Distillation* [106].

Deep ensembles and their lightweight variants have been applied to multimodal UQ by running modality-specific ensemble branches whose cross-member variance serves as a per-modality reliability signal in uncertainty-aware fusion (Section 4).

The third strategy focuses on **single-network deterministic methods**, which estimate uncertainty from a single forward pass of a deterministic network. Many of these methods predict the parameters of a second-order probability distribution over class probabilities, commonly the Dirichlet distribution. For example, *Dirichlet prior networks* [104] parameterize a prior over predictions and are trained to produce sharp Dirichlets for in-distribution (ID) inputs and uniform Dirichlets for out-of-distribution (OOD) inputs, requiring both ID and OOD samples. Similarly, *Evidential Deep Learning* (EDL) [126] parameterizes the *poste-*

rior Dirichlet directly, maximizing evidence for the correct class while encouraging uncertainty (uniform Dirichlet) for others. EDL requires only ID data and will be discussed in more detail later, as it forms a core component of this thesis. Other variants [105; 151; 111; 181] explore both prior and posterior formulations, with differing OOD training needs.

EDL [126] should not be confused with *evidential classification* approaches [26; 102; 147] based on Dempster–Shafer theory (DST) [24; 127]. While subjective logic [69] is conceptually related to DST, the methods differ: EDL predicts Dirichlet parameters to model epistemic uncertainty, whereas DST-based approaches compute mass functions over class hypotheses and combine them via Dempster's rule, deriving uncertainty from residual mass or pignistic probability. In this article, *EDL* refers to the subjective logic formulation unless otherwise specified.

Another group of deterministic approaches are *gradient-based methods*, which infer uncertainty from the magnitude of network gradients at inference. Large gradients indicate greater parameter adjustment would be needed to fit the input, implying higher epistemic uncertainty. [85] used gradients with confounding labels to train an OOD detector, while [61] proposed GradNorm, measuring the gradient norm of the KL divergence between the softmax output and the uniform distribution, requiring no extra classifier. Recent work includes low-rank gradient norms [10] and extensions to segmentation [103]. Gradient-based UQ can be applied post-hoc to trained models without retraining.

Evidential Deep Learning (EDL) via subjective logic is the single-network deterministic method most directly extended to the multimodal setting: per-modality EDL networks generate subjective opinions that are combined by specialized fusion operators, forming the core of TMC, ECML, and related approaches reviewed in Section 4.

Test Time Augmentation methods [158; 7; 101] try to create several augmentations for each test sample, and then pass them through the model to obtain the predictive distribution. Although it is a simple technique, there are many open questions such as what types of (valid) transformations one shall use, how many, and what's the quality of the quantified uncertainty. One solution to the problem of choosing the right transformations was suggested by [158], who proposed a search algorithm to find test-time augmentations policy, based on the predictive performance on validation set. Similar to deep ensembles and Bayesian methods, test time augmentation also requires multiple forward passes through the model, which can be computationally expensive.

Test-time augmentation has been used in multimodal settings as a lightweight alternative to ensemble-based reliability estimation: augmentation variance for each modality provides an uncertainty signal that can be integrated into adaptive fusion strategies (Section 4).

In summary, uncertainty quantification in deep learning can be achieved through a variety of approaches, ranging from Bayesian inference and ensemble strategies to deterministic single-pass and gradient-based methods. Each paradigm offers trade-offs in terms of computational cost, scalability, and the type of uncertainty captured, with no single approach being universally optimal. In the multimodal setting, these challenges are amplified by potential modality conflicts and varying information quality, making reliable UQ essential for robust decision-making. Specifically, Bayesian and ensemble methods are well suited for estimating per-modality epistemic uncertainty, while single-network deterministic methods such as Evidential Deep Learning (EDL) offer the computational efficiency required for real-time fusion. Section 4 shows how these paradigms are extended to multimodal architectures, where per-modality uncertainty estimates must be combined alongside conflict detection.

Table 2: Mapping of unimodal UQ paradigms to their multimodal extensions reviewed in Section 4. Each paradigm contributes a specific type of per-modality signal that uncertainty-aware fusion mechanisms consume.

UQ Paradigm (§3.2)	**Role in multimodal UQ (§4)**
Bayesian / MC-Dropout	Per-modality epistemic uncertainty → reliability weights in ensemble fusion
Deep Ensembles	Cross-member variance → modality confidence in uncertainty-aware late fusion
EDL / Subjective Logic	Subjective opinions → BCF/CBF/ECML multimodal fusion (TMC, ECML, MMLF)
Dempster–Shafer	Belief functions → DS combination rule for multimodal evidence fusion
Test-time Augmentation	Augmentation variance → per-modality reliability in adaptive fusion

One way to utilize uncertainty estimates in safety-critical or high-stakes applications is to allow the model to abstain from overly confident single-label predictions when uncertainty is high. *Set-valued classification* (SVC) formalizes this idea by returning a set of plausible labels whose size reflects the model's uncertainty, thereby reducing the risk of critical misclassifications while still providing actionable information. In the next section, we review the principles and methods of SVC, their advantages and shortcomings.

3.3 Set-valued classification

Most UQ techniques operate in the *precise classification* setting[1], where the model outputs one class or abstains under high uncertainty. However, in assisted decision-making scenarios such as medical diagnosis or risk assessment, suggesting a small set of plausible labels is often more useful than fully rejecting a prediction. *Set-valued classification* (SVC) [19] addresses this by returning a subset of candidate labels in uncertain cases. An example of application of SVC is illustrated in Figure 8. Empirical evidence [21] shows that such set-valued predictions can improve human decision-making compared to fixed top-k suggestions or no assistance.

There is a natural correspondence between epistemic uncertainty and the size of the prediction set [64]. Large sets reflect high uncertainty, while singleton sets correspond to

[1] *Precise classification* refers to returning a single label, in contrast to *set-valued* or *imprecise* classification, which returns multiple labels.

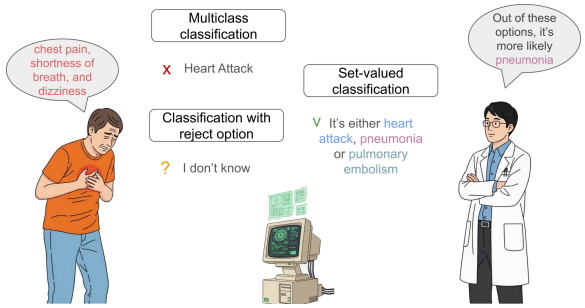

Figure 8: An example of application of set-valued classification in healthcare. Unlike multi-class classification and classification with reject option, set-valued classification proposes several plausible diagnosis options to the doctor, who makes the final decision.

confident predictions. However, under the open-world assumption, high epistemic uncertainty may indicate out-of-distribution samples. In such cases, predicting the full label set is inappropriate, and many approaches combine SVC with rejection or "null set" predictions [64].

Set-valued classification thus acts as a decision-level complement to the UQ methods surveyed in Section 3: where those methods estimate uncertainty, SVC translates it into actionable prediction sets. This relationship is especially important in multimodal settings, where disagreement between modalities often produces structured, irreducible ambiguity that is better expressed by a set of plausible classes than by a single forced prediction. The methods reviewed below are therefore directly relevant to the uncertainty-aware multimodal architectures discussed in Section 4.

Several methodological families have been proposed.

3.3.1 Top-k and Thresholding Approaches

Top-k classification returns the k most probable labels, but uses a fixed set size regardless of confidence. Threshold-based approaches include all labels whose predicted probability exceeds a predefined threshold, allowing adaptive set sizes but being sensitive to calibration errors [109]. Average-k strategies [25; 42] control the expected set size during training. While simple and efficient, these approaches rely directly on predicted probabilities and do not provide formal uncertainty guarantees.

3.3.2 Coverage-based Approaches

Conformal Prediction (CP) [155; 128] constructs prediction sets with distribution-free coverage guarantees. Efficient variants such as split conformal prediction [18] make the method practical for large-scale settings. However, CP guarantees only marginal coverage and may produce overly large sets [109; 163]. Moreover, classical CP does not provide calibrated probabilities for labels within the set, though recent extensions address this limitation [80].

3.3.3 Utility-based Approaches

Utility-based methods [23; 20; 178; 109] formalize SVC as an expected utility maximization problem. Let $u(y, \hat{Y})$ denote the utility of predicting set $\hat{Y}$ when the true label is y:

$$u(y, \hat{Y}) = \begin{cases} 0, & y \notin \hat{Y}, \\ g(|\hat{Y}|), & y \in \hat{Y}, \end{cases} \tag{4}$$

where g controls the trade-off between accuracy and set size. The Bayes-optimal set maximizes the expected utility under $P(c|x)$. In practice, efficient algorithms evaluate only a limited number of candidate sets [109]. These approaches assume well-calibrated probabilities, which motivates integrating uncertainty-aware models [50].

3.3.4 Evidence-Theoretic and Imprecise Probability Approaches

Dempster–Shafer (DS) theory [24; 127] and related evidential neural networks [26] naturally extend to SVC by assigning belief mass to subsets of labels. Extensions to deep learning [102; 147; 27] enable set-valued decisions while preserving uncertainty modeling, though computational complexity can increase with the number of classes. The same DS framework is extended to multimodal fusion in Section 4, where combination rules operate on modality-level evidence sources rather than on class-level belief functions within a single model.

Imprecise probability approaches [177; 172; 156; 150] represent uncertainty via sets of probability distributions (credal sets), supporting cautious decision rules. While theoretically appealing, such methods may be computationally demanding or difficult to scale [114].

Subjective logic–based approaches [69; 126] also provide mechanisms for representing composite hypotheses. Extensions such as HENN [88] model classification ambiguity but are often tailored to multi-label rather than classical SVC settings. The EDL component of these methods is extended to multimodal classification in Section *4.1.2*, where per-modality evidential networks produce subjective opinions that are combined via specialized SL fusion operators such as BCF, CBF, and ECML.

3.3.5 Summary of Limitations

Despite their diversity, existing SVC approaches face common challenges. Thresholding methods lack robustness to calibration errors. Conformal prediction provides coverage guarantees but may produce large sets. Utility-based methods depend on reliable probability estimates. Evidence-theoretic and imprecise probability approaches can become computationally demanding for large label spaces. Overall, effective SVC requires accurate uncertainty quantification, making it closely tied to the quality of the underlying UQ method. In multimodal settings, this dependency becomes even more pronounced: unreliable per-modality uncertainty estimates (e.g., due to modality imbalance or conflict, as identified in Section 2) propagate directly into suboptimal prediction sets. Addressing these limitations motivates the conflict-aware multimodal UQ approaches reviewed in the following section.

SVC in multimodal settings. The SVC methods reviewed above were developed primarily for unimodal models, yet they apply directly to multimodal systems via late fusion: each modality produces an independent evidence mass or probability distribution, which are combined before the SVC decision rule is applied. In this sense, SVC is a natural companion to uncertainty-aware late fusion (Section 2.1): where fusion produces a joint uncertainty estimate over the label space, SVC translates that estimate into a calibrated prediction set. Disagreement between modalities—a form of structured aleatoric uncertainty that cannot be reduced by gathering more data from a single source—is especially well handled by evidence-theoretic SVC methods, since DS theory and subjective logic can represent conflictive opinions as

high-uncertainty mass distributions before any decision rule is applied. The multimodal UQ frameworks reviewed in the following section leverage precisely these properties.

4. Multimodal uncertainty quantification

In the previous sections, we discussed uncertainty quantification (UQ) and set-valued classification (SVC) in the unimodal setting. Multimodal deep learning introduces additional challenges, especially for UQ and SVC. Ideally, incorporating complementary information from multiple modalities should reduce aleatoric uncertainty [56]. In practice, however, uncertainty can also increase when modalities are misaligned or contradictory. For example, in medical diagnosis, an X-ray may suggest one condition while an MRI suggests another. In such cases, it is essential to estimate both per-modality uncertainty and the combined uncertainty of the multimodal system.

While many UQ methods exist for unimodal learning, multimodal UQ remains relatively less explored. A straightforward approach is to treat the multimodal architecture as a single model with one input and one output, then apply unimodal UQ methods directly. However, this ignores modality-specific properties and prevents separate uncertainty estimation for each modality. Hence, various frameworks and approaches have been proposed for quantifying and integrating uncertainty into multimodal deep learning. These methods build directly on the fusion architectures reviewed in Section 2: uncertainty-aware late fusion (Section 2.1) already incorporates elementary modality-reliability weighting. The approaches discussed here extend this idea with principled probabilistic and evidence-theoretic frameworks, enabling systematic conflict detection and uncertainty propagation that simpler fusion strategies cannot provide. Figure 9 provides a brief timeline of some of the key approaches in this domain. For clarity,

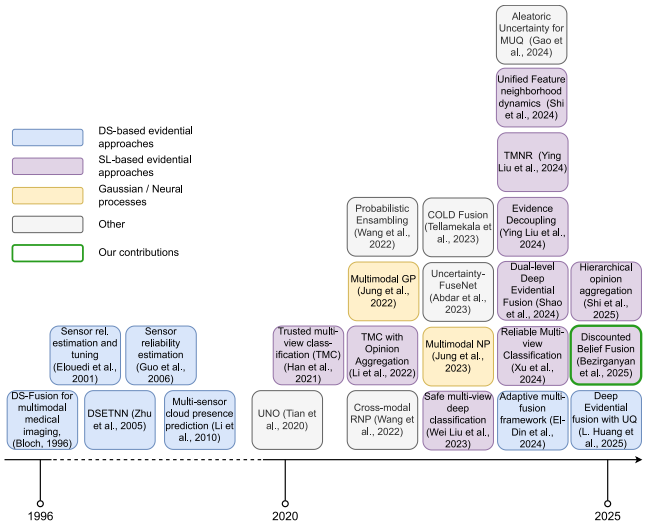

Figure 9: A timeline of non-exhaustive collection of key approaches in multimodal uncertainty quantification.

4.1 Evidence-Theoretic Approaches

One family of methods adopts evidence-theoretic frameworks to explicitly model uncertainty, belief, and inter-modal conflict. These approaches depart from classical probabilistic representations and operate on mass functions or opinion-based representations.

4.1.1 Dempster–Shafer Theory

Among the frameworks for fusing uncertain information, *Dempster–Shafer* (DS) theory [24; 127] is one of the most widely used in multimodal learning. It offers a principled way to combine evidence from multiple sources and to model both uncertainty and imprecision. [14] highlighted features that make DS theory particularly suitable for multimodal classification:

- flexible modeling of uncertainty and imprecision,
- ability to handle variable source reliability, and
- a well-defined combination rule for merging independent evidence.

The *Dempster's rule of combination* fuses multiple beliefs into a single coherent representation. However, in cases of strong conflict between sources, it can yield counter-intuitive results [176]. This limitation has led to numerous alternative combination functions and conflict management strategies [107].

Before the deep learning era, DS theory was already applied to multimodal classification and sensor fusion [35; 9; 51; 89], laying the groundwork for modern architectures. More recently, DS theory has been incorporated into multimodal deep learning [33; 59]. The work of [59] introduced a learnable discounting factor for modality reliability. While effective, their approach assigns a fixed discount per modality and class, limiting adaptability to discrepancies between training and deployment. In contrast, [12] compute sample-specific reliability estimates, enabling dynamic adjustment to modality misalignment, noise, and other real-world inconsistencies.

4.1.2 Subjective Logic

Subjective Logic (SL) [69] extends DS theory, retaining its evidence-combination capabilities while offering a flexible representation of uncertainty and imprecision. [52] introduced one of the first SL-based multimodal deep learning methods, *Trusted Multi-view Classification* (TMC), in which each modality is modeled by an evidential deep learning network [126], and evidences are fused via the *Belief Constrained Fusion* (BCF) rule. Similar BCF-based strategies have been explored by [130] and [173], but since BCF is a SL adaptation of Dempster's rule, it inherits its limitations under high conflict.

[96] employed *Cumulative Belief Fusion* (CBF), an SL operator designed for independent sources contributing new evidence, thereby always reducing uncertainty [69]. However, this behavior can be problematic when strongly conflicting views are fused.

[168] proposed *Evidential Conflictive Multi-view Learning* (ECML), which uses average pooling in belief space and a conflict-penalizing loss. Intended for dependent sources, ECML decreases uncertainty if the new view is less uncertain and increases it if more uncertain. For two views, the combined uncertainty is the harmonic mean of individual uncertainties. Nonetheless, two low-uncertainty but conflicting views can produce undesirably low combined uncertainty, and the non-associative nature of the operator [69] makes it sensitive to fusion order.

To further enhance reliability, [100] introduced trust-based discounting, learning modality- and input-specific trust scores via a separate network. While effective, this adds computational cost and may not generalize well from clean training data to noisy or misaligned test data.

[95] approach the problem from a different perspective, proposing a fusion method that guarantees the fused prediction is at least as accurate as the best unimodal prediction. In other words, their approach ensures that fusion does not deteriorate the prediction quality. [169] addressed the potential label noise in training data, by estimating per-view uncertainty and refining mislabeled samples using pseudo-labels and mixup [179].

In summary, SL-based multimodal UQ methods encompass a range of fusion strategies, including adaptation of Dempster's rule, cumulative and averaging fusion, and more recent approaches that introduce conflict mitigation or trust-based discounting. While these approaches improve robustness and reliability, they still face challenges, especially under strong inter-modal disagreement, and generalization to noisy or misaligned modalities.

4.2 Non-Evidence-Based Multimodal UQ Approaches

In contrast to evidence-theoretic methods, the following approaches estimate uncertainty through probabilistic modeling, representation-level signals, or uncertainty-guided fusion mechanisms, without explicitly modeling inter-modal conflict. For example, [41] proposed *Embracing Aleatoric Uncertainty* (EAU), which models per-modality aleatoric uncertainty via Gaussian embeddings and learns a fusion that is robust to noisy inputs. [16] introduced *HyperDUM*, a deterministic feature-level UQ method based on hyperdimensional computing, quantifying channel- and patch-level epistemic uncertainty before fusion. Other works, such as [17] and [2], leverage ensemble-based techniques or Monte Carlo dropout to estimate predictive uncertainty, without explicitly modeling conflicts between modalities.

Cross-modal Random Network Prediction [161] estimates uncertainty by comparing the outputs of a fixed, randomly initialized network with a smaller, trainable predictor over the feature space, leveraging discrepancies to assess uncertainty. These uncertainty estimates then inform a fusion mechanism that adaptively weights modalities during classification or segmentation tasks.

The *uncertainty-aware noisy-or (UNO)* approach [144] combines multiple uncertainty metrics, such as predictive entropy, mutual information, and a novel spatial temperature network, and propose a novel Noisy-Or fusion, which takes into account the uncertainties of the modalities, prioritizing more confident ones. *COLD Fusion* [142] models each modality as a latent Gaussian distribution and interprets the variance of these distributions as a measure of modality's confidence.

[73] and [72] propose alternative multimodal UQ approaches based on Gaussian processes and neural processes, respectively. While these methods perform well, the Multimodal Gaussian Process is computationally expensive due to its non-parametric nature. The Multimodal Neural Process is relatively faster; however, its results are highly dependent on the chosen context set, and there are currently no theoretically guaranteed methods to obtain an optimal context set.

While these approaches often improve robustness to noise or missing modalities, they generally assume either statistical independence or implicit alignment between modalities. As a result, they may produce overconfident predictions when strong inter modal disagreement occurs. To address this issue, several works have proposed to explicitly separate different types of information carried by each modality. For instance, [131] suggested disentangling common and view specific information between modalities. The objective is to isolate features consistently shared across views from those that remain unique to each modality. This separation allows the model to rely on common representations for cross view agreement while preserving view specific cues that may provide complementary discriminative information. Building on this idea, [97] proposed a Dynamic Evidence Decoupling framework that operates in the evidential space. In this formulation, each view's opinion is decomposed into *consistent* evidence, shared across modalities and trained to align with the ground truth, and *complementary* evidence, which captures modality specific signals and is allowed to remain uncertain. However, separating shared and modality specific evidence does not fully address situations where modalities strongly disagree. To explicitly account for such conflicts, [12] introduced a multimodal classification framework based on evidential deep learning and subjective logic. Their method detects conflictive modalities and applies a sample specific discount factor to their evidence, increasing predictive uncertainty when modalities disagree while maintaining low uncertainty for well aligned inputs. Since conflict detection, discounting, and fusion are parameter free, the approach remains efficient and can handle conflicts at test time even when such cases were absent during training.

textbfSummary. Overall, no single uncertainty quantification approach universally dominates in multimodal settings. Probabilistic methods provide strong theoretical grounding but often face scalability challenges, while ensemble-based approaches offer robustness at the cost of computational efficiency. Evidence-theoretic frameworks are particularly well suited to multimodal learning, as they explicitly model inter-modal conflict, but may suffer from instability under high disagreement. These trade-offs highlight the need for hybrid approaches that balance robustness, scalability, and principled uncertainty modeling.

5. Evaluation of Multimodal UQ

Evaluation protocols for multimodal uncertainty quantification remain heterogeneous across studies. Most works report task-specific metrics such as accuracy, F1 score, or correlation to evaluate predictive performance, while uncertainty-related indicators such as predictive entropy are sometimes used to assess the confidence of model predictions. However, there is currently no standardized evaluation protocol specifically designed for multimodal uncertainty quantification.

In practice, evaluating uncertainty-aware models is further complicated by the fact that the true level of uncertainty present in the data is rarely known, and datasets may not accurately reflect the variability encountered in real-world environments. As a result, it becomes difficult to assess how well uncertainty quantification algorithms perform under different uncertainty conditions. Moreover, deep learning models may behave differently depending on the level of uncertainty in the data, for example when inputs are corrupted by noise or contain ambiguous information. For this reason,

several works introduce controlled perturbations in the data, such as additional noise, in order to analyze how uncertainty estimates and predictive performance evolve under varying uncertainty levels. Since different uncertainty quantification approaches target different types of uncertainty, having mechanisms that allow the injection of diverse uncertainty sources can facilitate more systematic evaluation..

Several datasets have been used in multimodal uncertainty quantification settings. A notable line of work [52; 73; 71] has employed datasets such as HandWritten[2], CUB[3], Scene15[4], and Caltech101[5]. These datasets typically extract different features from unimodal sources to create a multi-view setup. While they have been instrumental, they primarily repurpose unimodal data for multimodal tasks, underscoring the need for more comprehensive and inherently multimodal datasets to better evaluate uncertainty in deep learning models.

Furthermore, the current approaches that introduce uncertainty into the data [52; 73; 71] add Gaussian noise to the views or the extracted features. While Gaussian noise does increase uncertainty, it does not accurately reflect the noise that can be found in real-world datasets and this process lacks fine-grained control over the type of uncertainty being injected.

Additionally, how different modalities' uncertainties interact significantly impacts the overall multimodal uncertainty. When both modalities encode redundant information, the total uncertainty might not decrease. Conversely, conflicting information can lead to increased uncertainty, while complementary information can reduce it. A deeper understanding of these phenomena is crucial. Fine-grained control over individual modalities' uncertainties opens the way for more theoretical research based on empirical observations.

To support the analysis of uncertain multimodal data and the evaluation of uncertainty quantification techniques in multimodal learning, [13] introduce a dataset together with an uncertainty generator package. This package provides several mechanisms for injecting uncertainty, including controlling data diversity, adding different types of real-world noise, randomly switching labels to their closest class, and injecting out-of-distribution (OOD) data.

6. Discussion and Research Directions

This survey presented an integrated perspective on multimodal classification under uncertainty, highlighting how design choices in representation learning, fusion strategies, and training objectives shape the emergence of uncertainty. We showed that uncertainty quantification is not merely an auxiliary component, but a central element for building reliable multimodal systems, particularly in the presence of noisy, incomplete, or conflicting modalities. However, current multimodal pipelines still tend to treat fusion, uncertainty estimation, and decision-making as loosely coupled stages. Fusion mechanisms often fail to account for partial dependence between modalities; uncertainty quantification methods frequently rely on restrictive independence assumptions or conflict-prone training settings; and decision layers such as set-valued classification are commonly implemented as post-hoc components rather than being tightly integrated with uncertainty-aware fusion. Crucially, the value of uncertainty quantification lies not only in estimating uncertainty, but also in enabling more informed and reliable decision-making processes.

Table 3: Cross-reference: structural limitations identified in Section 2 and the multimodal UQ methods in Section 4 that directly address them. This table illustrates the coherent progression from fusion architecture to uncertainty-aware design.

§2 Limitation	§4 Addressing Method
Equal-reliability fusion assumption	Uncertainty-aware late fusion (DS, SL); sample-specific discounting [12]
No per-modality confidence signal	EDL-based modality networks (TMC [52], ECML [168])
Modality imbalance / dominant modality	Trust-based discounting [100]; conflict-penalizing loss [168]
No inter-modal conflict detection	DS combination with conflict management [107; 59]; sample-specific discounting [12]
Overconfident joint prediction	SVC decision layer translating uncertainty into prediction sets (§3.3)

Beyond uncertainty estimation, we emphasized the importance of decision-level strategies such as set-valued classification, which translate uncertainty into actionable predictions. This is particularly relevant in multimodal settings, where disagreement between modalities naturally leads to ambiguity that cannot be adequately captured by single-label predictions.

Taken together, these observations highlight the need for a unified view that connects multimodal design, uncertainty estimation, and decision-making, with fusion mechanisms playing a central role in propagating and resolving uncertainty across modalities. Moving forward, key research directions include the development of scalable and conflict-aware uncertainty quantification methods, the establishment of standardized evaluation protocols, and tighter integration between uncertainty modeling and downstream decision-making processes. These advances are essential for deploying trustworthy multimodal AI systems in real-world applications.

6.1 Axis I: Modeling Inter-Modal Dependence for Reliable Fusion

A key open challenge in multimodal uncertainty quantification is moving beyond binary assumptions of full independence or full dependence between modalities. As discussed in Section *4.1.2*, subjective logic provides different fusion operators for independent and dependent opinions. However, real-world multimodal data typically exhibit varying degrees of partial dependence that cannot be adequately captured by these extreme assumptions.

For instance, Cumulative Belief Fusion (CBF) assumes inde-

[2]https://archive.ics.uci.edu/ml/datasets/Multiple+Features
[3]http://www.vision.caltech.edu/visipedia/CUB-200.html
[4]https://serre-lab.clps.brown.edu/resource/hmdb-a-largehuman-motion-database
[5]https://data.caltech.edu/records/mzrjq-6wc02

pendence and aggregates evidences additively, while Averaging Belief Fusion (ABF) assumes dependence and averages them as if redundant. In practice, modalities may share some information while also contributing distinctive signals. A promising direction is therefore to model inter-modal dependence in a continuous manner rather than as a binary choice.

One potential approach is disentangled information modeling [86; 157], where modality-specific opinions are decomposed into shared and modality-unique components. By separating overlapping and distinctive information before fusion, it becomes possible to apply more defensible independence assumptions at the aggregation stage. Such disentangled fusion mechanisms could lead to more accurate uncertainty estimates by explicitly accounting for information redundancy and partial dependence.

6.2 Axis II: Systematic Evaluation of Multimodal Uncertainty

Another major limitation identified in our review is the lack of standardized and controlled benchmarks for multimodal uncertainty quantification. Without systematic evaluation protocols, it remains difficult to compare methods fairly or to assess their robustness under varying levels of conflict, noise, or modality misalignment.

Future efforts should extend datasets such as LUMA by incorporating additional modalities with controlled interdependencies. For example, adding automatically generated textual descriptions of images would introduce causal relationships between modalities and allow controlled study of uncertainty propagation across correlated inputs.

In parallel, developing standalone toolkits capable of injecting controlled perturbations—such as noise, misalignment, modality dropout, and out-of-distribution samples—into existing multimodal datasets would enable reproducible and systematic evaluation of multimodal UQ methods. Such tools would facilitate rigorous comparison across approaches and promote standardized evaluation protocols for conflict-aware learning.

6.3 Axis III: From Multimodal Classification to Agentic Systems

The reliability challenges discussed in this survey naturally extend beyond classification to emerging agentic AI systems, where multimodal models are entrusted with autonomous decision-making [167]. Large language models, often forming the reasoning backbone of agents, are known to produce hallucinations and confidently generate incorrect outputs [60]. In addition, agents integrate information from external sources such as web search results, APIs, or knowledge bases, which may themselves be noisy or unreliable. Multimodal inputs—including text, images, audio, and structured data—can also contain internal conflicts or redundancies.

Ensuring reliability in such systems requires mechanisms for quantifying uncertainty and resolving conflicts both within multimodal inputs and across interacting agents [159; 36]. Uncertainty quantification in large language models [132], as well as conflict resolution in multi-agent systems [149], are active research areas. Recent discussions [77] suggest that new conceptual frameworks may be required to adequately capture the unique uncertainty sources of large-scale agentic systems.

Future research should explore how uncertainty can become an explicit component of agent reasoning rather than a residual by-product. Agents should be able to assess the trustworthiness of external sources [183; 149], reconcile conflicting information streams, and adapt their decisions accordingly. Ultimately, advancing conflict-aware multimodal fusion and uncertainty-driven decision strategies will be crucial for building transparent and reliable multimodal agents capable of reasoning about their own limitations.

Overall, advancing multimodal reliability requires integrating representation learning, uncertainty quantification, and decision strategies within a unified framework that explicitly models dependence, conflict, and uncertainty propagation across heterogeneous information sources.

7. REFERENCES

[1] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya, et al. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information fusion*, 76:243–297, 2021.

[2] M. Abdar, S. Salari, S. Qahremani, H.-K. Lam, F. Karray, S. Hussain, A. Khosravi, U. R. Acharya, V. Makarenkov, and S. Nahavandi. Uncertainty-fusenet: robust uncertainty-aware hierarchical feature fusion model with ensemble monte carlo dropout for covid-19 detection. *Information Fusion*, 90:364–381, 2023.

[3] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

[4] G. Andrew, R. Arora, J. Bilmes, and K. Livescu. Deep canonical correlation analysis. In *International conference on machine learning*, pages 1247–1255. PMLR, 2013.

[5] P. K. Atrey, M. A. Hossain, A. El Saddik, and M. S. Kankanhalli. Multimodal fusion for multimedia analysis: a survey. *Multimedia systems*, 16:345–379, 2010.

[6] N. Audebert, C. Herold, K. Slimani, and C. Vidal. Multimodal deep networks for text and image-based document classification. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 427–443. Springer, 2019.

[7] M. S. Ayhan and P. Berens. Test-time data augmentation for estimation of heteroscedastic aleatoric uncertainty in deep neural networks. In *Medical Imaging with Deep Learning*, 2018.

[8] T. Baltrušaitis, C. Ahuja, and L.-P. Morency. Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2):423–443, 2018.

[9] O. Basir, F. Karray, and H. Zhu. Connectionist-based dempster-shafer evidential reasoning for data fusion. *IEEE Transactions on Neural Networks*, 16(6):1513–1530, 2005.

[10] S. Behpour, T. L. Doan, X. Li, W. He, L. Gou, and L. Ren. Gradorth: A simple yet efficient out-of-distribution detection with orthogonal projection of gradients. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[11] G. Bezirganyan, S. Sellami, L. Berti-Équille, and S. Fournier. M2-mixer: A multimodal mixer with multi-head loss for classification from multimodal data. In *2023 IEEE International Conference on Big Data (BigData)*, pages 1052–1058. IEEE, 2023.

[12] G. Bezirganyan, S. Sellami, L. Berti-Équille, and S. Fournier. Multimodal learning with uncertainty quantification based on discounted belief fusion. In *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 258 of *Proceedings of Machine Learning Research*, pages 3142–3150. PMLR, 2025.

[13] G. Bezirganyan, S. Sellami, L. Berti-Équille, and S. Fournier. Luma: A benchmark dataset for learning from uncertain and multimodal data. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2025.

[14] I. Bloch. Some aspects of dempster-shafer evidence theory for classification of multi-modality medical images taking partial volume effect into account. *Pattern Recognition Letters*, 17(8):905–919, 1996.

[15] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra. Weight uncertainty in neural network. In *International conference on machine learning*, pages 1613–1622. PMLR, 2015.

[16] L. Chen, J. Wang, T. Mortlock, P. Khargonekar, and M. A. Al Faruque. Hyperdimensional uncertainty quantification for multimodal uncertainty fusion in autonomous vehicles perception. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22306–22316, 2025.

[17] Y.-T. Chen, J. Shi, Z. Ye, C. Mertz, D. Ramanan, and S. Kong. Multimodal object detection via probabilistic ensembling. In *European Conference on Computer Vision*, pages 139–158. Springer, 2022.

[18] G. Cherubin, K. Chatzikokolakis, and M. Jaggi. Exact optimization of conformal predictors via incremental and decremental learning. In *International Conference on Machine Learning*, pages 1836–1845. PMLR, 2021.

[19] E. Chzhen, C. Denis, M. Hebiri, and T. Lorieul. Set-valued classification–overview via a unified framework. *arXiv preprint arXiv:2102.12318*, 2021.

[20] G. Corani and M. Zaffalon. Learning reliable classifiers from small or incomplete data sets: The naive credal classifier 2. *J. Mach. Learn. Res.*, 9:581–621, 2008.

[21] J. C. Cresswell, Y. Sui, B. Kumar, and N. Vouitsis. Conformal prediction sets improve human decision making. In *International Conference on Machine Learning*, pages 9439–9457. PMLR, 2024.

[22] A. Damianou. *Deep Gaussian processes and variational propagation of uncertainty*. PhD thesis, University of Sheffield, 2015.

[23] J. J. del Coz, J. Díez, and A. Bahamonde. Learning nondeterministic classifiers. *J. Mach. Learn. Res.*, 10:2273–2293, 2009.

[24] A. P. Dempster. A generalization of bayesian inference. *Journal of the Royal Statistical Society: Series B (Methodological)*, 30(2):205–232, 1968.

[25] C. Denis and M. Hebiri. Confidence sets with expected sizes for multiclass classification. *Journal of Machine Learning Research*, 18(102):1–28, 2017.

[26] T. Denoeux. A neural network classifier based on dempster-shafer theory. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 30(2):131–150, 2000.

[27] L. Deregnaucourt, A. Lechervy, H. Laghmara, and S. Ainouz. An evidential deep network based on dempster-shafer theory for large dataset. *Advances and Applications of DSmT for Information Fusion*, page 907, 2023.

[28] T. Diethe, D. R. Hardoon, and J. Shawe-Taylor. Constructing nonlinear discriminants from multiple data views. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2010, Barcelona, Spain, September 20-24, 2010, Proceedings, Part I 21*, pages 328–343. Springer, 2010.

[29] X. Dong, Y. Yan, M. Tan, Y. Yang, and I. W. Tsang. Late fusion via subspace search with consistency preservation. *IEEE Transactions on Image Processing*, 28(1):518–528, 2018.

[30] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

[31] C. Du, J. Teng, T. Li, Y. Liu, T. Yuan, Y. Wang, Y. Yuan, and H. Zhao. On uni-modal feature learning in supervised multi-modal learning. In *International Conference on Machine Learning*, pages 8632–8656. PMLR, 2023.

[32] P. Duchnowski, U. Meier, and A. Waibel. See me, hear me: integrating automatic speech recognition and lip-reading. In *ICSLP*, volume 94, pages 547–550. Citeseer, 1994.

[33] D. M. El-Din, A. E. Hassanein, and E. E. Hassanien. An adaptive and late multifusion framework in contextual representation based on evidential deep learning and dempster–shafer theory. *Knowledge and Information Systems*, 66(11):6881–6932, 2024.

[34] B. B. Elallid, N. Benamar, A. S. Hafid, T. Rachidi, and N. Mrani. A comprehensive survey on the application of deep and reinforcement learning approaches in autonomous driving. *Journal of King Saud University-Computer and Information Sciences*, 34(9):7366–7390, 2022.

[35] Z. Elouedi, K. Mellouli, and P. Smets. The evaluation of sensors' reliability and their tuning for multisensor data fusion within the transferable belief model. In *European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, pages 350–361. Springer, 2001.

[36] E. Fadeeva, A. Rubashevskii, R. Vashurin, S. Dhuliawala, A. Shelmanov, T. Baldwin, P. Nakov, M. Sachan, and M. Panov. Faithfulness-aware uncertainty quantification for fact-checking the output of retrieval augmented generation. *arXiv preprint arXiv:2505.21072*, 2025.

[37] A. Foong, W. Bruinsma, J. Gordon, Y. Dubois, J. Requeima, and R. Turner. Meta-learning stationary stochastic process prediction with convolutional neural processes. *Advances in Neural Information Processing Systems*, 33:8284–8295, 2020.

[38] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov. Devise: A deep visual-semantic embedding model. *Advances in neural information processing systems*, 26, 2013.

[39] Y. Gal and Z. Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

[40] M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, and P. N. Suganthan. Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence*, 115:105151, 2022.

[41] Z. Gao, X. Jiang, X. Xu, F. Shen, Y. Li, and H. T. Shen. Embracing unimodal aleatoric uncertainty for robust multimodal fusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26876–26885, 2024.

[42] C. Garcin, M. Servajean, A. Joly, and J. Salmon. A two-head loss function for deep average-k classification. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 7358–7367. IEEE, 2025.

[43] M. Garnelo, D. Rosenbaum, C. Maddison, T. Ramalho, D. Saxton, M. Shanahan, Y. W. Teh, D. Rezende, and S. A. Eslami. Conditional neural processes. In *International conference on machine learning*, pages 1704–1713. PMLR, 2018.

[44] M. Garnelo, J. Schwarz, D. Rosenbaum, F. Viola, D. J. Rezende, S. Eslami, and Y. W. Teh. Neural processes. *arXiv preprint arXiv:1807.01622*, 2018.

[45] J. Gawlikowski, C. R. N. Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher, et al. A survey of uncertainty in deep neural networks. *Artificial Intelligence Review*, pages 1–77, 2023.

[46] R. Girdhar, A. El-Nouby, Z. Liu, M. Singh, K. V. Alwala, A. Joulin, and I. Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15180–15190, 2023.

[47] Y. Gong, Y. Chung, and J. R. Glass. AST: audio spectrogram transformer. In *22nd Annual Conference of the International Speech Communication Association, Interspeech 2021, Brno, Czechia, August 30 - September 3, 2021*, pages 571–575. ISCA, 2021.

[48] C. Gruber, P. O. Schenk, M. Schierholz, F. Kreuter, and G. Kauermann. Sources of uncertainty in machine learning–a statisticians' view. *arXiv preprint arXiv:2305.16703*, 2023.

[49] V. Guarrasi, F. Aksu, C. M. Caruso, F. Di Feola, A. Rofena, F. Ruffini, and P. Soda. A systematic review of intermediate fusion in multimodal deep learning for biomedical applications. *Image and Vision Computing*, page 105509, 2025.

[50] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.

[51] H. Guo, W. Shi, and Y. Deng. Evaluating sensor reliability in classification problems based on evidence theory. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 36(5):970–981, 2006.

[52] Z. Han, C. Zhang, H. Fu, and J. T. Zhou. Trusted multi-view classification. *arXiv preprint arXiv:2102.02051*, 2021.

[53] Z. S. Harris. Distributional structure. *Word*, 10(2-3):146–162, 1954.

[54] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[55] W. He, Z. Jiang, T. Xiao, Z. Xu, and Y. Li. A survey on uncertainty quantification methods for deep learning. *arXiv preprint arXiv:2302.13425*, 2023.

[56] A. Hoarau, B. Quost, S. Destercke, and W. Waegeman. Reducing aleatoric and epistemic uncertainty through multi-modal data acquisition. *arXiv preprint arXiv:2501.18268*, 2025.

[57] H. Hotelling. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377, 1936.

[58] G. Huang, Y. Li, G. Pleiss, Z. Liu, J. E. Hopcroft, and K. Q. Weinberger. Snapshot ensembles: Train 1, get m for free. *arXiv preprint arXiv:1704.00109*, 2017.

[59] L. Huang, S. Ruan, P. Decazes, and T. Denœux. Deep evidential fusion with uncertainty quantification and reliability learning for multimodal medical image segmentation. *Information Fusion*, 113:102648, 2025.

[60] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, 2025.

[61] R. Huang, A. Geng, and Y. Li. On the importance of gradients for detecting distributional shifts in the wild. In M. Ranzato, A. Beygelzimer, Y. N. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 677–689, 2021.

[62] Y. S. Huang, K. Liu, and C. Y. Suen. The combination of multiple classifiers by a neural network approach. *International Journal of Pattern Recognition and Artificial Intelligence*, 9(03):579–597, 1995.

[63] Y.-H. Hubert Tsai, L.-K. Huang, and R. Salakhutdinov. Learning robust visual-semantic embeddings. In *Proceedings of the IEEE International conference on Computer Vision*, pages 3571–3580, 2017.

[64] E. Hüllermeier and W. Waegeman. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine learning*, 110(3):457–506, 2021.

[65] V. Jayachitra, S. Nivetha, R. Nivetha, and R. Harini. A cognitive iot-based framework for effective diagnosis of covid-19 using multimodal data. *Biomedical Signal Processing and Control*, 70:102960, 2021.

[66] C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. Le, Y.-H. Sung, Z. Li, and T. Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021.

[67] J. Jiao, H. Sun, Y. Huang, M. Xia, M. Qiao, Y. Ren, Y. Wang, and Y. Guo. Gmrlnet: A graph-based manifold regularization learning framework for placental insufficiency diagnosis on incomplete multimodal ultrasound data. *IEEE Transactions on Medical Imaging*, 42(11):3205–3218, 2023.

[68] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine learning*, 37:183–233, 1999.

[69] A. Jøsang. *Subjective logic*, volume 3. Springer, 2016.

[70] A. Jøsang, D. Wang, and J. Zhang. Multi-source fusion in subjective logic. In *2017 20th International Conference on Information Fusion (Fusion)*, pages 1–8, 2017.

[71] M. C. Jung, H. Zhao, J. Dipnall, and L. Du. Beyond unimodal: Generalising neural processes for multimodal uncertainty estimation. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 42191–42216. Curran Associates, Inc., 2023.

[72] M. C. Jung, H. Zhao, J. Dipnall, and L. Du. Beyond unimodal: Generalising neural processes for multimodal uncertainty estimation. *Advances in Neural Information Processing Systems*, 36, 2024.

[73] M. C. Jung, H. Zhao, J. Dipnall, B. Gabbe, and L. Du. Uncertainty estimation for multi-view data: The power of seeing the whole picture. *Advances in Neural Information Processing Systems*, 35:6517–6530, 2022.

[74] A. I. Károly, P. Galambos, J. Kuti, and I. J. Rudas. Deep learning in robotics: Survey on model structures and training strategies. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 51(1):266–279, 2020.

[75] H. Kim, A. Mnih, J. Schwarz, M. Garnelo, S. M. A. Eslami, D. Rosenbaum, O. Vinyals, and Y. W. Teh. Attentive neural processes. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[76] T.-K. Kim, J. Kittler, and R. Cipolla. Learning discriminative canonical correlations for object recognition with image sets. In *Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006, Proceedings, Part III 9*, pages 251–262. Springer, 2006.

[77] M. Kirchhof, G. Kasneci, and E. Kasneci. Position: Uncertainty quantification needs reassessment for large language model agents. In *Forty-second International Conference on Machine Learning Position Paper Track*, 2025.

[78] J. Kittler. Combining classifiers: A theoretical framework. *Pattern analysis and Applications*, 1:18–27, 1998.

[79] A. D. Kiureghian and O. Ditlevsen. Aleatory or epistemic? Does it matter? *Structural Safety*, 31(2):105–112, March 2009.

[80] N. Kotelevskii, M. Guizani, É. Moulines, and M. Panov. Adaptive temperature scaling with conformal prediction. 2025.

[81] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[82] A. Kumar and B. Raj. Unsupervised fusion weight learning in multiple classifier systems. *arXiv preprint arXiv:1502.01823*, 2015.

[83] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

[84] J. Lampinen and A. Vehtari. Bayesian approach for neural networks—review and case studies. *Neural networks*, 14(3):257–274, 2001.

[85] J. Lee and G. AlRegib. Gradients as a measure of uncertainty in neural networks. In *IEEE International Conference on Image Processing, ICIP 2020, Abu Dhabi, United Arab Emirates, October 25-28, 2020*, pages 2416–2420. IEEE, 2020.

[86] M. Lee and V. Pavlovic. Private-shared disentangled multimodal vae for learning of hybrid latent representations. *arXiv preprint arXiv:2012.13024*, 2020.

[87] S. Lee, S. Purushwalkam, M. Cogswell, D. Crandall, and D. Batra. Why m heads are better than one: Training a diverse ensemble of deep networks. *arXiv preprint arXiv:1511.06314*, 2015.

[88] C. Li, K. Li, Y. Ou, L. M. Kaplan, A. Jøsang, J.-H. Cho, D. H. JEONG, and F. Chen. Hyper evidential deep learning to quantify composite classification uncertainty. In *The Twelfth International Conference on Learning Representations*, 2024.

[89] J. Li, S. Luo, and J. S. Jin. Sensor data fusion for accurate cloud presence prediction using dempster-shafer evidence theory. *Sensors*, 10(10):9384–9396, 2010.

[90] L. Li, C. Li, X. Lu, H. Wang, and D. Zhou. Multi-focus image fusion with convolutional neural network based on dempster-shafer theory. *Optik*, 272:170223, 2023.

[91] L. H. Li, M. Yatskar, D. Yin, C.-J. Hsieh, and K.-W. Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019.

[92] S. Li and H. Tang. Multimodal alignment and fusion: A survey. *arXiv preprint arXiv:2411.17040*, 2024.

[93] Y. Li, M. Yang, and Z. Zhang. A survey of multi-view representation learning. *IEEE transactions on knowledge and data engineering*, 31(10):1863–1883, 2018.

[94] T. Liu, J. Huang, T. Liao, R. Pu, S. Liu, and Y. Peng. A hybrid deep learning model for predicting molecular subtypes of human breast cancer using multimodal data. *Irbm*, 43(1):62–74, 2022.

[95] W. Liu, Y. Chen, X. Yue, C. Zhang, and S. Xie. Safe multi-view deep classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 8870–8878, 2023.

[96] W. Liu, X. Yue, Y. Chen, and T. Denoeux. Trusted Multi-View Deep Learning with Opinion Aggregation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(7):7585–7593, June 2022.

[97] Y. Liu, L. Liu, C. Xu, X. Song, Z. Guan, and W. Zhao. Dynamic evidence decoupling for trusted multi-view learning. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 7269–7277, 2024.

[98] D. G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.

[99] J. Lu, D. Batra, D. Parikh, and S. Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32, 2019.

[100] J. Lu, W. Buntine, Y. Qi, J. Dipnall, B. Gabbe, and L. Du. Navigating conflicting views: Harnessing trust for learning. *arXiv preprint arXiv:2406.00958*, 2024.

[101] A. Lyzhov, Y. Molchanova, A. Ashukha, D. Molchanov, and D. Vetrov. Greedy policy search: A simple baseline for learnable test-time augmentation. In *Conference on uncertainty in artificial intelligence*, pages 1308–1317. PMLR, 2020.

[102] L. Ma and T. Denoeux. Partial classification in the belief function framework. *Knowledge-Based Systems*, 214:106742, 2021.

[103] K. Maag and T. Riedlinger. Pixel-wise gradient uncertainty for convolutional neural networks applied to out-of-distribution segmentation. In *VISIGRAPP (2): VISAPP*, pages 112–122, 2024.

[104] A. Malinin and M. Gales. Predictive uncertainty estimation via prior networks. *Advances in neural information processing systems*, 31, 2018.

[105] A. Malinin and M. J. F. Gales. Reverse kl-divergence training of prior networks: Improved uncertainty and adversarial robustness. In *Neural Information Processing Systems*, 2019.

[106] A. Malinin, B. Mlodozeniec, and M. Gales. Ensemble distribution distillation. *arXiv preprint arXiv:1905.00076*, 2019.

[107] A. Martin. Conflict management in information fusion with belief functions. In *Information quality in information fusion and decision making*, pages 79–97. Springer, 2019.

[108] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[109] T. Mortier, M. Wydmuch, K. Dembczyński, E. Hüllermeier, and W. Waegeman. Set-valued prediction in multi-class classification. In *31st Benelux conference on Artificial Intelligence (BNAIC 2019); 28th Belgian Dutch conference on Machine Learning (Benelearn 2019)*, volume 2491. CEUR, 2019.

[110] B. Mucsányi, M. Kirchhof, and S. J. Oh. Benchmarking uncertainty disentanglement: Specialized uncertainties for specialized tasks. *Advances in Neural Information Processing Systems*, 37:50972–51038, 2024.

[111] J. Nandy, W. Hsu, and M. Lee. Towards maximizing the representation gap between in-domain & out-of-distribution examples. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[112] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, A. Y. Ng, et al. Multimodal deep learning. In *ICML*, volume 11, pages 689–696, 2011.

[113] K. Nguyen, S. Denman, S. Sridharan, and C. Fookes. Score-level multibiometric fusion based on dempster–shafer theory incorporating uncertainty factors. *IEEE Transactions on Human-Machine Systems*, 45(1):132–140, 2014.

[114] V.-L. Nguyen, H. Zhang, and S. Destercke. Learning sets of probabilities through ensemble methods. In *European Conference on Symbolic and Quantitative Approaches with Uncertainty*, pages 270–283. Springer, 2023.

[115] A. M. Ozbayoglu, M. U. Gudelek, and O. B. Sezer. Deep learning for financial applications: A survey. *Applied soft computing*, 93:106384, 2020.

[116] Y. Pan, H. Lai, C. Liu, and S. Yan. A divide-and-conquer method for scalable low-rank latent matrix pursuit. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 524–531, 2013.

[117] F. Parisi, F. Strino, B. Nadler, and Y. Kluger. Ranking and combining multiple predictors without labeled data. *Proceedings of the National Academy of Sciences*, 111(4):1253–1258, 2014.

[118] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[119] B. Quost, M.-H. Masson, and T. Denœux. Classifier fusion in the dempster–shafer framework using optimized t-norm based combination rules. *International Journal of Approximate Reasoning*, 52(3):353–374, 2011.

[120] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[121] D. Ramachandram and G. W. Taylor. Deep multimodal learning: A survey on recent advances and trends. *IEEE signal processing magazine*, 34(6):96–108, 2017.

[122] I. Reda, A. Khalil, M. Elmogy, A. Abou El-Fetouh, A. Shalaby, M. Abou El-Ghar, A. Elmaghraby, M. Ghazal, and A. El-Baz. Deep learning role in early diagnosis of prostate cancer. *Technology in cancer research & treatment*, 17:1533034618775530, 2018.

[123] Y. Sale, V. Bengs, M. Caprio, and E. Hüllermeier. Second-order uncertainty quantification: A distance-based approach. In *International Conference on Machine Learning*, pages 43060–43076. PMLR, 2024.

[124] K. Schweighofer, L. Aichberger, M. Ielanskyi, and S. Hochreiter. Introducing an improved information-theoretic measure of predictive uncertainty. In *NeurIPS 2023 workshop: Information-Theoretic Principles in Cognitive Systems*.

[125] M. Seeger. Gaussian processes for machine learning. *International journal of neural systems*, 14(02):69–106, 2004.

[126] M. Sensoy, L. Kaplan, and M. Kandemir. Evidential deep learning to quantify classification uncertainty. *Advances in neural information processing systems*, 31, 2018.

[127] G. Shafer. *A Mathematical Theory of Evidence*. Princeton university press, 1976.

[128] G. Shafer and V. Vovk. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(3), 2008.

[129] S. Shamshirband, M. Fathi, A. Dehzangi, A. T. Chronopoulos, and H. Alinejad-Rokny. A review on deep learning approaches in healthcare systems: Taxonomies, challenges, and open issues. *Journal of Biomedical Informatics*, 113:103627, 2021.

[130] Z. Shao, W. Dou, and Y. Pan. Dual-level deep evidential fusion: Integrating multimodal information for enhanced reliable decision-making in deep learning. *Information Fusion*, 103:102113, 2024.

[131] L. Shi, C. Tang, H. Deng, C. Xu, L. Xing, and B. Chen. Generalized trusted multi-view classification framework with hierarchical opinion aggregation. *arXiv preprint arXiv:2411.03713*, 2024.

[132] O. Shorinwa, Z. Mei, J. Lidard, A. Z. Ren, and A. Majumdar. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *ACM Computing Surveys*, 2025.

[133] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. *Advances in neural information processing systems*, 27, 2014.

[134] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[135] W. C. Sleeman IV, R. Kapoor, and P. Ghosh. Multimodal classification: Current landscape, taxonomy and future directions. *ACM Computing Surveys*, 55(7):1–31, 2022.

[136] E. Snelson and Z. Ghahramani. Sparse gaussian processes using pseudo-inputs. *Advances in neural information processing systems*, 18, 2005.

[137] N. Srivastava and R. R. Salakhutdinov. Multimodal learning with deep boltzmann machines. *Advances in neural information processing systems*, 25, 2012.

[138] S. R. Stahlschmidt, B. Ulfenborg, and J. Synnergren. Multimodal deep learning for biomedical data fusion: a review. *Briefings in bioinformatics*, 23(2):bbab569, 2022.

[139] D. Sun, M. Wang, and A. Li. A multimodal deep neural network for human breast cancer prognosis prediction by integrating multi-dimensional data. *IEEE/ACM transactions on computational biology and bioinformatics*, 16(3):841–850, 2018.

[140] S. Sun. A survey of multi-view machine learning. *Neural computing and applications*, 23:2031–2038, 2013.

[141] H. Tan and M. Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5100–5111, 2019.

[142] M. K. Tellamekala, S. Amiriparian, B. W. Schuller, E. André, T. Giesbrecht, and M. Valstar. Cold fusion: Calibrated and ordinal latent distribution fusion for uncertainty-aware multimodal emotion recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(2):805–822, 2023.

[143] J. Thomason, D. Gordon, and Y. Bisk. Shifting the baseline: Single modality performance on visual navigation & qa. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1977–1983, 2019.

[144] J. Tian, W. Cheung, N. Glaser, Y.-C. Liu, and Z. Kira. Uno: Uncertainty-aware noisy-or multimodal fusion for unanticipated input degradation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5716–5723. IEEE, 2020.

[145] M. Titsias. Variational learning of inducing variables in sparse gaussian processes. In *Artificial intelligence and statistics*, pages 567–574. PMLR, 2009.

[146] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in neural information processing systems*, 34:24261–24272, 2021.

[147] Z. Tong, P. Xu, and T. Denoeux. An evidential classifier based on dempster-shafer theory and deep learning. *Neurocomputing*, 450:275–293, 2021.

[148] Z. Tong, P. Xu, and T. Denœux. Fusion of evidential cnn classifiers for image classification. In *International Conference on Belief Functions*, pages 168–176. Springer, 2021.

[149] K.-T. Tran, D. Dao, M.-D. Nguyen, Q.-V. Pham, B. O'Sullivan, and H. D. Nguyen. Multi-agent collaboration mechanisms: A survey of llms. *arXiv preprint arXiv:2501.06322*, 2025.

[150] M. C. Troffaes. Decision making under uncertainty using imprecise probabilities. *International journal of approximate reasoning*, 45(1):17–29, 2007.

[151] T. Tsiligkaridis. Information aware max-norm dirichlet networks for predictive uncertainty estimation. *Neural Networks*, 135:105–114, 2021.

[152] J. van Hout, E. Yeh, D. C. Koelma, C. G. Snoek, C. Sun, R. Nevatia, J. Wong, and G. K. Myers. Late fusion and calibration for multimedia event detection using few examples. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4598–4602. IEEE, 2014.

[153] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[154] F. Verdoja and V. Kyrki. Notes on the behavior of mc dropout. In *ICML Workshop on Uncertainty & Robustness in Deep Learning*, 2021.

[155] V. Vovk, A. Gammerman, and G. Shafer. *Algorithmic learning in a random world*, volume 29. Springer, 2005.

[156] P. Walley. Statistical reasoning with imprecise probabilities. 1991.

[157] C. Wang, S. Gupta, X. Zhang, S. Tonekaboni, S. Jegelka, T. S. Jaakkola, and C. Uhler. An information criterion for controlled disentanglement of multimodal data. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

[158] G. Wang, W. Li, S. Ourselin, and T. Vercauteren. Automatic brain tumor segmentation using convolutional neural networks with test-time augmentation. In *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries - 4th International Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part II*, volume 11384 of *Lecture Notes in Computer Science*, pages 61–72. Springer, 2018.

[159] H. Wang, A. Prasad, E. Stengel-Eskin, and M. Bansal. Retrieval-augmented generation with conflicting evidence. *arXiv preprint arXiv:2504.13079*, 2025.

[160] H. Wang, V. Subramanian, and T. Syeda-Mahmood. Modeling uncertainty in multi-modal fusion for lung cancer survival analysis. In *2021 IEEE 18th international symposium on biomedical imaging (ISBI)*, pages 1169–1172. IEEE, 2021.

[161] H. Wang, J. Zhang, Y. Chen, C. Ma, J. Avery, L. Hull, and G. Carneiro. Uncertainty-aware multimodal learning via cross-modal random network prediction. In *European Conference on Computer Vision*, pages 200–217. Springer, 2022.

[162] W. Wang, D. Tran, and M. Feiszli. What makes training multi-modal classification networks hard? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12695–12705, 2020.

[163] Z. Wang and X. Qiao. Set-valued classification with out-of-distribution detection for many classes. *Journal of Machine Learning Research*, 24(375):1–39, 2023.

[164] A. G. Wilson, Z. Hu, R. Salakhutdinov, and E. P. Xing. Deep kernel learning. In *Artificial intelligence and statistics*, pages 370–378. PMLR, 2016.

[165] L. Wimmer, Y. Sale, P. Hofman, B. Bischl, and E. Hüllermeier. Quantifying aleatoric and epistemic uncertainty in machine learning: Are conditional entropy and mutual information appropriate measures? In *Uncertainty in artificial intelligence*, pages 2282–2292. PMLR, 2023.

[166] N. Wu, S. Jastrzebski, K. Cho, and K. J. Geras. Characterizing and overcoming the greedy nature of learning in multi-modal deep neural networks. In *International Conference on Machine Learning*, pages 24043–24055. PMLR, 2022.

[167] J. Xie, Z. Chen, R. Zhang, X. Wan, and G. Li. Large multimodal agents: A survey. *arXiv preprint arXiv:2402.15116*, 2024.

[168] C. Xu, J. Si, Z. Guan, W. Zhao, Y. Wu, and X. Gao. Reliable conflictive multi-view learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 16129–16137, 2024.

[169] C. Xu, Y. Zhang, Z. Guan, and W. Zhao. Trusted multi-view learning with label noise. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024, Jeju, South Korea, August 3-9, 2024*, pages 5263–5271. ijcai.org, 2024.

[170] Z. Xue and R. Marculescu. Dynamic multimodal fusion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023 - Workshops, Vancouver, BC, Canada, June 17-24, 2023*, pages 2575–2584. IEEE, 2023.

[171] X. Yan, S. Hu, Y. Mao, Y. Ye, and H. Yu. Deep multi-view learning methods: A review. *Neurocomputing*, 448:106–129, 2021.

[172] G. Yang, S. Destercke, and M.-H. Masson. Cautious classification with nested dichotomies and imprecise probabilities. *Soft Computing*, 21:7447–7462, 2017.

[173] Q. Yang, Y. Zhao, and H. Cheng. Mmlf: Multi-modal multi-class late fusion for object detection with uncertainty estimation. *arXiv preprint arXiv:2410.08739*, 2024.

[174] Y. Yang, F. Wan, Q.-Y. Jiang, and Y. Xu. Facilitating multimodal classification via dynamically learning modality gap. *Advances in Neural Information Processing Systems*, 37:62108–62122, 2024.

[175] G. Ye, D. Liu, I.-H. Jhuo, and S.-F. Chang. Robust late fusion with rank minimization. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3021–3028. IEEE, 2012.

[176] L. Zadeh. A mathematical theory of evidence (book review). *AI magazine*, 5:81–83, 1984.

[177] M. Zaffalon. The naive credal classifier. *Journal of statistical planning and inference*, 105(1):5–21, 2002.

[178] M. Zaffalon, G. Corani, and D. Mauá. Evaluating credal classifiers by utility-discounted predictive accuracy. *International Journal of Approximate Reasoning*, 53(8):1282–1301, 2012. Imprecise Probability: Theories and Applications (ISIPTA'11).

[179] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

[180] J. Zhao, X. Xie, X. Xu, and S. Sun. Multi-view learning overview: Recent progress and new challenges. *Information Fusion*, 38:43–54, 2017.

[181] X. Zhao, Y. Ou, L. M. Kaplan, F. Chen, and J. Cho. Quantifying classification uncertainty using regularized evidential neural networks. *CoRR*, abs/1910.06864, 2019.

[182] Z. Zheng, L. Zheng, M. Garrett, Y. Yang, M. Xu, and Y.-D. Shen. Dual-path convolutional image-text embeddings with instance loss. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 16(2):1–23, 2020.

[183] Y. Zhou, Y. Liu, X. Li, J. Jin, H. Qian, Z. Liu, C. Li, Z. Dou, T.-Y. Ho, and P. S. Yu. Trustworthiness in retrieval-augmented generation systems: A survey. *arXiv preprint arXiv:2409.10102*, 2024.

A Survey of Scaling in Large Language Model Reasoning

Zihan Chen[1], Song Wang[1], Zhen Tan[2], Xingbo Fu[1], Zhenyu Lei[1], Peng Wang[1], Huan Liu[2], Cong Shen[1], Jundong Li[1]
[1]University of Virginia, Charlottesville, VA, USA
[2]Arizona State University, Tempe, AZ, USA
{brf3rx, sw3wv, xf3av, vjd5zr, pw7nc, cong, jundong}@virginia.edu
{ztan36, huanliu}@asu.edu

ABSTRACT

The rapid advancements in large Language models (LLMs) have significantly enhanced their reasoning capabilities, driven by various strategies such as multi-agent collaboration. However, unlike the well-established performance improvements achieved through scaling data and model size, the scaling of reasoning in LLMs is more complex and can even negatively impact reasoning performance, introducing new challenges in model alignment and robustness. In this survey, we provide a comprehensive examination of scaling in LLM reasoning, categorizing it into multiple dimensions and analyzing how and to what extent different scaling strategies contribute to improving reasoning capabilities. We begin by exploring scaling in input size, which enables LLMs to process and utilize a more extensive context for improved reasoning. Next, we analyze scaling in reasoning steps that improve multi-step inference and logical consistency. We then examine scaling in reasoning rounds, where iterative interactions refine reasoning outcomes. Furthermore, we discuss scaling in training-enabled reasoning, focusing on optimization through iterative model improvement. Finally, we outline future directions for further advancing LLM reasoning. By synthesizing these diverse perspectives, this survey aims to provide insights into how scaling strategies fundamentally enhance the reasoning capabilities of LLMs and further guide the development of next-generation AI systems.

1. INTRODUCTION

Large Language Models (LLMs) have evolved rapidly, demonstrating remarkable advancements across various natural language processing (NLP) tasks, including text generation, comprehension, and problem-solving [66; 154; 214; 216; 215; 65]. One of the key driving forces behind these improvements is scaling, where increasing the size of training data and model parameters has led to substantial performance gains [86; 69; 193]. Scaling has played a pivotal role in the development of state-of-the-art LLMs such as GPT-4 [134], and Gemini [176], enabling them to generalize across a broad range of tasks with unprecedented accuracy and fluency [184]. The empirical success of scaling laws has reinforced the notion that simply increasing model size and data availability can significantly enhance LLM capabilities [25; 130; 31]. However, while such scaling strategies have led to more powerful models, they do not fully explain improvements in complex reasoning tasks, which require structured thinking and logical consistency [40; 155; 47].

Notably, unlike simpler tasks that rely on memorization or direct retrieval of information, reasoning demands deeper cognitive-like processes, including step-by-step deductions, counterfactual reasoning, and planning [142; 83]. While early LLMs exhibited shallow reasoning abilities [11; 117], recent advancements have introduced techniques aimed at enhancing LLM reasoning performance through various strategies [33; 54; 164]. For instance, s1 [131] explicitly extends the reasoning length, enabling models to engage in deeper, iterative reasoning that can identify and correct errors in previous inference steps. However, scaling reasoning length does not always guarantee improved performance—simply increasing the number of reasoning steps may introduce redundancy, compounding errors, or even diminished accuracy [149; 73; 125; 204; 18; 220]. This highlights the complex and non-trivial nature of scaling in reasoning, necessitating a deeper investigation into how different scaling strategies influence LLM reasoning effectiveness and when they yield diminishing returns. In this survey, we use *reasoning* to refer to tasks in which the model must perform nontrivial transformation over information, such as multi-step deduction or iterative refinement, rather than merely retrieve or fluently generate content. Under this view, not every improvement in general LLM capability should be interpreted as a reasoning improvement. For example, larger context windows, retrieval, or memory may improve performance simply by supplying missing evidence, while their reasoning benefit is most meaningful when that evidence must be integrated in a multi-step decision process. Conversely, these strategies may offer limited gains on tasks dominated by direct factual recall or shallow pattern matching. Throughout this survey, we focus on scaling strategies that strengthen reasoning behavior and highlight the task settings and failure modes under which their gains may diminish.

Several recent surveys have covered adjacent areas, including test-time scaling, general LLM reasoning, and post-training scaling [233; 71; 84]. However, these works typically emphasize a particular stage of scaling, a specific reasoning regime, or a broader architectural view of reasoning systems. More generally, existing surveys often organize reasoning methods by technique families (e.g., RAG, CoT, multi-agent systems, and RL). In contrast, our survey focuses on a different question: how different forms of scaling specifically influence reasoning. We organize the literature by *what form of computation or information is scaled* at inference or training time, which enables a unified comparison of otherwise dis-

Survey	Scope	Organizing lens	Trade-offs	Main emphasis
Zhang et al. [233]	Test-time scaling	*What, how, where, and how well* to scale	Yes	Taxonomy, assessment, and deployment guidance for *test-time* scaling.
Ke et al. [71]	LLM reasoning	*Regimes* and *architectures*, with input/output perspectives	Partial	Broad survey of inference scaling, learning to reason, and agentic systems.
Lai et al. [84]	Post-training scaling	*SFT, RLxF, and TTC* in post-training	Partial	Scaling after pre-training, especially alignment and post-training methodologies.
Ours	Scaling in LLM reasoning	**Input sizes, reasoning steps, reasoning rounds, and training-enabled reasoning**	**Yes**	**Reasoning-centric synthesis across scaling dimensions, with unified comparison of gains, costs, and failure modes.**

Table 1: Comparison with closely related recent surveys. Prior work focuses on test-time scaling, general reasoning regimes/architectures, or post-training scaling, while our survey emphasizes a unified, reasoning-centric view across multiple scaling dimensions.

Dimension	What is scaled	Main benefit	Main cost	Typical failure mode	Best-fit tasks
Input sizes	External information / context / demonstrations	Better grounding and task adaptation	Retrieval and long-context cost	Irrelevant context, distraction, lost-in-the-middle	Knowledge-intensive QA, long-context tasks
Reasoning steps	Intermediate reasoning depth	Better decomposition and verification	Higher token and search cost	Overthinking, compounding errors	Math, logic, planning
Reasoning rounds	Interaction rounds across agents/humans	Critique, diversity, iterative refinement	Coordination and latency overhead	Redundancy, premature consensus, noisy debate	Open-ended reasoning, collaborative decision-making
Model optimization	Internal reasoning via optimization / latent computation	Stronger amortized reasoning	Training compute and data cost	Underthinking, overthinking, and compute-scaling saturation	High-value domains with reusable reasoning improvements

Table 2: Cross-dimensional comparison of scaling strategies for LLM reasoning.

connected lines of work in terms of their reasoning gains, costs, limitations, and characteristic failure modes. Specifically, we categorize reasoning scaling into four dimensions. We first discuss *input scaling*, which expands the external information available to the model through larger contexts, retrieval, demonstrations, or memory. We then examine *reasoning step scaling*, which allocates more intermediate computation to decomposition, verification, and search. Next, we study *reasoning round scaling*, in which LLMs iteratively refine their outputs through interaction, such as multi-agent collaboration, debate, and human-in-the-loop feedback. Finally, we consider *training-enabled reasoning*, which improves reasoning by internalizing stronger reasoning behaviors through optimization. Table 1 compares our survey with prior surveys, and Table 2 summarizes the core trade-offs across these four scaling dimensions, including what is scaled, their main benefits and costs, typical failure modes, and the task settings for which they are best suited.

Table 2 summarizes the core trade-offs across these four scaling dimensions, including what is scaled, their main benefits and costs, typical failure modes, and the task settings for which they are best suited. Although these dimensions are often studied separately, they differ systematically in where computation is allocated, what benefits they provide, and what limitations they introduce. The following sections examine each dimension in detail.

By systematically reviewing the scaling of reasoning in LLMs, this survey aims to bridge the gap between empirical scaling strategies and reasoning improvements. Beyond summarizing representative methods, we seek to clarify when and why scaling improves reasoning, where its returns diminish, and what new challenges it introduces. We hope this survey serves as a useful resource for both researchers and practitioners seeking effective, efficient, and reliable ways to advance LLM reasoning.

2. SCALING IN INPUT SIZES

Scaling input sizes expands the amount of information available to an LLM during reasoning, rather than increasing the depth of reasoning itself. This dimension improves performance by supplying richer contextual evidence, including demonstrations, retrieved documents, and persistent memory, which can enhance grounding, task adaptation, and long-horizon consistency. Its central trade-off is that additional context often improves coverage and robustness, but also increases retrieval overhead, long-context computation,

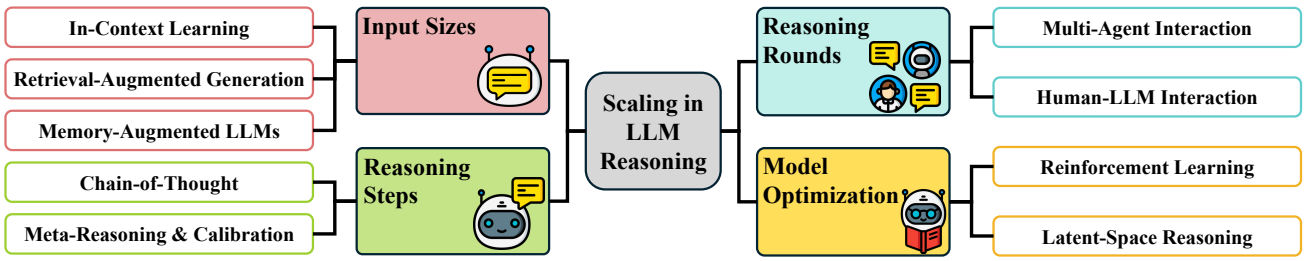

Figure 1: Taxonomy for Scaling in Large Language Model Reasoning.

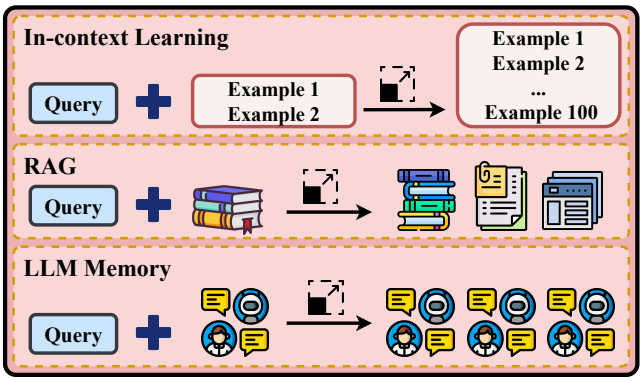

Figure 2: Scaling in LLM input sizes.

and the risk of distraction from irrelevant or poorly organized information. In this section, we examine three major strategies for input scaling—ICL, RAG, and memory-augmented LLMs—and analyze how they improve reasoning performance as well as the bottlenecks they introduce.

2.1 In-Context Learning

In-Context Learning (ICL) enables LLMs to adapt to new tasks without parameter updates by conditioning on demonstrations provided in the input prompt. Various algorithms have been developed to improve ICL performance by optimizing demonstration selection [186; 151; 222; 26], ordering [110; 105], and formatting [179; 109; 77]. More broadly, ICL illustrates a core form of input scaling: increasing the amount of task-relevant context so that the model can better infer the desired behavior from examples alone. Research has shown that model performance often improves as the number of in-context examples increases [1; 126; 11; 117]. However, traditional ICL methods remain constrained by the maximum input context length, which has historically limited them to the few-shot regime [38]. Although some works, such as SAICL [12], modify the attention structure to scale ICL to hundreds of demonstrations [93; 94; 55], they do not fully explore the broader benefits and challenges of operating with substantially larger demonstration sets.

With the expansion of context windows, researchers have increasingly investigated many-shot ICL, in which models leverage hundreds or even thousands of demonstrations [7; 2]. Scaling from few-shot to many-shot ICL has yielded substantial gains across a wide range of generative and discriminative tasks [169; 245; 140]. However, these gains are not unbounded: as the number of in-context demonstrations continues to grow, performance often plateaus and can even decline. This highlights a key limitation of input scaling: more context is only useful when the added information remains relevant, diverse, and well organized. To address these challenges, several methods have been proposed to improve the effectiveness and robustness of many-shot ICL [5; 234; 180]. For example, DrICL [234] adjusts demonstration weights using reinforcement-learning-inspired cumulative advantages to improve generalization, while BRIDGE [180] identifies a subset of influential examples and uses them to generate additional high-quality demonstrations.

2.2 Retrieval-Augmented Generation

Retrieval-Augmented Generation (RAG) has become a widely adopted strategy to address the limitations of LLMs, such as hallucinations and restricted generalization to concepts beyond their training data [66; 70; 53; 88]. By incorporating retrieved external information, RAG enhances factual grounding and expands the model's accessible knowledge base. However, traditional RAG operates on short retrieval units, requiring the retriever to scan a massive document corpus to find relevant passages [147; 213; 15]. This approach is constrained by input context length limitations, making long-context RAG a challenge. A common strategy is document chunking [154; 214], where LLMs retrieve relevant chunks instead of full documents. However, defining optimal chunk boundaries is difficult, often leading to semantic incoherence and contextual loss, which degrade retrieval effectiveness [97]. Recent advances in long-context LLMs allow models to process millions of tokens [176]. Integrating RAG with long-context LLMs enables the processing of extended contexts while reducing semantic incoherence in chunked retrieval [96; 216; 215].

As input length increases, the burden on retrieval systems grows. LongRAG [65] mitigates this by grouping related documents, reducing the number of retrieval operations while maintaining relevance. ReComp [214] addresses this challenge by compressing retrieved documents into textual summaries before in-context integration, ensuring information remains concise yet informative. Despite these improvements, a key challenge known as "lost-in-the-middle" bias arises [108], where LLMs assign less importance to passages in the middle of a retrieved context. MOI [86] counters this bias by aggregating inference calls from permuted retrieval orders, ensuring a more balanced weighting across the retrieved information.

Another dimension of scaling RAG involves expanding the

amount of data available at inference time [181; 8; 182; 144]. Shao et al. [160] find that increasing datastore size monotonically improves performance across various language modeling and downstream tasks without clear saturation. Their MASSIVEDS datastore, containing trillions of tokens, is designed to support large-scale retrieval efficiently. Further, Yue et al. [229] explore inference-time scaling, showing that allocating more retrieval computation leads to nearly linear performance gains when optimally distributed. Their work introduces a predictive model for optimizing retrieval parameters under computational constraints. Together, these findings suggest that input scaling in RAG is effective not only through longer contexts, but also through larger and more searchable external knowledge stores.

2.3 Memory-Augmented LLMs

Scaling reasoning capabilities of LLMs often necessitates extending their effective context beyond the limited token windows supported by existing architectures [187]. Although increasing context length allows LLMs to process longer sequences, such scaling alone quickly encounters computational bottlenecks and diminishing returns due to quadratic complexity in attention mechanisms [44]. Moreover, even very long-context models struggle to efficiently capture and retrieve critical historical information from past interactions, leading to degraded reasoning performance over extended contexts [45]. To address these limitations, memory augmentation strategies have emerged, enabling LLMs to persistently store, manage, and dynamically retrieve relevant contextual information. Current memory augmentation approaches typically follow two directions: internal architectural modifications to enhance the model's inherent memory capabilities and external memory mechanisms that extend the model context through additional memory components.

Architectural adaptations focus on internalizing long-term dependencies within the model itself. This includes techniques such as augmenting attention mechanisms to better capture extended context [104; 114], refining key-value cache mechanisms to optimize retrieval efficiency over long sequences [95; 111], and modifying positional encodings to enhance length generalization [236; 237]. While effective, these modifications require direct intervention in the model's structure, making them impractical for proprietary or black-box API-based LLMs.

An alternative approach is the integration of external memory modules to supplement the model's limited native context window. Summarization-based methods [115; 185; 122; 188] condense past interactions into structured representations that can be efficiently retrieved during inference. However, fixed-granularity summarization risks fragmenting the discourse, leading to incoherent retrieval. To address this, recent advancements incorporate dynamic memory mechanisms that adaptively refine stored information. RMM [173] exemplifies this strategy by leveraging retrospective reflection to improve retrieval selection, ensuring that the model accesses the most relevant and contextually cohesive knowledge. Scaling memory-augmented LLMs requires balancing efficiency with contextual fidelity. A key challenge is mitigating memory saturation, where excessive storage of past interactions results in retrieval inefficiencies. Techniques such as hierarchical memory organization [160] and retrieval-conditioned compression [214] help alleviate this issue by structuring and filtering stored context dynamically. As research progresses, the convergence of retrieval-augmented memory with scalable long-context architectures offers a promising avenue for enabling LLMs to maintain reasoning consistency over prolonged interactions.

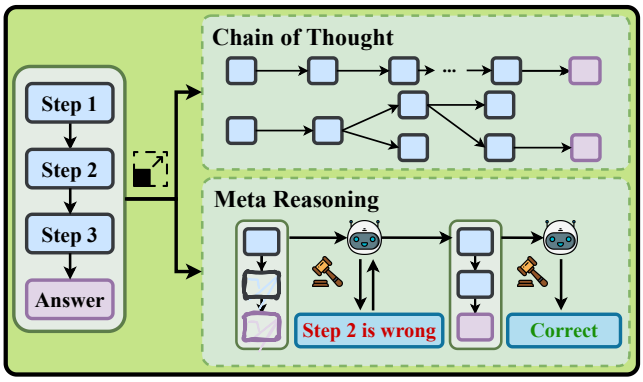

Figure 3: Scaling in LLM reasoning steps.

Overall, input scaling improves reasoning primarily by enriching the information available to the model, rather than by increasing the depth of intermediate reasoning or the amount of interactive refinement. This makes it especially effective for knowledge-intensive QA, long-context understanding, and conversational settings that depend on grounding in demonstrations, retrieved evidence, or prior interactions. At the same time, its gains are constrained by context quality, retrieval precision, and the model's ability to use long inputs effectively. Compared with later dimensions such as reasoning steps and reasoning rounds, the main bottleneck of input scaling is not insufficient inference depth, but whether the added context is relevant, well-structured, and accessible at the right time.

3. SCALING IN REASONING STEPS

Scaling reasoning steps increases the amount of intermediate computation allocated to solving a problem. Unlike input scaling, which improves reasoning by supplying more contextual information, this dimension improves performance by encouraging models to decompose problems, explore candidate solution paths, iteratively refine intermediate outputs, and verify correctness before committing to an answer. Such additional reasoning depth can substantially improve logical consistency and problem-solving ability, especially on tasks requiring structured multi-step inference. However, it also introduces important trade-offs, including higher token and search costs, longer latency, and the risk that additional reasoning may am: Chain-of-Thought prompting and meta-reasoning techniques, and discuss both their benefits and strategies to mitigate the challenges that arise from deeper reasoning processes.

3.1 Chain-of-Thought

Chain-of-Thought (CoT) prompting has emerged as a key technique for improving the reasoning capabilities of LLMs by eliciting explicit step-by-step deliberation, either through zero-shot prompting [79] or few-shot demonstrations [194]. More broadly, CoT represents a direct form of step scaling: rather than supplying the model with more external infor-

mation, it allocates more inference-time computation to constructing intermediate reasoning traces. Since LLMs operate probabilistically [61; 82], greedy decoding does not always yield the best reasoning path or final answer [190]. To mitigate this limitation, repeated sampling approaches such as self-consistency [189] and Best-of-N [132; 10] generate multiple reasoning chains in parallel and then select the final answer based on criteria such as majority agreement, external reward models, or auxiliary verifiers. They improve robustness by exploring multiple candidate trajectories, while introducing substantial computational overhead.

Although simple parallel sampling is computationally straightforward, it remains inefficient and suboptimal by randomly allocating the test-time computation budget to less promising branches [203; 168]. To mitigate this issue, researchers have explored strategies that prioritize promising reasoning paths or intermediate steps over less viable alternatives to effectively prune the search space by applying tree search-enabled reasoning [189; 221; 133; 113; 159; 127; 75] Generally, it structures the reasoning process as a branching tree, where each node represents a discrete thinking step, and branches correspond to different potential solution paths. Like CoT which organizes reasoning in a hierarchical manner, tree search-enabled reasoning enables LLMs to decompose intricate problems into manageable components. However, LLM reasoning with tree search can maintain awareness of multiple hypothesis threads simultaneously and systematically explore the solution space through different search algorithms (e.g., BFS or DFS), making it more powerful for handling complex problems.

The pioneering work CoT-SC [189] extends CoT to the tree structure, where multiple CoTs originate from the same initial (root) prompt, forming a "tree of chains". The chain that provides the best outcome to the initial question, is selected as the final answer. Skeleton-of-Thought (SoT) [133] instead effectively harnesses a tree with a specific level of depth. It performs reasoning through a divide-and-conquer manner, which significantly reduces the generation latency of LLMs. In the first prompt, the LLM is instructed to generate a skeleton of the answer, i.e., a list of points that can be answered independently. For each point, a new prompt is issued in parallel to address only the corresponding part of the question.

Recently, numerous studies have explored Tree of Thoughts (ToT) [221; 113] for tree search-enabled reasoning. Compared to CoT-SC where multiple CoTs originate from the same initial (root) prompt, ToT employs a tree structure to decompose a problem into subproblems and solve them using separate LLM prompts. Unlike ToT using multiple prompts, Algorithm of Thoughts (AoT) [159] uses only a single prompt with in-context examples formulated in an algorithmic fashion. Tree of Uncertain Thought (TouT) [127] enhances ToT with local uncertainty scores by incorporating the variance of multiple LLM responses into the state evaluation function. Tree of Clarifications (ToC) [75] focuses on answering ambiguous questions using ToT. It first retrieves relevant external information and then recursively prompts an LLM to construct a disambiguation tree for the question.

3.2 Meta-Reasoning and Calibration

Numerous works [35; 142; 49; 231; 67] have shown that LLMs have inherited capabilities of self-correction with proper prompt engineering. Typically, an LLM can self-reflect its responses by generating feedback on its answers. It first generates an initial response to an input question. Next, it generates feedback given the original input and its initial response. Finally, it generates a refined response given the input, initial response, and feedback. Generally, self-correction may rely on different sources of feedback, including intrinsic prompts and external information. Intrinsic prompts let LLMs generate feedback on their own responses. For example, CoVe [35] plans verification questions to check an initial response and then systematically answers those questions in order to finally produce an improved revised response. FLARE [67] performs self-correction by iteratively generating a temporary next sentence and check whether it contains low-probability tokens. In contrast, external information enables LLMs to rely on external tools, such as external knowledge from search engines, oracle information, and task-specific metrics, to enhance self-correction. For example, REFINER [142] interacts with a critic model that provides automated feedback on the reasoning. CRITIC [49] interacts with external tools like search engines and code interpreters to verify the desired aspects of an initial output and subsequently amends the output based on the critiques from the verification.

One major concern centers around the efficiency of self-refinement: LLMs need to generate feedback and refined responses iteratively, which can significantly increase the inference time of LLMs. To overcome the scaling issue, Quiet-STaR [231] designs a tokenwise parallel sampling algorithm, using learnable tokens indicating a thought's start and end, and an extended teacher-forcing technique. Another concern is caused by generation-time correction. Prevalent self-correction approaches are based on generation-time correction, heavily depending on the capacity of the critic model to provide accurate quantifiable feedback for intermediate outputs. Nevertheless, this might be quite challenging for many NLP tasks with long token sizes, such as summarization—the summary can be accurately assessed only after the entire summary is generated. This limitation makes generation-time correction infeasible in many NLP tasks. One solution to this issue is post-hoc correction [138]. Unlike general generation-time correction which generates feedback on the intermediate reasoning steps, post-hoc correction involves refining the output after it has been generated.

Overall, step scaling improves reasoning by allocating more computation to decomposition, search, verification, and correction. Compared with input scaling, which primarily addresses deficiencies in available evidence or context, step scaling is most effective when the main bottleneck lies in the reasoning process itself, as in mathematical, logical, and planning tasks. However, its gains are constrained by search efficiency, verification reliability, and the model's ability to avoid overthinking or compounding early mistakes. Relative to later dimensions such as reasoning rounds, which broaden reasoning through interaction, step scaling deepens a single model's inference process and therefore offers a more direct but often less diverse form of reasoning improvement.

4. SCALING IN REASONING ROUNDS

Scaling reasoning rounds expands the reasoning process through iterative interaction rather than through a single forward pass or a single model's internal chain of thought. Un-

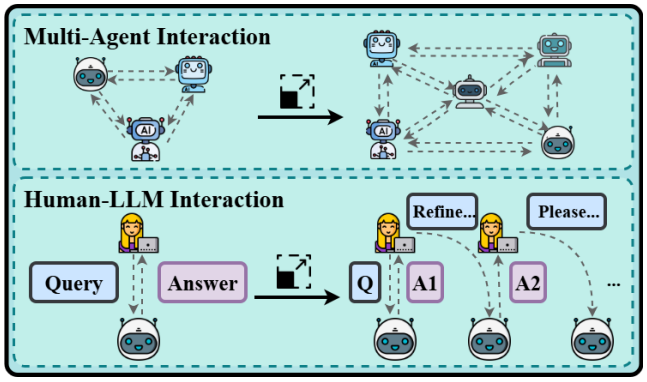

Figure 4: Scaling in reasoning rounds.

like input scaling, which enriches the information available to the model, and step scaling, which deepens intermediate computation within one reasoning trajectory, round scaling improves performance by introducing repeated communication, critique, and refinement across multiple turns. This additional interaction can increase diversity of perspectives, expose hidden errors, and support iterative improvement, but it also comes with significant trade-offs, including coordination overhead, latency, redundancy, and diminishing returns as the number of rounds grows. This section examines two major paradigms of round scaling: multi-agent interaction, where multiple LLMs coordinate or debate across rounds, and human–LLM interaction, where iterative human feedback guides and stabilizes the reasoning process.

4.1 Multi-Agent Interaction

Multi-agent interaction has emerged as a powerful paradigm for scaling LLM reasoning by enabling multiple models to iteratively exchange information, challenge assumptions, and refine outputs. Broadly, existing approaches fall into two major categories: *collaborative* frameworks, which emphasize specialization and division of labor, and *debate-based* frameworks, which introduce adversarial reasoning to expose errors and strengthen the final decision.

In collaborative settings, multiple LLMs work together in a coordinated manner to achieve improved problem-solving capabilities [100; 72; 90]. In particular, in these frameworks, each LLM (agent) is assigned a distinct role—such as planner, executor, verifier, or critic—and iteratively refines its output through structured interactions with other agents [217]. For example, CAMEL [90] introduced a framework where LLM agents assume different personas and interact through structured role-playing, enabling more effective task completion through multi-turn communication. The core idea is to enhance the specialization and division of labor among LLMs, ensuring that different agents contribute unique perspectives to improve overall task performance. Unlike single-agent systems, which rely on an LLM's internal reasoning capability [51; 198], multi-agent frameworks distribute tasks across multiple agents that engage in iterative interactions [90].

Increasing the number of agents can improve task diversity and allow for role specialization, where different agents assume distinct functions such as problem decomposition, tool usage, or evaluation [52]. Research has demonstrated that larger multi-agent systems can achieve greater accuracy and better adaptability in open-ended reasoning tasks, as seen in software development frameworks like MetaGPT [57]. However, these gains are not monotonic. Beyond a certain scale, performance may plateau or even decline due to conflicting reasoning paths, redundancy, and growing coordination overhead [100]. Similarly, [149] shows that structured dialogue among LLM agents improves reasoning depth and solution diversity, but also finds that too many interaction rounds lead to diminishing returns, as agents increasingly reinforce each other's biases rather than contribute genuinely new insights. These findings suggest that naive scaling of agent count or communication depth is insufficient; effective round scaling requires careful coordination protocols and complementary role assignment. Several works therefore introduce explicit communication structures to mitigate these issues. Hierarchical frameworks, in which some LLMs act as supervisors while others serve as task executors, have shown consistent gains in both accuracy and efficiency [13]. Another interesting finding is introduced in LLM Harmony [149], which optimizes inter-agent communication by structuring dialogue between multiple LLM agents. Instead of simple turn-based exchanges, this framework enables agents to dynamically negotiate task objectives, delegate subtasks, and refine outputs iteratively. The results suggest that scaling the number of interacting agents improves performance only when they are given complementary roles, while increasing homogeneous agents leads to redundant reasoning patterns.

In contrast to collaborative frameworks, debate-based methods deliberately assign *adversarial* roles to LLMs and often introduce an explicit judge. In these setups, each agent acts as a debater that challenges others and attempts to persuade a judge, with the goal of surfacing errors and stronger arguments rather than directly solving subtasks. A pioneering example is Multi-Agent Debate (MAD) [101], which proposes a structured debate protocol with a "tit-for-tat" mechanism: multiple debaters exchange arguments over several rounds, and a designated judge aggregates the discussion to reach a final decision. The key idea is to amplify disagreement and critical scrutiny. Compared with self-reflection approaches [120; 165], MAD induces stronger disagreement, helping to avoid premature convergence on incorrect answers. Building on this idea, subsequent debate-based frameworks improve reasoning robustness and factual accuracy by refining debate protocols and judge designs [40].

The scaling effect in debate frameworks manifests in multiple dimensions. In [73], the authors find that when employing a judge LLM to evaluate responses from debater LLMs, increasing the number of debate rounds does not necessarily lead to greater clarity—especially for weaker models, where additional rounds introduce confusion rather than improving accuracy. However, in consultancy-based interactions, where a single LLM attempts to persuade a judge LLM, the judge's accuracy improves over successive rounds. Notably, enhancing the persuasiveness of debater LLMs—making them more effective at convincing the judge—has been shown to yield performance improvements. This scaling effect provides further insights into optimizing debate-based reasoning frameworks. Similarly, [125] suggests that scaling LLM debates with increasingly skilled debaters (e.g., progressing from AI to human debaters) enhances oversight mechanisms, improving overall debate efficacy, whereas consul-

tancy frameworks tend to perform worse under similar conditions. Distinct from these approaches, CIPHER [143] proposes embedding-based communication to facilitate debate, enabling smaller LLMs to retain stronger debate capabilities by mitigating information loss. Their findings indicate that increasing the number of debate rounds improves performance up to a threshold of three rounds, beyond which additional rounds provide diminishing returns. Overall, multi-agent interaction shows that round scaling can improve reasoning not only by increasing deliberation length, but also by introducing complementary roles, disagreement, and iterative critique. At the same time, it reveals a defining challenge of this dimension: additional rounds are helpful only when they generate genuinely new information or perspectives, rather than repeated, biased, or poorly exchanges.

4.2 Human-LLM Interaction

Reasoning rounds can also be scaled through iterative interaction between humans and LLMs. In this setting, improvement does not come from communication among multiple models, but from repeated user guidance that helps the model clarify goals, correct mistakes, and refine its responses. This human-in-the-loop paradigm shifts LLMs from static inference engines to adaptive assistants whose reasoning can be steered and stabilized through feedback [3; 202]. Recent work explores multi-turn reasoning scenarios where users provide incremental clarifications or corrections, allowing models to refine their responses iteratively [120; 80]. This process mirrors how humans engage in collaborative problem-solving, gradually converging on an accurate and well-structured answer. Methods such as self-reflection prompting [165] and feedback-based reinforcement learning [17] demonstrate improvements in factual consistency and reasoning depth by enabling LLMs to assess and revise their outputs.

A key challenge in human-LLM interaction is balancing efficiency with adaptability. Over-reliance on explicit feedback mechanisms can introduce cognitive overhead for users, while insufficient adaptability limits the model's ability to incorporate nuanced human guidance. Recent strategies mitigate this tradeoff through adaptive interaction mechanisms, such as retrieval-enhanced dialogue memory [139] and user-intent modeling [92], allowing LLMs to anticipate user needs and refine responses proactively.

As interaction frameworks scale, ensuring alignment with human cognitive processes remains critical. Fine-tuning strategies that incorporate user feedback loops have shown promise in enhancing model interpretability and trustworthiness [76]. Furthermore, inference-time intervention mechanisms [123; 172] enable LLMs to allocate computational resources efficiently based on user engagement patterns. By refining the synergy between LLMs and human oversight, interactive reasoning systems hold the potential to scale beyond static prompt-response architectures, evolving towards more adaptive and contextually aware AI assistants.

Overall, round scaling improves reasoning by broadening the inference process through interaction, critique, and iterative refinement. Compared with step scaling, which deepens a single reasoning trajectory, round scaling introduces external feedback and perspective diversity, making it particularly useful for open-ended reasoning, oversight, and collaborative decision-making. However, its gains are constrained by communication quality, role complementarity, and coordination efficiency, and its characteristic failure modes include redundancy, premature consensus, noisy debate, and excessive reliance on user effort. Relative to training-enabled reasoning, which seeks to internalize better reasoning policies in advance, round scaling remains more flexible at inference time but also more dependent on well-designed interaction protocols.

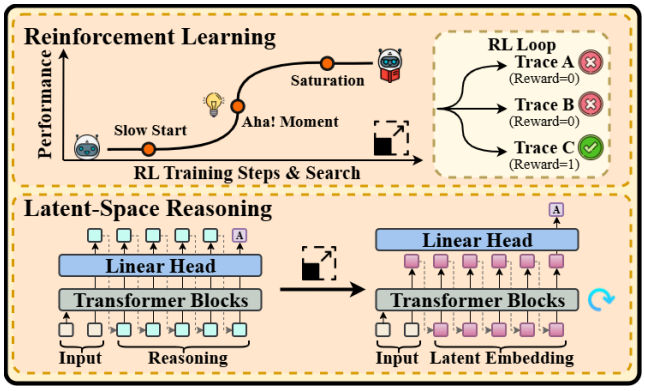

Figure 5: Scaling in model optimization.

5. SCALING IN MODEL OPTIMIZATION

Scaling model optimization improves reasoning not by supplying more external information or extending explicit inference trajectories, but by strengthening the model's internal reasoning capacity through training or latent computation. Unlike the previous three dimensions, which primarily allocate additional resources at inference time, this dimension seeks to internalize better reasoning policies in advance or to expand internal computation without proportionally increasing explicit reasoning length. Reinforcement learning (RL)–based methods scale reasoning by optimizing the model's policies over increasingly complex tasks, aligning behavior with human intentions and enabling deeper multi-step inference. Complementing RL approaches, latent-space reasoning methods scale internal computation without increasing sequence length or model size. By iterating over hidden representations, such as an in-looped transformer, these methods allow models to perform additional internal reasoning steps, effectively scaling "thinking time" while keeping inference cost manageable. Together, Reinforcement Fine-Tuning (RFT) and latent-space reasoning illustrate how optimization-time scaling can deepen LLM reasoning capacity, offering alternatives to traditional scaling via larger models or longer explicit reasoning traces.

5.1 Reinforcement Learning

Although previous studies have shown that distilling knowledge from superior LLMs, regardless of whether supervised fine-tuning (SFT) data are amassed in large quantities or carefully curated [240; 223], can enhance the reasoning abilities of smaller models for solving complex tasks [166; 121; 56], recent studies contend that, merely increasing the volume of SFT data typically yields only a log-linear performance improvement [228]. Moreover, models trained exclusively on SFT data tend to overfit by memorizing the training set, thereby struggling to generalize to out-of-distribution (OOD) tasks [30]. To mitigate these issues, reinforcement

learning (RL) has emerged as a key approach in LLM post-training, aligning models with human preferences [136; 148] and enhancing their reasoning abilities [161; 218; 50].

Recent studies indicate that conducting RL-based fine-tuning following SFT can further improve the reasoning abilities of LLMs. ReFT [119] first performs a warm-up SFT on distilled CoT data followed by fine-tuning the SFT model using Proximal Policy Optimization (PPO) [158] on the same training questions, which eventually leads to significant performance gains on mathematical reasoning tasks. T1 [58] employs a similar training strategy, but emphasizes scaling sampling diversity during RL training through techniques such as high-temperature sampling, on-policy KL normalization, and rule-based reward penalties for undesirable repetition responses. They observed that increasing the number of sampled responses, raising the sampling temperature during RL training, and extending inference-time reasoning steps collectively contribute to improved reasoning performance. DeepSeek-R1 [50] shares a similar strategy as ReFT but employs self-training by directly applying Group Relative Policy Optimization (GRPO) [161] to the base model. This base model is then used to generate long-form CoT data for the warm-up SFT stage, after which GRPO is applied again to the SFT model, ultimately achieving reasoning performance comparable to OpenAI-o1 [60]. They observed an "aha-moment" during the training of DeepSeek-R1-Zero, where the model learned to rethink as the response length increased. Following DeepSeek-R1, recent works observed similar "aha-moment" and think related words on different tasks, including real-world software engineering [197], logical puzzles [210], and automated theorem proving [37] when scaling up the training steps and response length using RL-based fine-tuning.

At the same time, RL-trained reasoning models that produce long CoT traces can exhibit notable failure modes, including "underthinking" [191], where the model frequently switches between shallow reasoning branches without engaging in sustained deliberation, and "overthinking" [22], where excessive reasoning on simple instances can degrade accuracy. Beyond these empirical observations, recent work has begun to systematically characterize the compute–scaling behavior of RL post-training. ScaleRL [74] demonstrates that RL reward improvements follow a predictable sigmoidal compute–performance curve, with early low-gain regions, a sharp transition phase, and a clear asymptotic limit. Their analysis shows that many commonly tuned components, including curriculum design, normalization, and loss aggregation, primarily affect compute efficiency, whereas only a small subset of architectural or algorithmic choices (e.g., loss formulation, precision handling, off-policy setup) materially shifts the ultimate performance ceiling. In parallel, ProRL [107] provides complementary evidence that prolonged RL optimization can elicit qualitatively new reasoning strategies even in relatively small models, though it does not explicitly analyze the predictability of compute scaling. Taken together, these studies suggest that RL post-training is not only empirically effective but also exhibits a structured, saturating scaling pattern, underscoring the importance of understanding compute–performance curves rather than evaluating methods solely at isolated endpoints.

5.2 Latent-Space Reasoning

A second pathway for optimization-based scaling increases reasoning capacity by allocating additional computation in latent space rather than through longer explicit reasoning traces. In explicit reasoning [194], models generate intermediate natural-language steps before producing a final output. While such reasoning improves interpretability and decomposition, it can also be verbose and computationally expensive. Latent-space reasoning aims to address this limitation by performing additional computation over hidden representations without requiring every intermediate thought to be verbalized [33; 164]. This makes it possible to scale "thinking time" without proportionally increasing sequence length or model size.

Several recent methods instantiate this idea in different ways. Deng et al. [33] propose distilling multi-step reasoning into latent representations across layers, allowing the model to solve complex problems in a single forward pass while improving efficiency and scalability. CoCoMix [170] trains LLMs to predict selected semantic concepts directly from hidden states; by interleaving token embeddings with high-level continuous concepts, it enhances abstract reasoning while reducing data and computation costs. More broadly, this line of work reflects a key observation: natural language is not always the most efficient substrate for reasoning. Hao et al. [54] argue that many word tokens mainly serve textual coherence rather than reasoning itself, whereas only certain critical tokens require deeper planning. Based on this insight, Coconut [54] iteratively processes hidden states and enables parallel exploration of multiple reasoning paths in latent space. Additional work explores how iterative latent computation can deepen reasoning without requiring parameter expansion. ITT [23], for example, dynamically allocates computation to critical tokens and iteratively refines hidden representations. The same iterative paradigm has also been explored for test-time scaling [47; 129], where repeated latent transformations improve efficiency relative to explicitly lengthening verbalized reasoning. Similarly, Saunshi et al. [155] show that model depth can effectively be scaled under a limited parameter budget through looping, introducing a new scaling paradigm based on iterative latent-space transformations rather than simply enlarging the model. Collectively, these methods suggest that deeper reasoning need not always correspond to longer visible chains of thought; in some settings, the key resource being scaled is internal computation itself.

Overall, optimization-based scaling improves reasoning by internalizing stronger reasoning policies or by expanding internal computation through latent iterative processing. Compared with input scaling, step scaling, and round scaling, which primarily invest additional resources at inference time, this dimension shifts the trade-off toward up-front optimization cost in exchange for potentially reusable reasoning improvements across many downstream tasks. This makes it particularly attractive in high-value settings where stronger reasoning behavior can be amortized over repeated deployment. However, its gains are constrained by optimization stability, reward fidelity, transferability, and the difficulty of understanding how internalized or latent reasoning generalizes beyond the training conditions.

6. APPLICATION

6.1 AI Research

Scaling in LLMs has fundamentally reshaped AI research, both extending traditional domains and opening entirely new research avenues. This section explores how scaling has influenced three critical areas: LLM-as-a-Judge, fact-checking, and dialogue systems.

LLM-as-a-Judge. Using LLMs to evaluate model outputs or other models has emerged as a pivotal research direction, enabling evaluation at scale beyond traditional approaches and human assessment [89]. Notably, larger models demonstrate a significantly higher correlation with human preferences compared to their smaller counterparts [238]. To further improve evaluation quality, recent work has explored multi-step reasoning processes [152], where scaling the number of reasoning steps enhances evaluation capabilities [29]. Additionally, scaling across multiple judge models has emerged as an effective approach to improve evaluation reliability [99]. Different LLMs functioning as agents collaborate through multi-round discussions before reaching a final judgment, thereby enhancing evaluation consistency [146].

Fact-Checking. The capacity of AI systems to generate misinformation has driven substantial research into automated fact checking [200; 32; 242]. Initial fact verification approaches relied on smaller models with limited contextual understanding, primarily focusing on matching claims to evidence [32]. Large-scale LLMs have shown remarkable fact-checking capabilities by supporting fact-checkers with their extensive knowledge and sophisticated reasoning [175]. Scaling in reasoning steps has been demonstrated to improve claim detection, making the process more methodical [157]. Additionally, RAG has been employed for evidence-backed fact-checking with reduced hallucination and improved performance, with performance scaling with the number of retrieved documents [167]. Multi-agent systems have been widely implemented for fact-checking, where multiple imperfect fact-checkers can collectively provide reliable assessments [178].

Dialogue Systems. Dialogue systems represent the most visible application of LLM scaling [224; 239; 43], where advances in context length, reasoning steps, and training data have dramatically transformed interactive capabilities. Enhanced context handling has significantly impacted dialogue coherence and consistency. Scaling of context provides dialogue agents with more information, enabling more informative long-term conversations [6; 173]. External augmentation has been widely adopted to facilitate long-term dialogue as well. Commonly integrated external knowledge, including commonsense [183], medical [21], and psychological [24] knowledge, serves as supplementary guidance for the reasoning process, ensuring logical coherence across extended contexts. Multi-agent dialogue systems have also demonstrated exceptional capabilities, where multiple LLMs collaborate to comprehensively evaluate and select the most appropriate responses [42].

6.2 Production

The scaling reasoning capabilities of LLMs have significantly enhanced production applications, particularly in software development, data science workflows, and interactive AI systems. This subsection discusses these areas with illustrative examples.

Software Development. The scaling reasoning capabilities of LLMs enhance software development by enabling a better understanding of complex coding tasks and facilitating accurate multi-step reasoning over intricate software dependencies and structures. Advanced reasoning techniques, such as chain-of-thought prompting, allow code-generation assistants to systematically approach and solve coding tasks [20; 64]. Furthermore, structured reasoning strategies can effectively handle larger coding contexts and reduce developer cognitive load during debugging and iterative improvement processes [64].

Data Science Workflows. Scaling reasoning in LLMs substantially improves data science workflows by enabling sophisticated analytical and exploratory tasks. Multi-step reasoning allows LLMs to iteratively explore, interpret, and synthesize insights from diverse datasets [171], effectively supporting hypothesis generation and validation processes [162; 201]. Retrieval-augmented reasoning frameworks extend these capabilities further by dynamically integrating external knowledge during reasoning, thus enriching the comprehensiveness of exploratory analysis [144]. Multi-agent systems are also proposed to collaboratively solve real-world data science challenges [98].

Interactive AI Systems. Scaling reasoning steps and context length transforms interactive AI systems by significantly improving their adaptability and context-awareness. Expanded reasoning capabilities enable dialogue agents to maintain coherent and informative long-term interactions, effectively integrating historical context and external knowledge [6; 43]. Multi-agent systems leverage iterative refinement and structured verification among specialized reasoning agents, further enhancing accuracy and reducing errors such as hallucinations [42]. Interactive AI environments such as LLM-based Cursor [34] leverage LLMs' contextual reasoning to facilitate precise user interactions, enabling targeted queries and refined outputs.

6.3 Science

The scaling of LLMs has significantly benefited scientific domains, with medicine, finance, and disaster management emerging as prominent application areas.

Medical Domain. The medical domain has experienced remarkable advances through scaled LLMs. Research demonstrates that increasing model size leads to enhanced medical reasoning capabilities, with performance on medical questions improving proportionally [9; 102; 116]. This pattern extends to diagnostic reasoning [48; 156], where larger models can identify complex disease progression patterns that smaller models miss [230; 46]. Multi-round reasoning approaches such as CoT have demonstrated exceptional effectiveness in medical diagnosis [199; 106], with additional reasoning steps yielding more accurate diagnoses [59; 16] by enabling consideration of alternative explanations and confounding factors. RAG techniques enhance medical question answering, with performance improving as the number of retrieved snippets increases [212]. Many-shot ICL shows particular efficacy for drug design tasks, with performance scaling with the number of examples provided [128]. Additionally, multi-agent LLM frameworks that simulate medical team consultations have demonstrated superior diagnostic accuracy, with specialized agents collaborating on complex cases to outperform single LLMs when benchmarked against

gold-standard diagnoses [41; 78].

Finance. Financial applications demonstrate improved performance with large-scale LLMs. Studies indicate that fine-tuned large-scale LLMs substantially outperform smaller alternatives [68], with performance scaling with model size [145; 91]. The multi-step reasoning capabilities of scaled LLMs prove particularly valuable for complex financial analysis, significantly outperforming direct approaches [243; 145]. Financial sentiment analysis benefits from increased numbers of examples in many-shot ICL scenarios [2; 196]. RAG-based approaches incorporating banking webpages and policy guides improve question-answering performance, with results scaling with the number of retrieved documents [235]. Multi-agent debate frameworks yield promising results in investment and trading decision scenarios [227; 226; 209], with specialized agents covering distinct functions outperforming single-agent approaches.

Disaster Management. Disaster management has undergone substantial transformation through large-scale LLMs [87]. Social media text classification for disaster types has improved significantly through LLM fine-tuning compared to traditional machine learning methods [225; 39]. The in-context learning capabilities of large-scale LLMs enable context-aware disaster applications including conversational agents for disaster-related queries and situational analysis [135; 150]. Large-scale disaster knowledge graphs enhance in-context learning through retrieval augmentation, enabling LLMs to generate more informative and less hallucinated responses [19; 205]. For high-stakes disaster-related decision-making, multi-agent LLM approaches have been effectively deployed to facilitate adaptive and collaborative decision processes [36; 177], largely outperforming a single LLM.

7. FUTURE DIRECTIONS

Efficiency in Scalable Reasoning. Scaling reasoning capability in LLMs enhances their ability to solve complex problems but also increases response length, making it inefficient for simpler tasks. However, current LLMs apply uniform reasoning effort across all queries, leading to unnecessary computational overhead. A key direction for improvement is adaptive reasoning frameworks, where models dynamically adjust the depth of reasoning based on task difficulty [232; 195]. For example, "Proposer-Verifier" framework [168] offers a promising approach by generating multiple candidate solutions and selecting the most reliable one through verification, reducing redundant reasoning steps while maintaining accuracy. However, achieving dynamic computation allocation requires robust uncertainty estimation, ensuring that models allocate resources efficiently without excessive overhead.

Another challenge is balancing search-based reasoning methods with computational cost. Approaches like ToT and Monte Carlo search refine reasoning iteratively but incur significant compute overhead. Selective pruning strategies that eliminate irrelevant reasoning paths while maintaining solution integrity could help optimize performance [211]. Additionally, RL-based multi-step reasoning faces credit assignment issues, where sparse rewards make optimizing intermediate reasoning steps difficult [82]. Future work should explore hybrid reward models [163] that combine process-based supervision (evaluating stepwise correctness) with some outcome-based rewards (final answer validation) to improve long-horizon reasoning stability and efficiency.

Beyond single-model scaling, collaborative multi-agent systems present a promising avenue for large-scale reasoning [85; 137], but they also introduce significant coordination overhead. As the number of agents increases, computational redundancy and inefficient communication can slow down reasoning instead of improving it [51]. One approach to mitigate this is dynamic agent selection [112], where the system dynamically selects only the most relevant agents for a given reasoning task while discarding redundant ones. Another strategy is hierarchical multi-agent reasoning, where a smaller subset of expert agents handles complex queries, while simpler queries are resolved by lightweight, lower-cost agents. Additionally, inter-agent communication should be optimized through compressed latent representations rather than verbose token-based exchanges, further reducing computational overhead [244]. Future research should explore pruning and optimization techniques that enable multi-agent systems to scale efficiently without unnecessary computational waste, ensuring that reasoning is distributed optimally across agents.

Inverse Scaling and Stability. Inverse scaling refers to the phenomenon where LLMs unexpectedly perform worse on certain tasks, contradicting standard scaling laws that predict consistent improvements with increased model size. Lin et al. [103] first observed this effect when evaluating LLMs such as GPT-2 and GPT-3 on truthfulness tasks, noting that common training objectives incentivize imitative falsehoods, where models produce false but high-likelihood responses due to patterns in their training distribution. McKenzie et al. [124] systematically analyzed different datasets exhibiting inverse scaling and identified key causes like solving distractor tasks instead of intended tasks.

While inverse scaling is widely observed, Wei et al. [192] challenge its universality, showing that some tasks previously exhibiting inverse scaling follow a U-shaped scaling trend—where performance initially declines with increasing model size but later recovers at even larger scales. This suggests that larger models can sometimes unlearn distractor tasks and correct their errors, emphasizing the importance of evaluating scaling trends beyond mid-sized models.

Since scaling laws were originally developed in the context of pretraining, they remain decoupled from downstream task performance, making it an open question of how to systematically predict and mitigate inverse scaling across different reasoning benchmarks. Additionally, challenges like reward hacking [4]—where models exploit superficial signals rather than true reasoning improvements—necessitate adaptive reward models to maintain stability in multi-step reasoning.Future work should focus on developing predictive models for inverse scaling, refining adaptive fine-tuning methods, and leveraging world models for richer environmental feedback, ensuring that multi-step reasoning generalizes effectively across domains such as code generation, planning, question answering, and cross-lingual tasks.

Security Risks in Scaled Reasoning Models. While CoT prompting enhances LLMs' ability to perform structured reasoning, it also introduces new security vulnerabilities, particularly backdoor attacks that manipulate the model's reasoning process. BadChain [207] exploits the model's step-by-step reasoning by injecting backdoor reasoning steps,

causing malicious alterations in the final response when a hidden trigger is present in the query. Similarly, H-CoT [83] manipulates the model's internal reasoning pathways, hijacking its safety mechanisms to weaken its ability to detect harmful content. While defenses such as backdoor detection (CBD) [208] and modified decoding strategies [63] offer some protection, their effectiveness against novel attacks remains largely unexplored. This highlights the urgent need for more robust defenses capable of adapting to emerging threats.

Unlike CoT, RAG integrates external data sources, making them prone to data extraction attacks [28]. Existing defenses primarily focus on retrieval corruption attacks [206; 174; 241], aiming to maintain performance, but data leakage prevention remains an underexplored area. For example, RAG-Thief demonstrates how attackers can extract scalable amounts of private data from proprietary retrieval databases [62]. Beyond attacks on individual LLMs, the scaling of multi-agent reasoning systems introduces new attack surfaces. AgentPoison [27] specifically targets RAG-based and memory-augmented LLM agents, poisoning long-term memory or altering the knowledge base to induce faulty reasoning over time. As multi-agent LLM systems grow in scale, collusive behaviors among malicious agents present an even greater risk [219]. BlockAgents proposes a blockchain-integrated framework for LLM-based cooperative multi-agent systems, mitigating Byzantine behaviors that arise from adversarial agents [14].

As AI adoption increases, the computational and environmental costs of inference also become a growing concern [118; 153; 141]. Large-scale LLMs demand significant energy resources on inference [141]. This opens the door to a new form of attack, OverThink attack [81], where an adversary intentionally inflates the number of reasoning tokens in an LLM's response, drastically increasing financial and computational costs. As LLM reasoning continues to scale, deploying cost-effective safeguards against such attacks will become necessary for sustainable AI deployment.

8. CONCLUSION

In this survey, we presented a comprehensive view of how different scaling strategies shape the reasoning capabilities of large language models. We organized the literature along four major dimensions: input information, reasoning steps, reasoning rounds, and model optimization, and discussed the key methods, benefits, trade-offs, and failure modes associated with each. Our analysis highlights that scaling can substantially improve LLM reasoning across a wide range of domains, but these gains are not uniform: they often come with increased computational cost, instability, diminishing returns, and emerging safety and security risks. We further outlined several promising directions for future research, including adaptive computation allocation, more robust and stable optimization, principled evaluation beyond final-answer accuracy, and safer multi-agent and human-LLM interaction. As LLMs continue to advance, a deeper understanding of how to scale reasoning effectively and responsibly will be essential for building AI systems that are not only more capable but also more efficient, reliable, and trustworthy.

9. ACKNOWLEDGEMENTS

This work is supported in part by the National Science Foundation (NSF) under grants IIS-2006844, IIS-2144209, IIS-2223769, CNS-2154962, BCS-2228534, CMMI-2411248, ECCS-2143559, and CPS-2313110; the Office of Naval Research (ONR) under grant N000142412636; and the Commonwealth Cyber Initiative (CCI) under grant VV1Q24-011.

10. REFERENCES

[1] A. Abedsoltan, A. Radhakrishnan, et al. Context-scaling versus task-scaling in in-context learning. *arXiv*, 2024.

[2] R. Agarwal, A. Singh, et al. Many-shot in-context learning. *NeurIPS*, 2024.

[3] D. Alsagheer, R. Karanjai, et al. Comparing rationality between large language models and humans: Insights and open questions. *arXiv*, 2024.

[4] D. Amodei, C. Olah, et al. Concrete problems in ai safety. *arXiv*, 2016.

[5] J. Baek, S. J. Lee, et al. Revisiting in-context learning with long context language models. *arXiv*, 2024.

[6] J. Bang, H. Noh, et al. Example-based chat-oriented dialogue system with personalized long-term memory. In *BIGCOMP*, 2015.

[7] A. Bertsch, M. Ivgi, et al. In-context learning with long-context models: An in-depth exploration. *arXiv*, 2024.

[8] S. Borgeaud, A. Mensch, et al. Improving language models by retrieving from trillions of tokens. In *ICML*, 2022.

[9] D. Brin, V. Sorin, et al. How large language models perform on the united states medical licensing examination: a systematic review. *MedRxiv*, 2023.

[10] B. Brown, J. Juravsky, et al. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv*, 2024.

[11] T. Brown, B. Mann, et al. Language models are few-shot learners. *NeurIPS*, 2020.

[12] T. Cai, K. Huang, et al. Scaling in-context demonstrations with structured attention. *arXiv*, 2023.

[13] T. Cai, X. Wang, et al. Large language models as tool makers. *arXiv*, 2023.

[14] B. Chen, G. Li, et al. Blockagents: Towards byzantine-robust llm-based multi-agent coordination via blockchain. In *ACM TURC*, 2024.

[15] D. Chen, A. Fisch, et al. Reading wikipedia to answer open-domain questions. *arXiv*, 2017.

[16] J. Chen, Z. Cai, et al. Huatuogpt-o1, towards medical complex reasoning with llms. *arXiv*, 2024.

[17] K. Chen, M. Cusumano-Towner, et al. Reinforcement learning for long-horizon interactive llm agents. *arXiv*, 2025.

[18] L. Chen, J. Q. Davis, et al. Are more llm calls all you need? towards the scaling properties of compound ai systems. *NeurIPS*, 2024.

[19] M. Chen, Z. Tao, et al. Enhancing emergency decision-making with knowledge graphs and large language models. *IJDRR*, 2024.

[20] M. Chen, J. Tworek, et al. Evaluating large language models trained on code. *arXiv*, 2021.

[21] S. Chen, M. Wu, et al. Llm-empowered chatbots for psychiatrist and patient simulation: application and evaluation. *arXiv*, 2023.

[22] X. Chen, J. Xu, et al. Do not think that much for 2+ 3=? on the overthinking of o1-like llms. *arXiv*, 2024.

[23] Y. Chen, J. Shang, et al. Inner thinking transformer: Leveraging dynamic depth scaling to foster adaptive internal thinking. *arXiv*, 2025.

[24] Y. Chen, X. Xing, et al. Soulchat: Improving llms' empathy, listening, and comfort abilities through fine-tuning with multi-turn empathy conversations. *arXiv*, 2023.

[25] Z. Chen, A. H. Cano, et al. Meditron-70b: Scaling medical pretraining for large language models. *arXiv*, 2023.

[26] Z. Chen, S. Wang, et al. Fastgas: Fast graph-based annotation selection for in-context learning. *ACL*, 2024.

[27] Z. Chen, Z. Xiang, et al. Agentpoison: Red-teaming llm agents via poisoning memory or knowledge bases. *NeurIPS*, 2024.

[28] P. Cheng, Y. Ding, et al. Trojanrag: Retrieval-augmented generation can be backdoor driver in large language models. *arXiv*, 2024.

[29] C.-H. Chiang, H.-y. Lee, et al. Tract: Regression-aware fine-tuning meets chain-of-thought reasoning for llm-as-a-judge. *arXiv*, 2025.

[30] T. Chu, Y. Zhai, et al. Sft memorizes, rl generalizes: A comparative study of foundation model post-training. *arXiv*, 2025.

[31] H. W. Chung et al. Scaling instruction-finetuned language models. *JMLR*, 2024.

[32] G. Demartini, S. Mizzaro, et al. Human-in-the-loop artificial intelligence for fighting online misinformation: Challenges and opportunities. *IEEE Data Eng. Bull.*, 2020.

[33] Y. Deng, K. Prasad, et al. Implicit chain of thought reasoning via knowledge distillation. *arXiv*, 2023.

[34] D. S. R. Devi, O. U. C. BhagyaSri, R. Sravanthi, S. Chaitrika, M. Priyanka, M. Swarna, and M. Srilekha. Ai-enhanced cursor navigator. *R. and Chaitrika, SL and Priyanka, MN and Swarna, M. and Srilekha, M., AI-Enhanced Cursor Navigator (May 10, 2024)*, 2024.

[35] S. Dhuliawala, M. Komeili, et al. Chain-of-verification reduces hallucination in large language models. In *ACL*, 2024.

[36] A. Dolant and P. Kumar. Agentic llm framework for adaptive decision discourse. *arXiv*, 2025.

[37] K. Dong and T. Ma. Stp: Self-play llm theorem provers with iterative conjecturing and proving. *arXiv*, 2025.

[38] Q. Dong, L. Li, et al. A survey on in-context learning. In *EMNLP*, 2024.

[39] V. G. dos Santos, G. L. Santos, et al. Identifying citizen-related issues from social media using llm-based data augmentation. In *CAiSE*, 2024.

[40] Y. Du, S. Li, et al. Improving factuality and reasoning in language models through multiagent debate. *arXiv*, 2023.

[41] Z. Fan, J. Tang, et al. Ai hospital: Benchmarking large language models in a multi-agent medical interaction simulator. *arXiv*, 2024.

[42] J. Fang, S. Gao, et al. A multi-agent conversational recommender system. *arXiv*, 2024.

[43] L. Friedman, S. Ahuja, et al. Leveraging large language models in conversational recommender systems. *arXiv*, 2023.

[44] Z. Fu, W. Song, et al. Sliding window attention training for efficient large language models. *arXiv*, 2025.

[45] Y. Gao, Y. Xiong, et al. U-niah: Unified rag and llm evaluation for long context needle-in-a-haystack. *arXiv*, 2025.

[46] Á. García-Barragán, A. G. Calatayud, et al. Step-forward structuring disease phenotypic entities with llms for disease understanding. In *CBMS*, 2024.

[47] J. Geiping, S. McLeish, et al. Scaling up test-time compute with latent reasoning: A recurrent depth approach. *arXiv*, 2025.

[48] E. Goh and R. o. Gallo. Large language model influence on diagnostic reasoning: a randomized clinical trial. *JAMA Network Open*, 2024.

[49] Z. Gou, Z. Shao, et al. Critic: Large language models can self-correct with tool-interactive critiquing. *arXiv*, 2023.

[50] D. Guo, D. Yang, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv*, 2025.

[51] T. Guo, X. Chen, et al. Large language model based multi-agents: A survey of progress and challenges. *arXiv*, 2024.

[52] T. Guo, X. Chen, et al. Large language model based multi-agents: A survey of progress and challenges. *arXiv*, 2024.

[53] K. Guu, K. Lee, et al. Retrieval augmented language model pre-training. In *ICML*, 2020.

[54] S. Hao, S. Sukhbaatar, et al. Training large language models to reason in a continuous latent space. *arXiv*, 2024.

[55] Y. Hao, Y. Sun, et al. Structured prompting: Scaling in-context learning to 1,000 examples. *arXiv*, 2022.

[56] N. Ho et al. Large language models are reasoning teachers. *arXiv*, 2022.

[57] S. Hong, M. Zhuge, et al. Metagpt: Meta programming for a multi-agent collaborative framework. In *ICLR*, 2023.

[58] Z. Hou, X. Lv, et al. Advancing language model reasoning through reinforcement learning and inference scaling. *arXiv*, 2025.

[59] Z. Huang, G. Geng, et al. O1 replication journey–part 3: Inference-time scaling for medical reasoning. *arXiv*, 2025.

[60] A. Jaech, A. Kalai, et al. Openai o1 system card. *arXiv*, 2024.

[61] Z. Ji, N. Lee, et al. Survey of hallucination in natural language generation. *ACM Comp.Sur.*, 2023.

[62] C. Jiang, X. Pan, et al. Rag-thief: Scalable extraction of private data from retrieval-augmented generation applications with agent-based attacks. *arXiv*, 2024.

[63] F. Jiang, Z. Xu, et al. Safechain: Safety of language models with long chain-of-thought reasoning capabilities. *arXiv*, 2025.

[64] J. Jiang, F. Wang, et al. A survey on large language models for code generation. *arXiv*, 2024.

[65] Z. Jiang, X. Ma, et al. Longrag: Enhancing retrieval-augmented generation with long-context llms. *arXiv*, 2024.

[66] Z. Jiang, F. F. Xu, et al. Active retrieval augmented generation. In *EMNLP*, 2023.

[67] Z. Jiang, F. F. Xu, et al. Active retrieval augmented generation. In *EMNLP*, 2023.

[68] K. Kalluri. Scalable fine-tunning strategies for llms in finance domain-specific application for credit union, 2024.

[69] J. Kaplan, S. McCandlish, et al. Scaling laws for neural language models. *arXiv*, 2020.

[70] V. Karpukhin, B. Oguz, et al. Dense passage retrieval for open-domain question answering. In *EMNLP*, 2020.

[71] Z. Ke, F. Jiao, Y. Ming, X.-P. Nguyen, A. Xu, D. X. Long, M. Li, C. Qin, P. Wang, S. Savarese, et al. A survey of frontiers in llm reasoning: Inference scaling, learning to reason, and agentic systems. *arXiv preprint arXiv:2504.09037*, 2025.

[72] Z. Kenton, N. Y. Siegel, et al. On scalable oversight with weak llms judging strong llms. *arXiv*, 2024.

[73] A. Khan, J. Hughes, et al. Debating with more persuasive llms leads to more truthful answers. In *ICML*, 2024.

[74] D. Khatri, L. Madaan, et al. The art of scaling reinforcement learning compute for llms. *arXiv*, 2025.

[75] G. Kim, S. Kim, et al. Tree of clarifications: Answering ambiguous questions with retrieval-augmented large language models. In *EMNLP*, 2023.

[76] H. Kim, K. Lee, et al. Human implicit preference-based policy fine-tuning for multi-agent reinforcement learning in usv swarm. *arXiv*, 2025.

[77] H. J. Kim, H. Cho, et al. Self-generated in-context learning: Leveraging auto-regressive language models as a demonstration generator. *arXiv*, 2022.

[78] Y. Kim, C. Park, et al. Mdagents: An adaptive collaboration of llms for medical decision-making. *NeurIPS*, 2024.

[79] T. Kojima, S. S. Gu, et al. Large language models are zero-shot reasoners. *NeurIPS*, 2022.

[80] S. Krishna, C. Agarwal, et al. Understanding the effects of iterative prompting on truthfulness. *arXiv*, 2024.

[81] A. Kumar et al. Overthinking: Slowdown attacks on reasoning llms. *arXiv*, 2025.

[82] K. Kumar, T. Ashraf, et al. Llm post-training: A deep dive into reasoning large language models. *arXiv*, 2025.

[83] M. Kuo, J. Zhang, et al. H-cot: Hijacking the chain-of-thought safety reasoning mechanism to jailbreak large reasoning models, including openai o1/o3, deepseek-r1, and gemini 2.0 flash thinking. *arXiv*, 2025.

[84] H. Lai, X. Liu, J. Gao, J. Cheng, Z. Qi, Y. Xu, S. Yao, D. Zhang, J. Du, Z. Hou, et al. A survey of post-training scaling in large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2771–2791, 2025.

[85] H. D. Le et al. Multi-agent causal discovery using large language models. *arXiv*, 2024.

[86] Y. Lee, S.-w. Hwang, et al. Inference scaling for bridging retrieval and augmented generation. *arXiv*, 2024.

[87] Z. Lei, Y. Dong, et al. Harnessing large language models for disaster management: A survey. *arXiv*, 2025.

[88] P. Lewis, E. Perez, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *NeurIPS*, 2020.

[89] D. Li, B. Jiang, et al. From generation to judgment: Opportunities and challenges of llm-as-a-judge. *arXiv*, 2024.

[90] G. Li, H. Hammoud, et al. Camel: Communicative agents for" mind" exploration of large language model society. *NeurIPS*, 2023.

[91] H. Li, Y. Cao, et al. Investorbench: A benchmark for financial decision-making tasks with llm-based agent. *arXiv*, 2024.

[92] J. Li, T. Tang, et al. The web can be your oyster for improving large language models. *arXiv*, 2023.

[93] M. Li, S. Gong, et al. In-context learning with many demonstration examples. *arXiv*, 2023.

[94] X. Li, X.-P. Nguyen, et al. Paraicl: Towards robust parallel in-context learning. *arXiv*, 2024.

[95] Y. Li, H. Jiang, et al. Scbench: A kv cache-centric analysis of long-context methods. *arXiv*, 2024.

[96] Z. Li, C. Li, et al. Retrieval augmented generation or long-context llms? a comprehensive study and hybrid approach. In *EMNLP*, 2024.

[97] Z. Li, J. Xiong, et al. Uncertaintyrag: Span-level uncertainty enhanced long-context modeling for retrieval-augmented generation. *arXiv*, 2024.

[98] Z. Li, Q. Zang, et al. Autokaggle: A multi-agent framework for autonomous data science competitions. *arXiv*, 2024.

[99] J. Liang, R. Ye, et al. Debatrix: Multi-dimensional debate judge with iterative chronological analysis based on llm. *arXiv*, 2024.

[100] T. Liang, Z. He, et al. Encouraging divergent thinking in large language models through multi-agent debate. *arXiv*, 2023.

[101] T. Liang, Z. He, et al. Encouraging divergent thinking in large language models through multi-agent debate. In *EMNLP*, 2024.

[102] V. Liévin, C. E. Hother, et al. Can large language models reason about medical questions? *Patterns*, 2024.

[103] S. Lin, J. Hilton, et al. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv*, 2021.

[104] H. Liu, M. Zaharia, et al. Ring attention with blockwise transformers for near-infinite context. *arXiv*, 2023.

[105] J. Liu, D. Shen, et al. What makes good in-context examples for gpt-3? *arXiv*, 2021.

[106] J. Liu, Y. Wang, et al. Medcot: Medical chain of thought via hierarchical expert. *arXiv*, 2024.

[107] M. Liu, S. Diao, et al. Prorl: Prolonged reinforcement learning expands reasoning boundaries in large language models. *arXiv*, 2025.

[108] N. F. Liu, K. Lin, et al. Lost in the middle: How language models use long contexts. *TACL*, 2024.

[109] S. Liu, H. Ye, et al. In-context vectors: Making in context learning more effective and controllable through latent space steering. In *ICML*, 2024.

[110] Y. Liu, J. Liu, et al. Let's learn step by step: Enhancing in-context learning ability with curriculum learning. *arXiV*, 2024.

[111] Y. Liu, P. Yang, et al. Chunkkv: Chunk-based key-value cache management for transformer models. *arXiv*, 2025.

[112] Z. Liu, Y. Zhang, et al. Dynamic llm-agent network: An llm-agent collaboration framework with agent team optimization. *arXiv*, 2023.

[113] J. Long. Large language model guided tree-of-thought. *arXiv*, 2023.

[114] C. Lou, Z. Jia, Z. Zheng, et al. Sparser is faster and less is more: Efficient sparse attention for long-range transformers. *arXiv*, 2024.

[115] J. Lu, S. An, et al. Memochat: Tuning llms to use memos for consistent long-range open-domain conversation. *arXiv*, 2023.

[116] K. Lu, Z. Liang, et al. Med-R^2: Crafting Trustworthy LLM Physicians through Retrieval and Reasoning of Evidence-Based Medicine. *arXiv*, 2025.

[117] Y. Lu, M. Bartolo, et al. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. *arXiv*, 2021.

[118] S. Luccioni, Y. Jernite, et al. Power hungry processing: Watts driving the cost of ai deployment? In *FAccT*, 2024.

[119] T. Q. Luong et al. Reft: Reasoning with reinforced fine-tuning. *arXiv*, 2024.

[120] A. Madaan, N. Tandon, et al. Self-refine: Iterative refinement with self-feedback. *NeurIPS*, 2023.

[121] L. C. Magister et al. Teaching small language models to reason. *arXiv*, 2022.

[122] A. Maharana, D.-H. Lee, et al. Evaluating very long-term conversational memory of llm agents. *arXiv*, 2024.

[123] R. Manvi, A. Singh, et al. Adaptive inference-time compute: Llms can predict if they can do better, even mid-generation. *arXiv*, 2024.

[124] I. McKenzie et al. Inverse scaling: When bigger isn't better. *TMLR*, 2024.

[125] J. Michael et al. Debate helps supervise unreliable experts. *arXiv*, 2023.

[126] S. Min, X. Lyu, et al. Rethinking the role of demonstrations: What makes in-context learning work? In *EMNLP*, 2022.

[127] S. Mo and M. Xin. Tree of uncertain thoughts reasoning for large language models. In *ICASSP*, 2024.

[128] S. Moayedpour, A. Corrochano-Navarro, et al. Many-shot in-context learning for molecular inverse design. *arXiv*, 2024.

[129] A. Mohtashami, M. Pagliardini, et al. Cotformer: A chain-of-thought driven architecture with budget-adaptive computation cost at inference. *arXiv*, 2023.

[130] N. Muennighoff, A. Rush, et al. Scaling data-constrained language models. *NeurIPS*, 2023.

[131] N. Muennighoff, Z. Yang, et al. s1: Simple test-time scaling. *arXiv*, 2025.

[132] R. Nakano, J. Hilton, et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv*, 2021.

[133] X. Ning, Z. Lin, et al. Skeleton-of-thought: Prompting llms for efficient parallel generation. *arXiv*, 2023.

[134] OpenAI. Gpt-4 technical report, 2024.

[135] H. T. Otal, E. Stern, et al. Llm-assisted crisis management: Building advanced llm platforms for effective emergency response and public collaboration. In *CAI*, 2024.

[136] L. Ouyang, J. Wu, et al. Training language models to follow instructions with human feedback. *NeurIPS*, 2022.

[137] D. M. Owens, R. A. Rossi, et al. A multi-llm debiasing framework. *arXiv*, 2024.

[138] L. Pan, M. Saxon, et al. Automatically correcting large language models: Surveying the landscape of diverse automated correction strategies. *TACL*, 2024.

[139] Z. Pan, Q. Wu, et al. On memory construction and retrieval for personalized conversational agents. *arXiv*, 2025.

[140] C. F. Park, A. Lee, et al. Iclr: In-context learning of representations. *arXiv*, 2024.

[141] D. Patterson, J. Gonzalez, et al. The carbon footprint of machine learning training will plateau, then shrink. *Computer*, 2022.

[142] D. Paul, M. Ismayilzada, et al. Refiner: Reasoning feedback on intermediate representations. In *EACL*, 2024.

[143] C. Pham, B. Liu, et al. Let models speak ciphers: Multiagent debate through embeddings. In *ICLR*, 2024.

[144] A. Piktus, F. Petroni, et al. The web is your oyster-knowledge-intensive nlp against a very large web corpus. *arXiv*, 2021.

[145] L. Qian, W. Zhou, et al. Fino1: On the transferability of reasoning enhanced llms to finance. *arXiv*, 2025.

[146] Y. Qian, S. Zhang, et al. Enhancing llm-as-a-judge via multi-agent collaboration. 2025.

[147] Y. Qu, Y. Ding, et al. Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering. *arXiv*, 2020.

[148] R. Rafailov, A. Sharma, et al. Direct preference optimization: Your language model is secretly a reward model. *NeurIPS*, 2023.

[149] S. Rasal. Llm harmony: Multi-agent communication for problem solving. *arXiv*, 2024.

[150] R. Rawat. Disasterqa: A benchmark for assessing the performance of llms in disaster response. *arXiv*, 2024.

[151] O. Rubin, J. Herzig, et al. Learning to retrieve prompts for in-context learning. In *NAACL*, 2022.

[152] S. Saha, X. Li, et al. Learning to plan & reason for evaluation with thinking-llm-as-a-judge. *arXiv*, 2025.

[153] S. Samsi, D. Zhao, et al. From words to watts: Benchmarking the energy costs of large language model inference. In *HPEC*, 2023.

[154] P. Sarthi, S. Abdullah, et al. Raptor: Recursive abstractive processing for tree-organized retrieval. In *ICLR*, 2024.

[155] N. Saunshi, N. Dikkala, et al. Reasoning with latent thoughts: On the power of looped transformers. *arXiv*, 2025.

[156] T. Savage, A. Nayak, et al. Diagnostic reasoning prompts reveal the potential for large language model interpretability in medicine. *NPJ Digital Medicine*, 2024.

[157] M. Sawiński, K. Węcel, et al. Openfact at checkthat! 2023: head-to-head gpt vs. bert-a comparative study of transformers language models for the detection of check-worthy claims. In *CEUR*, 2023.

[158] J. Schulman et al. Proximal policy optimization algorithms. *arXiv*, 2017.

[159] B. Sel, A. Al-Tawaha, et al. Algorithm of thoughts: Enhancing exploration of ideas in large language models. *arXiv*, 2023.

[160] R. Shao, J. He, et al. Scaling retrieval-based language models with a trillion-token datastore. *NeurIPS*, 2024.

[161] Z. Shao, P. Wang, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv*, 2024.

[162] L. Shen, E. Shen, et al. Towards natural language interfaces for data visualization: A survey. *TVCG*, (6), 2022.

[163] W. Shen, X. Zhang, et al. Improving reinforcement learning from human feedback using contrastive rewards. *arXiv*, 2024.

[164] X. Shen, Y. Wang, et al. Efficient reasoning with hidden thinking. *arXiv*, 2025.

[165] N. Shinn, F. Cassano, et al. Reflexion: Language agents with verbal reinforcement learning. *NeurIPS*, 2023.

[166] K. Shridhar, A. Stolfo, et al. Distilling reasoning capabilities into smaller language models. *ACL*, 2023.

[167] R. Singhal, P. Patwa, et al. Evidence-backed fact checking using rag and few-shot in-context learning with llms. *arXiv*, 2024.

[168] C. Snell, J. Lee, et al. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv*, 2024.

[169] M. Song, M. Zheng, et al. Can many-shot in-context learning help llms as evaluators? a preliminary empirical study. *arXiv9*, 2024.

[170] J. Tack et al. Llm pretraining with continuous concepts. *arXiv*, 2025.

[171] Z. Tan, A. Beigi, et al. Large language models for data annotation: A survey. *arXiv*, 2024.

[172] Z. Tan, J. Peng, et al. Tuning-free accountable intervention for llm deployment–a metacognitive approach. *arXiv*, 2024.

[173] Z. Tan, J. Yan, et al. In prospect and retrospect: Reflective memory management for long-term personalized dialogue agents, 2025.

[174] Z. Tan and C. a. Zhao. Glue pizza and eat rocks-exploiting vulnerabilities in retrieval-augmented generative models. In *EMNLP*, 2024.

[175] L. Tang, P. Laban, et al. Minicheck: Efficient fact-checking of llms on grounding documents. *arXiv*, 2024.

[176] G. Team, P. Georgiev, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv*, 2024.

[177] K.-T. Tran, D. Dao, et al. Multi-agent collaboration mechanisms: A survey of llms. *arXiv*, 2025.

[178] A. Verma, S. Mohajer, et al. Multi-agent fact checking. *arXiv*, 2025.

[179] X. Wan, R. Sun, et al. Teach better or show smarter? on instructions and exemplars in automatic prompt optimization. *NeurIPS*, 2025.

[180] X. Wan, H. Zhou, et al. From few to many: Self-improving many-shot reasoners through iterative optimization and generation. *arXiv*, 2025.

[181] B. Wang, W. Ping, et al. Shall we pretrain autoregressive language models with retrieval? a comprehensive study. In *EMNLP*, 2023.

[182] B. Wang, W. Ping, et al. Instructretro: instruction tuning post retrieval-augmented pretraining. In *ICML*, 2024.

[183] H. Wang, W. Huang, et al. Unims-rag: A unified multi-source retrieval-augmented generation for personalized dialogue systems. *arXiv*, 2024.

[184] L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin, W. X. Zhao, Z. Wei, and J. Wen. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6), Mar. 2024.

[185] Q. Wang, L. Ding, et al. Recursively summarizing enables long-term dialogue memory in large language models. *arXiv*, 2023.

[186] S. Wang, Z. Chen, et al. Mixture of demonstrations for in-context learning. *NeurIPS*, 2025.

[187] X. Wang, M. Salmani, et al. Beyond the limits: A survey of techniques to extend the context length in large language models. *arXiv*, 2024.

[188] X. Wang, P. Sen, et al. Adaptive retrieval-augmented generation for conversational systems. *arXiv*, 2024.

[189] X. Wang, J. Wei, et al. Self-consistency improves chain of thought reasoning in language models. *arXiv*, 2022.

[190] X. Wang and D. Zhou. Chain-of-thought reasoning without prompting. *arXiv*, 2024.

[191] Y. Wang, Q. Liu, et al. Thoughts are all over the place: On the underthinking of o1-like llms. *arXiv*, 2025.

[192] J. Wei, N. Kim, et al. Inverse scaling can become u-shaped. In *EMNLP*, 2023.

[193] J. Wei, Y. Tay, et al. Emergent abilities of large language models. *arXiv*, 2022.

[194] J. Wei, X. Wang, et al. Chain-of-thought prompting elicits reasoning in large language models. *NeurIPS*, 2022.

[195] T.-R. Wei, H. Liu, et al. A survey on feedback-based multi-step reasoning for large language models on mathematics. *arXiv*, 2025.

[196] X. Wei and L. Liu. Are large language models good in-context learners for financial sentiment analysis? *arXiv*, 2025.

[197] Y. Wei, O. Duchenne, et al. Swe-rl: Advancing llm reasoning via reinforcement learning on open software evolution. *arXiv*, 2025.

[198] L. Weng. Llm-powered autonomous agents. *lilianweng.github.io*, Jun 2023.

[199] Y. Weng, B. Li, et al. Large language models with holistically thought could be better doctors. In *NLPCC*, 2024.

[200] R. Wolfe and T. Mitra. The impact and opportunities of generative ai in fact-checking. In *FAccT*, 2024.

[201] C. Wu, S. Yin, et al. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv*, 2023.

[202] X. Wu, L. Xiao, et al. A survey of human-in-the-loop for machine learning. *Futur. Gener. Comput. Syst.*, 2022.

[203] Y. Wu, Z. Sun, et al. Inference scaling laws: An empirical analysis of compute-optimal inference for problem-solving with language models. *arXiv*, 2024.

[204] Y. Wu, Y. Wang, et al. When more is less: Understanding chain-of-thought length in llms. *arXiv*, 2025.

[205] Y. Xia, Y. Huang, et al. A question and answering service of typhoon disasters based on the t5 large language model. *IJGI*, 2024.

[206] C. Xiang, T. Wu, et al. Certifiably robust rag against retrieval corruption. *arXiv*, 2024.

[207] Z. Xiang, F. Jiang, et al. Badchain: Backdoor chain-of-thought prompting for large language models. *arXiv*, 2024.

[208] Z. Xiang, Z. Xiong, et al. Cbd: A certified backdoor detector based on local dominant probability. *NeurIPS*, 2023.

[209] Y. Xiao, E. Sun, et al. Tradingagents: Multi-agents llm financial trading framework. *arXiv*, 2024.

[210] T. Xie, Z. Gao, et al. Logic-rl: Unleashing llm reasoning with rule-based reinforcement learning. *arXiv*, 2025.

[211] Y. Xie, A. Goyal, et al. Monte carlo tree search boosts reasoning via iterative preference learning. *arXiv*, 2024.

[212] G. Xiong, Q. Jin, et al. Benchmarking retrieval-augmented generation for medicine. In *ACL*, 2024.

[213] L. Xiong, C. Xiong, et al. Approximate nearest neighbor negative contrastive learning for dense text retrieval. *arXiv*, 2020.

[214] F. Xu, W. Shi, et al. Recomp: Improving retrieval-augmented lms with compression and selective augmentation. *arXiv*, 2023.

[215] P. Xu, W. Ping, et al. Retrieval meets long context large language models. In *ICLR*, 2023.

[216] P. Xu, W. Ping, et al. Chatqa 2: Bridging the gap to proprietary llms in long context and rag capabilities. *arXiv*, 2024.

[217] Z. Xu, C. Yu, F. Fang, Y. Wang, and Y. Wu. Language agents with reinforcement learning for strategic play in the werewolf game, 2024.

[218] A. Yang, B. Yang, et al. Qwen2. 5 technical report. *arXiv*, 2024.

[219] W. Yang, X. Bi, et al. Watch out for your agents! investigating backdoor threats to llm-based agents. *NeurIPS*, 2024.

[220] W. Yang, S. Ma, et al. Towards thinking-optimal scaling of test-time compute for llm reasoning. *arXiv*, 2025.

[221] S. Yao, D. Yu, et al. Tree of thoughts: Deliberate problem solving with large language models. *NeurIPS*, 2023.

[222] J. Ye, Z. Wu, et al. Compositional exemplars for in-context learning. In *ICML*, 2023.

[223] Y. Ye, Z. Huang, et al. Limo: Less is more for reasoning. *arXiv*, 2025.

[224] Z. Yi, J. Ouyang, et al. A survey on recent advances in llm-based multi-turn dialogue systems. *arXiv*, 2024.

[225] K. Yin, C. Liu, et al. Crisissense-llm: Instruction fine-tuned large language model for multi-label social media text classification in disaster informatics. *arXiv*, 2024.

[226] Y. Yu, H. Li, Z. Chen, Y. Jiang, Y. Li, D. Zhang, R. Liu, J. W. Suchow, and K. Khashanah. Finmem: A performance-enhanced llm trading agent with layered memory and character design. In *Proceedings of the AAAI Symposium Series*, number 1, pages 595–597, 2024.

[227] Y. Yu, Z. Yao, et al. Fincon: A synthesized llm multi-agent system with conceptual verbal reinforcement for enhanced financial decision making. *NeurIPS*, 2024.

[228] Z. Yuan, H. Yuan, et al. Scaling relationship on learning mathematical reasoning with large language models. *arXiv*, 2023.

[229] Z. Yue, H. Zhuang, et al. Inference scaling for long-context retrieval augmented generation. In *ICLR*, 2025.

[230] R. Zamora-Resendiz, I. Khurram, et al. Towards maps of disease progression: Biomedical large language model latent spaces for representing disease phenotypes and pseudotime. *medRxiv*, 2024.

[231] E. Zelikman, G. R. Harik, et al. Quiet-star: Language models can teach themselves to think before speaking. In *COLM*, 2024.

[232] L. Zhang, A. Hosseini, et al. Generative verifiers: Reward modeling as next-token prediction. *arXiv*, 2024.

[233] Q. Zhang, F. Lyu, Z. Sun, L. Wang, W. Zhang, W. Hua, H. Wu, Z. Guo, Y. Wang, N. Muennighoff, et al. A survey on test-time scaling in large language models: What, how, where, and how well? *arXiv preprint arXiv:2503.24235*, 2025.

[234] X. Zhang, A. Lv, et al. More is not always better? enhancing many-shot in-context learning with differentiated and reweighting objectives. *arXiv*, 2025.

[235] Y. Zhao and P. a. Singh. Optimizing llm based retrieval augmented generation pipelines in the financial domain. In *NAACL*, 2024.

[236] C. Zheng, Y. Gao, et al. Cape: Context-adaptive positional encoding for length extrapolation. *arXiv*, 2024.

[237] C. Zheng, Y. Gao, et al. Dape: Data-adaptive positional encoding for length extrapolation. *NeurIPS*, 2024.

[238] L. Zheng, W.-L. Chiang, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *NeurIPS*, 2023.

[239] L. Zheng, W.-L. Chiang, et al. Lmsys-chat-1m: A large-scale real-world llm conversation dataset. *arXiv*, 2023.

[240] C. Zhou, P. Liu, et al. Lima: Less is more for alignment. *NeurIPS*, 2023.

[241] H. Zhou, K.-H. Lee, et al. Trustrag: Enhancing robustness and trustworthiness in rag. *arXiv*, 2025.

[242] J. Zhou, Y. Zhang, et al. Synthetic lies: Understanding ai-generated misinformation and evaluating algorithmic and human solutions. In *CHI*, 2023.

[243] F. Zhu, Z. Liu, et al. Tat-llm: A specialized language model for discrete reasoning over financial tabular and textual data. In *ICAIF*, 2024.

[244] H. Zou, Q. Zhao, et al. Genainet: Enabling wireless collective intelligence via knowledge transfer and reasoning. *arXiv*, 2024.

[245] K. Zou, M. Khalifa, et al. Retrieval or global context understanding? on many-shot in-context learning for long-context evaluation. *arXiv*, 2024.

Topological Data Analysis Applications in Natural Language Processing: A Survey

Adaku Uchendu
MIT Lincoln Laboratory
MA, USA
adaku.uchendu@ll.mit.edu

Thai Le
Indiana University
IN, USA
tle@iu.edu

ABSTRACT

The surge of data available on the Internet has driven the adoption of a wide range of computational methods for analyzing and extracting insights from large-scale data. Among these, Machine Learning (ML) has become a central paradigm, offering powerful tools for pattern discovery, prediction, and representation learning across many domains. At the same time, real-world data often exhibit properties such as noise, imbalance, sparsity, limited supervision, and high dimensionality, motivating the use of additional analytical perspectives that can complement standard ML pipelines. One such perspective is Topological Data Analysis (TDA), a statistical framework that focuses on the intrinsic shape and structural organization of data. Rather than replacing ML, TDA offers a complementary lens for characterizing geometric and topological properties that may be difficult to capture with conventional feature-based or purely predictive approaches. This has motivated a growing body of work that integrates TDA into ML workflows, particularly in settings where data structure plays an important role. Despite this promise, TDA has received relatively limited attention in Natural Language Processing (NLP) compared to domains with more overt structural regularities, such as computer vision. Nevertheless, a dedicated community of researchers has explored its use in NLP, leading to **137 papers** that we comprehensively survey in this work. We organize these studies into theoretical and non-theoretical approaches. Theoretical approaches use topology to explain linguistic phenomena, whereas non-theoretical approaches incorporate TDA into ML-based pipelines through a variety of numerical representations. We conclude by discussing the key challenges and open questions that continue to shape this emerging area. Resources and a list of papers are available at: `https://github.com/AdaUchendu/AwesomeTDA4NLP`[1].

[1]DISTRIBUTION STATEMENT A. Approved for public release. Distribution is unlimited. This material is based upon work supported by the Department of the Air Force under Air Force Contract No. FA8702-15-D-0001 or FA8702-25-D-B002. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of the Air Force. © 2026 Massachusetts Institute of Technology. Delivered to the U.S. Government with Unlimited Rights, as defined in DFARS Part 252.227-7013 or 7014 (Feb 2014). Notwithstanding any copyright notice, U.S. Government rights in this work are defined by DFARS 252.227-7013 or DFARS 252.227-7014 as detailed above. Use of this work other than as specifically authorized by the U.S. Government may violate any copyrights that exist in this work.

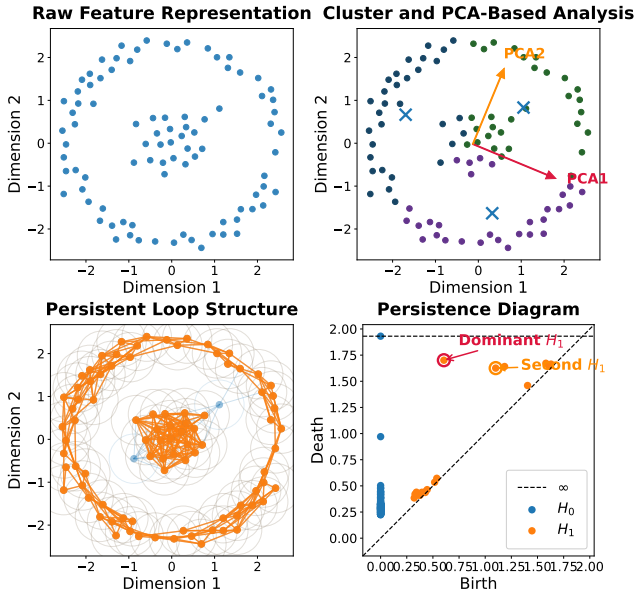

Figure 1: The raw point cloud contains a dense central region together with loop-like outer structure. Conventional ML analysis such as PCA and clustering is strongly influenced by variance and local density, so much of the summary is affected by the central cluster. In contrast, TDA emphasizes global structure. In the persistence diagram, blue points represent *connected component* features and orange points represent *loop* features. For each topological feature, the horizontal coordinate ("birth") is the scale at which the feature first appears, and the vertical coordinate ("death") is the scale at which it disappears. Here, the prominent orange points that are further from the diagonal correspond to the persistent loop structures visible in the third panel.

1. INTRODUCTION

Proliferation of the Internet has given rise to the generation of massive amounts of data. These massive amounts of data when processed can solve many crucial issues plaguing our current society. Due to this well-established notion among stake-holding institutions, the Machine Learning (ML) field has been thriving as a tool that extracts trends and solutions to non-trivial problems. However, real-world data tends to be noisy, heterogeneous, imbalanced, have missing labels, contain high-dimensionality, etc., often making the

adoption of ML techniques to such datasets non-trivial. Therefore, to extract meaningful findings from data, specifically real-world data, clever techniques that extract additional features, while *preserving the overall structure of the data* need to be employed.

To that end, a small niche community for *Topological Data Analysis (TDA) applications in NLP* has emerged. Being promised as a technique that can extract and analyze the shape/topology of data, TDA has great potential in mitigating such issues witnessed in real-world data. Thus, by applying TDA to NLP, we obtain "*topological structures from language*," which refers not to intrinsic properties of raw text itself, but to the structures that emerge when linguistic data is mapped into high-dimensional embedding spaces. These induced topologies capture relationships among words, sentences, or documents based on their learned representations, rather than any inherent topological features of the text.

TDA is a "collection of powerful tools that can quantify shape and structure in data"[2] and is inspired by the algebraic topology and geometry mathematical fields. The benefits of TDA are vast, including the ability to extract additional features that are typically not captured by other feature extraction techniques [157, 96, 107]. These features are known as *topological features*. Unsurprisingly, since TDA is used to capture topological features, it has been applied to many tasks where data has distinct graphical structures [107, 63]. These include tasks that have obvious graph-like structures, such as protein classification [33, 82, 158] and drug discovery [2]; to those that are not so obvious, such as diabetes classification [166, 140], image classification [66, 151], and time series analysis [110, 155, 53]. However, since the shape of a text is not apparent, it has not gained as much attention in NLP as it has in the Computer Vision field [66, 151] specifically in the Medical domains [139, 102]. Still, several researchers have found ways to extract unique, global-level features using TDA. Other typical numerical representation techniques in text are unable to extract global-level features, making TDA suitable for the task.

Figure 1 illustrates a toy example that shows the utility of TDA to standard ML analysis. Conventional ML methods often summarize data through variance, centroids, or local grouping, which means dense regions can disproportionately shape the result. In the example, the central cluster absorbs much of the variance, while the outer loop structure plays a less visible role in PCA- and cluster-based summaries. TDA instead tracks topological features such as connected components and loops across multiple scales, allowing it to recover meaningful global structure.

More broadly, TDA aims to answer the central research question - *what is the true shape of data?* We survey **137 papers** that have attempted to find an answer through various approaches. The first application of TDA in NLP was published in 2012 [165], and since then, there have been over 100 papers applying TDA in NLP. There have been a gradual acceleration in the number of published works in TDA applications on various NLP tasks, including ones pertaining to the recent emergence of Large Language Models (LLMs) such as hallucination detection [14, 128], mechanistic interpretability [174, 120], and model efficiency [50, 98]; we project that this trend will continue in the future (Figure 3). Therefore, based on these approaches, we categorize these applications into two - *theoretical* [74, 113] and *non-theoretical* [187, 35] approaches. *Theoretical* approaches involve using TDA to explain linguistic phenomena by probing the topological space, shape, and evolution of topics. On the other hand, *Non-theoretical* approaches mainly discuss how to effectively apply existing numerical representation techniques in NLP to extract novel topological features with TDA.

[2] https://www.indicative.com/resource/topological-data-analysis/

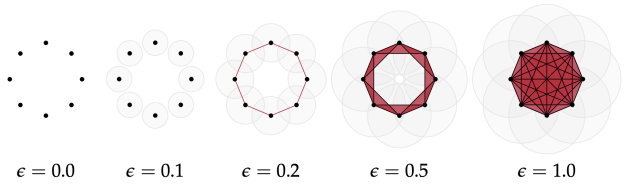

Figure 2: Illustration of the Persistent Homology technique using different radii to find the persistent features [123]. ϵ is the ball diameter.

In addition, we observe that theoretical approaches only span 13 papers, while non-theoretical approaches have over 100 papers. Due to the higher number of non-theoretical approaches, we discuss several categories that could be useful for distinguishing applications: numerical representation, tasks, TDA technique, data modality, and learning type. TDA techniques (i.e., Persistent Homology and Mapper), data modality (i.e., text and speech), and learning type (i.e., unsupervised and supervised) are binary, making it difficult to meaningfully grasp distinctness from almost 120 papers. However, with tasks which refer to the problems in which approaches are adopted for; these have seven categories - *(1) classification, (2) clustering & topic modeling, (3) sentiment & semantic analysis, (4) structure & visualization, (5) health, social, & scholarly analysis, (6) speech processing,* and *(7) model interpretation & analysis*. We observe that classification and model interpretation & analysis are the most popular tasks explored by researchers. In addition, numerical representations leveraged in non-theoretical application include - *(1) TF-IDF, (2) Word2Vec, (3) GloVe, (4) FastText, (5) ELMo, (6) Transformers, (7) Symbolic,* and *(8) Multi-Modal*. We use numerical representation as our main taxonomy for non-theoretical applications because it is the bottleneck for extracting topological features from text.

Finally, we will first discuss the principles behind TDA and the two main techniques employed for TDA feature extraction: *Persistent Homology* and *Mapper*. In addition, we will discuss the selection criteria and taxonomy development of the survey, both the theoretical and non-theoretical approaches, and discuss interesting findings, open problems, and future directions.

2. TOPOLOGICAL DATA ANALYSIS

Topology is defined as *"the study of geometric properties and spatial relations unaffected by the continuous change of shape or size of figures,"* (Oxford Dictionary). TDA is then a collection of powerful techniques that can quantify the shape and structure of data [100]. Two main techniques are used to extract TDA features: *Persistent Homology* and *Mapper*.

2.1 Persistent Homology

Persistent Homology (PH) [38] is the most popular TDA technique. It uses algebraic topology methods to extract topological signatures at different spatial dimensions. This process involves representing data as a point cloud and performing deformation or perturbation processes to extract the true "shape" of data after the noise has been removed. To achieve this, PH employs Vietoris-Rips complex [100]. Vietoris-Rips complex is a way to build simplicial com-

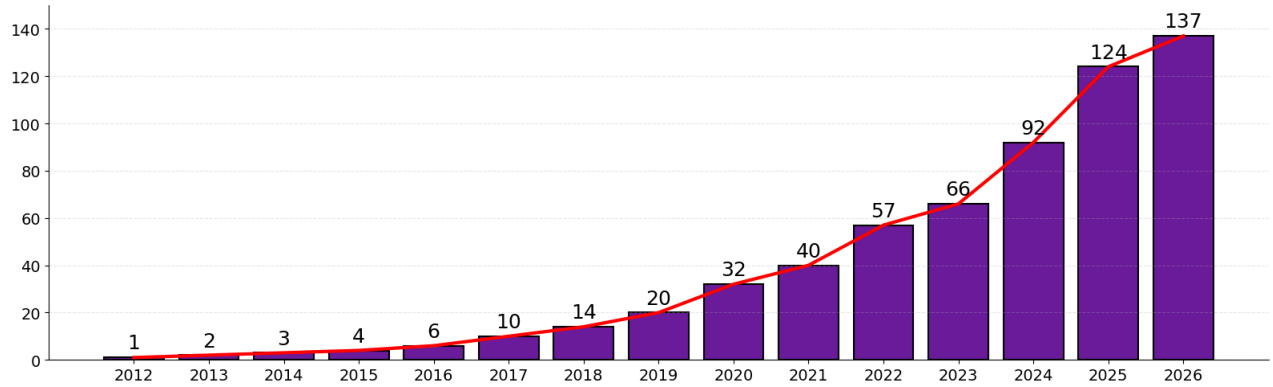

Figure 3: Cumulative number of TDA papers for NLP published from 2012 to April-2026.

plexes which are used to represent data in a topological space. A simplicial complex is a topological space built by putting points, lines, and higher dimensional shapes together. These formations reveal features that are holes in different dimensions, represented as *betti numbers* (β_d, d-dimension). Holes in the 0-dimension (β_0) is represented as one vertex, 1-dimension (β_1) is represented as an edge, 2-dimension (β_2) is represented as a triangle. Further, these features are called connected components, loops/tunnels, and voids, respectively.

Using the method described above, data is represented as a point cloud, and circles are drawn around each point. Next, the radius of each circle is increased using a defined range of points, such that if the circles get bigger and touch, one of the points disappears and this is recorded as a $death$. Additionally, this process of perturbation in different dimensions can cause the $birth$ of a new hole, which is also recorded. Persistence is defined as the length of time it took a feature to disappear or die ($death - birth$). The $death$ is recorded with the radius value at which the points overlap. Lastly, TDA features are typically visualized in a persistence diagram, which is a visual representation of the $birth$ (x-axis) vs. $death$ (y-axis) features. Other ways of visualizing TDA features include persistence images [1] and barcode plots [54]. Figure 2 illustrates an example of the process of extracting TDA features using PH. In terms of application, PH has been used to extract novel features to complement existing NLP representations and improve various classification performances [35, 157, 170].

Persistent Homology. *This is a TDA technique that studies the deformation of "holes" in different dimensions. Using PH, we can track when features appear and disappear and visualize these features, usually in a persistence diagram. This process allows us to find the true structure of data, typically devoid of noise.*

2.2 Mapper

Mapper is a dimension reduction clustering technique for visualizing TDA-extracted topological structures/signatures. It was proposed by Singh et al. [138] and has been used extensively to visualize topological structures in data to create visually pleasing figures. In addition, Mapper figures have been used to interpret model performance through data probing [23]. The Mapper algorithm works

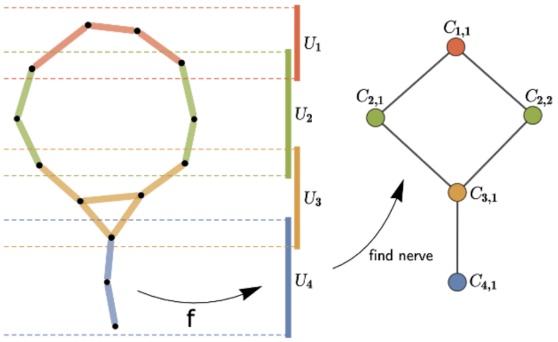

Figure 4: Illustration of Mapper from [101]. The filter function f is a height function, which is a projection onto the y-axis. The cover of the projected space is the four intervals U_i. The Mapper graph on the right is a result of applying the rest of the Mapper algorithm and clustering each preimage in the nearest neighbor.

in four steps[3] (Figure 4) following the instructions of [101]: (1) Transform the data to a lower-dimensional space using a filter function f, also known as a lens. This implies projecting from one space to another. Options for filter functions include PCA [93], UMAP [94], and any other dimension-reduction algorithms; (2) Create a cover $(U_i)_{i \in I}$ for the projected space, which is typically composed of overlapping intervals with a constant length; (3) Cluster the points in the preimage $f^{-1}(U_i)$ into sets $C_{i,1}, \ldots, C_{i,k_i}$ per interval U_i; (4) Create a graph where each vertex represents a cluster set. There is an edge between two vertices if the corresponding clusters share common points. Points in the same neighborhood are clustered using a defined clustering technique, such as DBSCAN [42] to change a cluster of several points into a node of a graph.

The intrinsic nature of the Mapper algorithm makes it advantageous in preserving structure, even with mapping from one dimension to another. Furthermore, the clustering techniques allow it to be used to explain model performance as the clusters and colors have meaning that can be further explored. Finally, Mapper is more useful for exploratory data analysis, while PH is more useful for analyzing point clouds and examining the persistence of features. In this survey, we will discuss how several researchers use Mapper to explain or enhance several phenomena in NLP tasks.

[3] `https://www.quantmetry.com/blog/topological-data-analysis-with-mapper/`

Mapper. *This is a TDA technique that visualizes the graphical representation of data in order to capture the intrinsic structure. It is very useful for preserving data structure and creating visually pleasing plots, which can be investigated manually to find insights.*

3. SURVEY SCOPE

3.1 Selection Criteria

In order to find all NLP papers that applied TDA, we manually searched on Google Scholar using key terms such as *text mining persistent homology, language model topological data analysis*, etc., checking related articles of the relevant papers, their cited papers, and different combinations of all three methods. After, obtaining over 60 papers initially, we started creating a taxonomy and categorizing the papers. Initially, we focused on TDA applications in textual data, but as we searched, we found several applications in speech, and collected such papers. Finally, we removed papers that did not apply TDA to text or human speech data. Papers removed, either applied TDA to a graphical representation of reddit social networks, applied non-TDA topological techniques, or applied TDA to non-speech audio data. Using these criteria we selected only papers that fit schema and collected the rest following the same schema.

3.2 Taxonomy Development

Based on the papers selected for the survey, we were able to categorize the applications of these papers into two approaches - *theoretical* and *non-theoretical* applications:

Theoretical applications of TDA in NLP: These focus on understanding, characterizing, or proving properties of language and its representations through the lens of topology. They are less about immediate performance gains and more about insight. This application aims to answer the question - "What do the shapes of embedding spaces tell us about language itself and our models of it?" Example - Analyzing embedding spaces: Using persistent homology to study whether semantic clusters, or syntactic structures, correspond to stable topological features, and finding out what that tells us about language.

Non-theoretical (practical) applications of TDA in NLP: These treat TDA as a tool for solving tasks, regardless of whether deeper linguistic/topological insights are obtained. The emphasis is on utility. This application aims to answer the question - "How can topological summaries directly help with applied NLP tasks?" Example - Feature engineering: Augmenting classifiers for sentiment analysis, topic detection, or authorship attribution with topological signatures.

4. THEORETICAL APPROACHES

Since the field of NLP is very interested in representing and analyzing texts or speech in meaningful ways, several theoretical approaches have been proposed to investigate how well these approaches align with linguistic principles. Thus to explain or confirm linguistic phenomena within the NLP paradigm, a few researchers have proposed topological approaches for probing data. See Figure 5 for an illustration of this pipeline. By employing TDA techniques - Persistent homology or Mapper to probe for linguistic phenomena, researchers aim to capture the *topological space* (both *semantic* and *syntactic* relationships) in language, analyze and visualize the *topology of topic evolution* within texts, and extract the *topological shape* of words. See Table 1 for the theoretical approaches and Figure 6 for the flowchart illustrating the taxonomy of theoretical applications of TDA in NLP tasks. In essence, these theoretical topological methods provide a conceptual bridge between linguistic theory and mathematical topology.

4.1 Topological Space

4.1.1 Semantic Topological Space

A **semantic topological space** is a conceptual framework used to represent and analyze the relationships between the meanings (semantics) of words, phrases, or other linguistic units in a topological or shape structure. This representation involves mapping these units into a mathematical space where the distance or structure between them reflects semantic similarity or other relationships (i.e., Euclidean space $\rightarrow$ Topological space).

Karlgren et al. [74] visualizes the topological semantic space of text using Mapper, which identifies the topical density of the space. To capture topological properties, they train two semantic spaces in a specific topical domain [74]. One space was trained only on articles of similar topics, and the other on introductory paragraphs of those same articles. Findings reveal that clusters of main concepts remained close for the space trained only on articles of similar topics. For the other topological space, the main concepts were randomly distributed [74]. This suggests that semantic topological space can be better captured with richer and denser data than with sparser data.

In addition, Cavaliere et al. [24] extracts main concepts from the texts by probing the context-aware semantic topological space built with simplicial complexes. Gromov et al. [58] builds a semantic space with bigrams and trigrams of Word2vec embeddings of English and Russian languages to ascertain how distinct the languages are. Next, they use these findings for bot detection. Sakib et al. [126] proposes a metric - *S M Nazmuz Sakib Topological Affix Isometry Index* to measure the structural preservation of the semantic space after the addition of affixes (i.e., -ness, -er, un-). They use persistent homology to create word manifolds to measure how affixes preserve (i.e., isometry) or distorts the meanings and relationships of the words it attaches to. Christianson et al. [29] uses persistent homology to structure the semantic networks from mathematical concepts in college-level linear algebra texts. They find that the networks show strong core-periphery architecture, where concepts are dense and sparse periphery for concepts presented throughout. Their results could inform the optimal design of principles for textbooks.

Furthermore, Wagner et al. [165] uses TF-IDF to numerically represent the top 10-50 words in a corpus and build a topological space that analyzes the structure of similarities within several documents. This topological space is built using discrete Morse theory and persistent homology to find meaningful topological patterns [165]. However, in 2012, they found that their technique was unsuccessful due to computational costs, which is a testament to how the field of NLP has improved so that we now have more tractable solutions, such as dimensionality reduction algorithms [94], and TDA packages (Ripser [12], Sklearn-TDA [133], PHAT [13], pytorch-topological[4]), as well as compute resources to efficiently construct topological spaces from large or complex data.

[4] `https://github.com/aidos-lab/pytorch-topological`

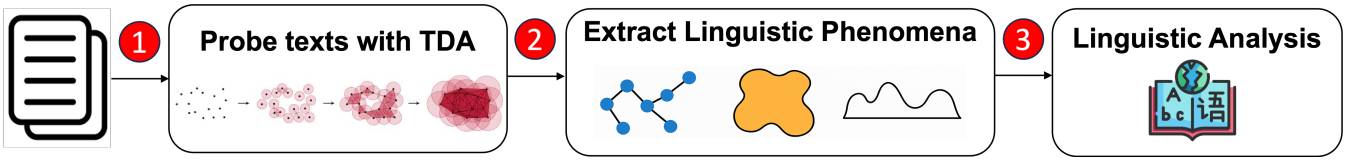

Figure 5: Illustration of the theoretical approaches researchers have employed to (1) probe texts, (2) extract TDA features, (3) use these features to explain or confirm known linguistic phenomena.

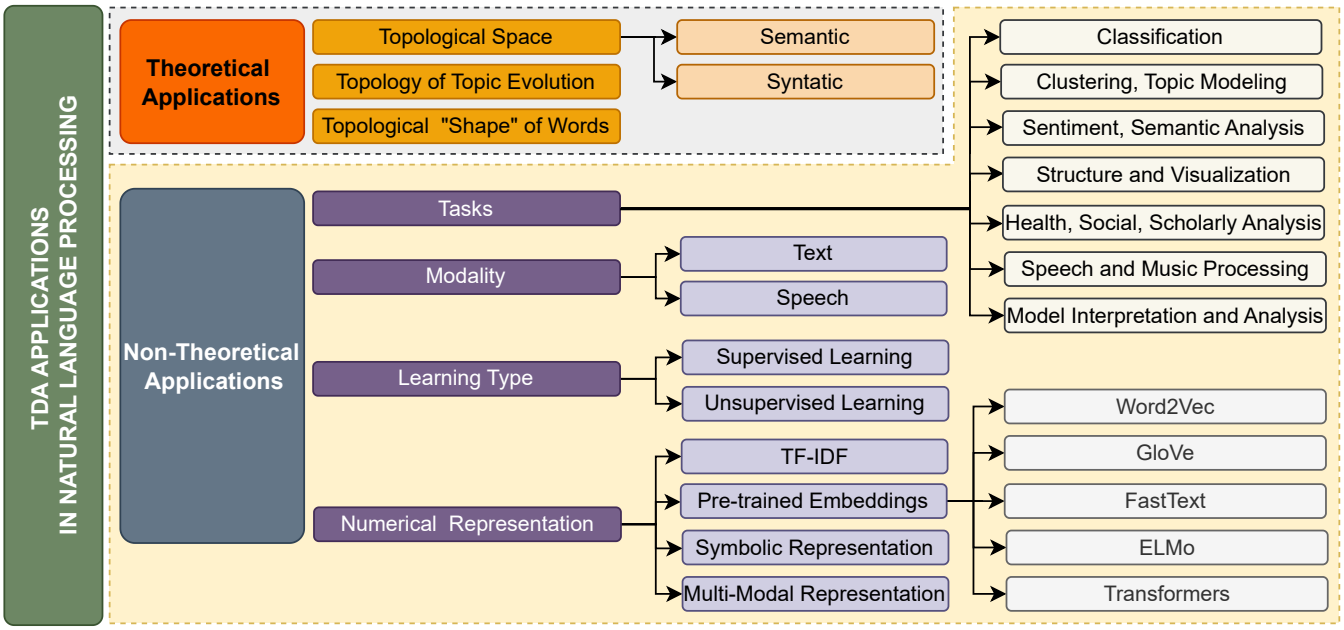

Figure 6: Taxonomy of Topological Data Analysis (TDA) for Natural Language Processing (NLP) Applications

Table 1: Theoretical Applications of TDA in NLP

Name	Category	Task	TDA Technique
[74]	TS-Sem	Identify topical density of the space	Mapper
[24]	TS-Sem	Extracts main concepts from text	Persistent Homology
[165]	TS-Sem	Analyzes document similarities	Persistent Homology
[58]	TS-Sem	Analyzes holes in English and Russian languages	Persistent Homology
[126]	TS-Sem	Measures how word affixes (i.e., -ness) distort or preserve semantic structure	Persistent Homology
[29]	TS-Sem	Creates a topological structure for college-level linear algebra texts	Persistent Homology
[113]	TS-Syn	Analyzes syntactic parameters of different language families	Persistent Homology
[114]	TS-Syn	Explains linguistic structures with homoplasy phenomena	Persistent Homology
[36]	TSW	Investigates the "shape" of language phylogenies in the Indo-European language family	Persistent Homology
[45]	TSW	Captures grammatical structure expressed by corpus using *word manifold*	Persistent Homology
[34]	TSW	Analyzes shapes of South American languages: Nuclear-Macro-Jê & Quechuan families	Persistent Homology
[20]	TSW	Analyzes topological similarity between Tifinagh and Phoenician scripts	Persistent Homology
[129]	TTE	Topic evolution within documents	Persistent Homology

Note: TS-Sem = Topological Space (Semantic); TS-Syn = Topological Space (Syntactic); TTE = Topology of Topic Evolution; TSW = Topological "Shape" of Words.

Insight (Semantic Space). *The semantic topological space is explored by researchers to identify semantic linguistic principles captured in texts through a topological lens. Most of the applications in this section involve understanding the semantic similarity between text pairs from a topological lens.*

4.1.2 *Syntactic Topological Space*

A **syntactic topological space** is a theoretical framework used to represent and analyze the relationships between syntactic structures from a topological perspective. This concept is particularly relevant in linguistics, where it helps model and understand the structural aspects of language, such as grammar, sentence construction, or the hierarchical organization of those linguistic units.

Therefore, Port et al. [113] analyzes how syntactic parameters are distributed over different language families, including Indo-

European, Niger-Congo, Austronesian, and Afro-Asiatic families. For instance, features in β_0 capture the subdivision into historical, and features in β_1 capture syntactic differences between branches of families of languages, as well as the syntactic influences between them [113]. They investigate the syntactic topological structures of language families, specifically Indo-European, Niger-Congo, Austronesian, and Afro-Asiatic families. Port et al. [113] shows that the three persistent connected components (β_0) in the Niger-Congo family represents its three subfamilies - Mande, Atlantic-Congo, and Kordofania. The syntactic topological structures of these languages also reveal the historical linguistic phenomena that the Hellenic branch played a role in the historical development of the Indo-European languages [113].

Similarly, Port et al. [114] probes the interpretability of the syntactic topological space by introducing *homoplasy* phenomena to explain persistent loops. Homoplasy phenomena in syntax are observed when dissimilar languages exhibit syntactic similarities [114]. Findings reveal that the Indo-European family languages - Czech, Lithuanian, Middle Dutch, and Swiss German have the same homoplasy phenomena [114] due to the appearance of persistent loops in these languages. This is because Middle Dutch and Swiss German are similar, while Czech and Lithuanian are so different from them, making the homoplasy phenomenon the most reasonable explanation [114].

Insight (Syntactic Space). *The syntactic topological space captures the syntactic structure of language (i.e., grammar, etc.) from a topological lens. Using this framework, researchers confirm linguistic phenomena in language families and subfamilies by exploring the syntactic relationship between languages. Thus, a novel application of this framework could include the discovery of new linguistic phenomena within syntactic structures.*

4.2 Topology of Topic Evolution

The **topology of topic evolution** refers to the study and representation of how topics, themes, or concepts develop and change over time within a given corpus of texts or discourses in a topological space/structure. This concept is particularly relevant in fields where understanding the temporal dynamics of topics can provide insights into trends, shifts in public opinion, or the development of scientific or cultural themes.

Sami et al. [129] utilizes TDA to visualize the relationship between words in a text block, words in a corpus, and text blocks in a corpus. Text blocks represent a chapter/section in a book, a document in a media corpus, and a webpage in a web corpus [129]. They visualize both local context (i.e., each text block in a set of sentences) and global context (i.e., occurrence of extracted words in the corpus) features. These features are extracted by using the circular topology to represent words. Then, the peripheral nature of the text block and corpus can be visualized using these features. With the Local context features, dimension reduction is achieved by stemming the prefixes and suffixes of words. For the Global context features, word movement is captured, which analyzes topic evolution. Finally, findings reveal that using the circular topology in 2D space, core words from the corpus stay close to the center, and the explanatory words remain close to the circle's periphery.

Insight (Topic Evolution). *Exploring the topology of topic evolution is a novel framework for capturing the topology of topics in a corpus. The findings suggest that this framework can be adopted to evaluate the utility of a summarization, paraphrasing, or obfuscating model predictions by comparing the topology of the topic evolution in the original vs. the perturbed texts.*

4.3 Topological "Shape" of Words

The **topological "shape" of words** is a conceptual framework in linguistics and cognitive science that explores the structural properties of *words*. This framework leverages ideas from topology to capture the true shape of words in a linguistically meaningful way. Thus, using topological methods such as TDA, the structural properties of words can be extracted and analyzed.

Draganov et al. [36] captures the "shape" of words for several languages by comparing the phylogenies or evolutionary history of language in the Indo-European language family. Initially, numerically representing the texts with FastText [17], they use persistent homology to construct language phylogenetic trees for over 81 Indo-European languages. Experiments reveal that: (1) the shape of the word embedding of a language carries historical and structural information, similar to Port et al. [113, 114]'s findings; and (2) TDA methods can successfully capture aspects of the shape of language [36].

Similar to [36, 113, 114], Dong et al. [34] extracts the topological shapes of languages. Specifically, South American languages - the Nuclear-Macro-Jê (NMJ) and Quechuan families using TDA. By using techniques like multiple correspondence analysis (MCA) for dimension reduction of the categorical-valued dataset and persistent homology, Dong et al. [34] visualizes each language in the selected families as a point cloud. This forms the topological shape of the South American languages, such that languages close together are more similar. By comparing the topological shapes of the languages, it is observed that there are major distinctions between the Jê-proper and the non-Jê-proper languages, as well as the northern and southern Quechuan languages [34].

Fitz et al. [45] introduces a novel terminology - *word manifold*, which is a simplicial complex, whose topological space captures grammatical structure expressed by the corpus. This is done by implementing a technique for generating topological structure directly from strings of words. Experiments reveal that the homotopy type of the word manifold is also influenced by linguistic structure [45]. Finally, Bouazzaoui et al. [20] explores the topological similarity of the shapes of two writing systems - Tifinagh and Phoenician scripts.

Insight (Shape of Words). *The topological "shape" of words is a concept that has interested several linguists, as it can be used to confirm and discover linguistic phenomena within languages. It is focused on capturing the shape of several languages. This concept combines all other frameworks like the semantic and syntactic topological spaces to capture a linguistically informed topological shape of words.*

5. NON-THEORETICAL APPROACHES

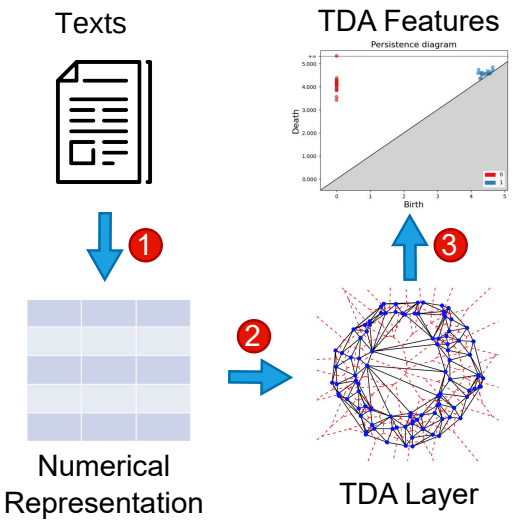

Figure 7: Illustration (inspired by [157]) of the Non-theoretical approach of using TDA as a feature extractor in NLP with three steps: (1)-extracting numerical representations, (2)-reformatting for TDA's inputs, and (3)-extracting TDA features.

There are several ways to categorize the applied/non-theoretical TDA applications in NLP. These applications can be categorized by *task*, *learning type*, *data modality*, *TDA technique*, and *numerical representation*. We observe that categorizing these TDA applications by task and numerical representation is more meaningful than the other categories, since those categories are binary and not very descriptive of the landscape. Out of these dimensions, the numerical representation showcases the bottleneck for extracting useful TDA features. See Figure 7 for an illustration of the pipeline for extracting TDA features from numerically represented texts. In addition, while we focus on both task and numerical representation, our main taxonomy for the non-theoretical applications is centered on how TDA features are extracted from different forms of numerical representations. Part of Figure 6 illustrates the taxonomy of non-theoretical applications of TDA in NLP tasks. See Table 3 and 4 in Appendix for the list of non-theoretical approaches.

Learning types have supervised [40, 83], and unsupervised [144, 18]; *Modality* has text [150, 75], and Speech [131]; and *TDA techniques*, have Persistent Homology [149, 27], and Mapper [64, 40]. For data modality, 90% of applications are concentrated in Text, and for TDA techniques 89% of applications are concentrated in persistent homology. In this survey, we focus on two other broad categories, *Tasks* and *Numerical Representation*, where the connections to prior work are richer.

5.1 Tasks

Tasks in this context are defined as the problem for which a solution is attempted. We categorize these NLP problems that TDA practitioners have attempted into seven categories below. We refer the readers to Figure 8 (left) for the distribution of the number of publications per task.

1. **Classification**: The most popular application is deepfake text detection [89, 154, 80, 79, 157, 168].
2. **Clustering and Topic Modeling**: The most popular application is document clustering and topic modeling [64, 59].
3. **Sentiment and Semantic Analysis**: The most popular applications are linguistic/grammatical acceptability [27, 72], word sense induction & disambiguation [121, 147], and polysemy word classification [73, 136].
4. **Structure and Visualization**: The most popular is using Mapper to visualize model hidden weights [48, 120].
5. **Health, Social, and Scholarly Analysis**: Since this is not a popular application for TDA, the most interesting applications are - prediction of epidemics [110] and categorization of lonely people [39].
6. **Speech Processing**: The most popular applications are studying vocalizations [19, 18].
7. **Model Interpretation and Analysis**: The most popular applications are model probing to reveal behavior in hidden weights [75, 57].

5.2 Numerical Representation

See Figure 8 (right) for the distribution of the number of applications for each numerical representation.

5.2.1 TF-IDF

TF-IDF (Term Frequency - Inverse Document Frequency) is a well-known statistical formula that calculates the importance of words relative to a corpus. A few works investigated the extraction of topological features from TF-IDF representations as part of the pipeline illustrated in Figure 7. For instance, SIFT, a persistent homology-based model with TF-IDF, is developed to differentiate between child and adolescent writings [187]. This model represented the TF-IDF features as a time series, and then extracted topological features to enhance text classification. Several other researchers applied this model to other *classification task*, such as deepfake text detection [89], presidential election speech attribution [68], distinguishing between languages by averaging the persistence landscapes [143], age group categorization of lonely people [39], and movie genre classification [35, 137]. Additionally, Elyasi et al. [40] compares the two popular TDA approaches - Persistent Homology and Mapper to classify Persian poems. Also, using Mapper for the *structure & visualization tasks*, Maadarani et al. [91] explains linguistic properties in poetry writing styles, and Van et al. [159] interprets NLP model behavior. Lastly, we observe applications in the *clustering & topic modeling task* - keyphrase extraction [59], text summarization [78], and twitter topic detection [149]; *sentiment & semantic analysis task* - legal entailment [134], and sentiment analysis of movie reviews [96].

5.2.2 Pre-trained Non-contextual Embeddings

Word2Vec Embeddings. Word2Vec embeddings are a type of word representation that allows words with similar meanings to have similar vector representations [97]. Thus, we observe applications in the *structure & visualization task*, where Haghighatkhah et al. [61] creates story trees to trace story lines. Next, we observe applications in *sentiment & semantic analysis task*, where TDA is applied to novel problems such as the creation of a topologically-enhanced search engine using Mapper [30], measuring distance between the literary style of Spanish poets - Francisco de Quevedo, Luis de Góngora, and Lope de Vega [105], detecting narrative shifts in media discourse [10], distinguishing news articles and poems by detecting plot holes [5], analysis of contradictions within texts [170], and word sense induction and disambiguation [121, 147]. For the *classification* task, researchers detect fraudulent papers [155], and topological loops in logical statements [156].

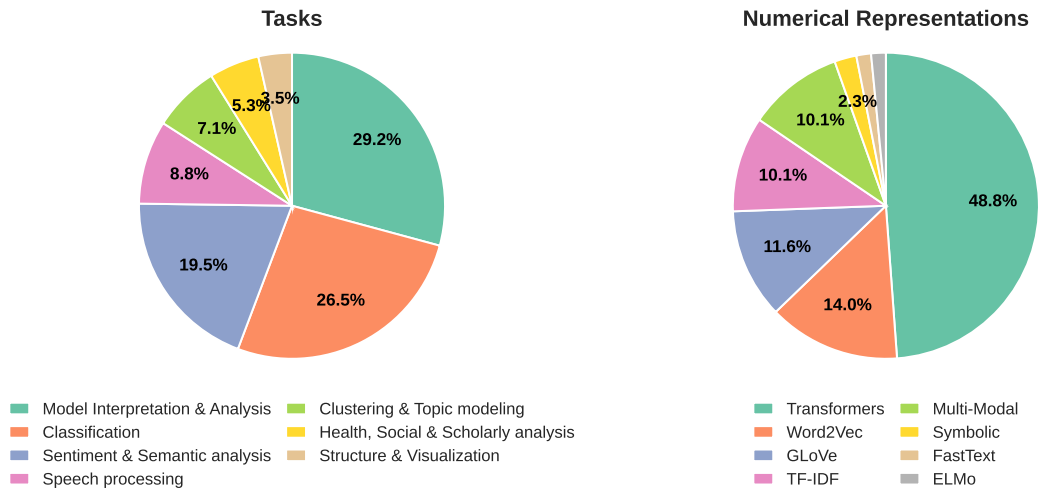

Figure 8: Distribution of the number of publication for each task (**Left**) and numerical representation type (**Right**).

Furthermore, we observe applications in the *health, social, and scholarly analysis* task - disease prediction from epidemic curves [110], and Yadav et al. [173] uses doc2vec to create top2vec (i.e., topological features $\rightarrow$ vec) of publication documents to use the holes found with persistent homology to determine missing documents vs. innovative research. Additionally, [64, 169] perform *topic modeling tasks*. Finally, for the *model interpretation and analysis* task, Feng et al. [44] uses both topological and geometrical features to investigate the quality of LLM-enhanced data augmentation, Sun et al. [145] derives the correlation between sentence vectors and their semantics, and Yessenbayev et al. [175, 176] compares the semantic alignment of text and speech embeddings for text-speech pairs.

GloVe Embeddings. GloVe or Global Vectors for Word Representation is another technique for numerically representing texts as embeddings. Topological features extracted from GloVe embeddings have been applied to the following tasks - *classification task*, which include author attribution of novelists [51], fake news detection [32], and deepfake text detection [89, 3]; *sentiment & semantic analysis task*, which include document categorization [52], and capturing circles in circular arguments [156, 181]; *model interpretation and analysis*, where both Haim et al. [62] and Michel et al. [96] compare text representations & embeddings, Spannaus et al. [144] explains model performance, and Zadrozny et al. [180] tests the manifestation of intelligence and understanding in models; and *health, social, and scholarly analysis task*, which include social anxiety detection [21], and keywords extraction of scholarly documents [103].

FastText Embeddings. FastText embeddings are built on the Word2Vec approach by incorporating subword information, improving the representation of rare words, and allowing for embedding out-of-vocabulary words [17]. This type of embedding is not a popular feature extractor that researchers employ to enhance topological features, as only two tasks are attempted - *sentiment & semantic analysis task*, which include polysemy word classification [73, 150, 136], and word sense induction & disambiguation [73]; and *text classification*, where Tymochko et al. [155] detects fraudulent papers.

5.2.3 *Pre-trained Contextual Embeddings*

Transformer Embeddings. Researchers have evaluated the strength of the TDA features extracted from Transformer-based [161] embeddings. Using the idea of self-attention, the neural network can encode more semantic and syntactic features than previous embeddings, which should allow for richer TDA features to be extracted. To incorporate TDA features for various tasks, several researchers have investigated the efficacy of using other outputs of encoder and decoder Transformer models - *CLS Embedding output*, *hidden weights*, and *attention weights* to extract high-quality additional features.

`CLS Embedding Output.` Researchers have applied these features on *text classification task*, specifically for deepfake text detection [154, 80, 168, 60], fake news detection [83], and TEDtalk public speaking ratings classification [31]. Additionally, we observe applications to the *topic modeling task* [22, 65]. Next, for the *model interpretation & analysis task*, Gourgoulia et al. [57] probes LLMs to estimate class separability of text datasets, and Proskura et al. [115] uses topological information from encoder models to select the best models to use when building an ensemble. In addition, Rair et al. [118] uses Mapper to visualize how fine-tuning of models like RoBERTa-Large restructures the embedding space into modular, non-convex regions to align with model predictions. This technique aims to visualize the geometry and topology of when annotators agree and disagree [118]. Furthermore, Arun et al. [6] employs TDA for the *structure & semantic task*, detecting controversial vs. non-controversial political discourse by capturing the shifts in discourse for controversial data. Similarly, Meng et al. [95] performed a *semantic task* of using persistent homology to augment and improve personalized web search. Next, Chandra et al. [25] performs a *health analysis* task using persistent homology to track mental health journeys in online communities. Finally, Rathore et al. [120] performs the *structure & visualization task* in combination with the *model interpretation & analysis task* by visualizing the training process of transformer-based models.

`Hidden Weights.` *Classification task* include deepfake text detection [157, 122], language translation [7], and code attribution [92]. Next, Garcia et al. [48] explores a combination of the *sentiment & semantic analysis* and *structure & visualization* tasks by using Mapper to visualize polysemous words in the hidden representations of the BERT transformer model. Bensalem et al. [15]

performs a *sentiment analysis* task by using the sentiment scores of original vs. translated texts extracted from a Transformer-based model. They represent these scores in a time series form and use zigzag persistent homology to detect sentiment shift in translated texts. Similarly, Goshev et al. [56] investigates the semantic topology of sentences encoded by transformer embeddings. Zhang et al. [184] uses persistent homology to improve text summarization by capturing the global structure of texts. Alexander et al. [4] combines the *health analysis* and *visualization* tasks to visualize GPT-3's embeddings of hate speech, misinformation, and psychiatric disorder texts with Mapper. Ruppik et al. [125] performs the *clustering & topic modeling task* through dialogue term extraction.

The rest of the applications attempt the *model interpretation & analysis task*, making it the most popular task. Gardinazzi et al. [50] proposes a novel metric - *persistence similarity* to prune redundant layers in LLMs. Zheng et al. [186] uses the same technique to prune large vision-language models. Balderas et al. [11] proposes Persistent BERT Compression and Explainability (PBCE) to compress BERT by pruning redundant layers. Sun et al. [145] probes the correlation between sentence vectors and their semantics. Garcia et al. [49] performs zero-shot model stitching by employing *topological densification* (i.e., creating a topology-aware loss function). Huang et al. [69] uses a topology-aware loss function to improve prompt tuning. Athreya et al. [8] proposes a framework - HOLE (Homological Observation of Latent Embeddings for Neural Network Interpretability) to visualize the latent space of BERT for Named Entity Recognition (NER) task.

Next, several researchers probe the reasoning processes of LLMs: (1) Tan et al. [146] captures the geometry and topology of reasoning in LLMs using a mathematics examination dataset to capture step-by-step reasoning for solving non-trivial word problems; (2) Li et al. [85] and Zhang et al. [183] probe the reasoning process of the latent space of LLMs when using the chain-of-thought (CoT), tree-of-thought, and graph-of-thought prompts; (3) More et al. [98] proposes Enhanced Dirichlet and Topology Risk (EDTR), which is a novel decoding strategy that leverages persistent homology and Dirichlet-based uncertainty quantification to calculate LLM confidence; (4) Zhang et al. [185] proposes GHS-TDA, a reasoning technique to improve CoT reasoning by constructing a semantically enriched global hypothesis graph using persistent homology to capture global structures and remove redundancies; and (5) Ishimtsev et al. [71] proposes a theory that consciousness might appear when a thinking process loops back on itself with CoT prompting of LLMs. Using this theory, Ishimtsev et al. [71] implements a topology analogy, guided by persistent homology to define: normal reasoning as a *straight path*, self-reflection as a *loop*, and consciousness as a *stable loop around a hole* [71].

Finally, in support of the *model interpretation & analysis task*, several researchers probe the embedding space of LLMs to quantify the topology of the latent space. Fitz et al. [47] measures the topological complexity (known as *perforation*) of the hidden representation of LLMs to understand their topological shapes. Also, Fitz et al. [46] investigates the topological structure of the brain of ChatGPT concerning its notion of fairness. Chauhan et al. [26] proposes a novel scoring metric - *persistence scoring function* which captures the homology of the hidden representations of BERT. Fay et al. [43] investigates the differences in the topological structure of the latent space of adversarial vs. non-adversarial texts in LLMs. Kudriashov et al. [77] probes BERT's hidden weights on new grammatical features, known as *polypersonality*. Lastly, Yan et al. [174] builds an *Explainable Mapper* framework that uses two mapper agents to probe the embedding space of language models, and generate readable linguistic explanations using summarization, comparison, and perturbation operations.

`Attention Weights.` Attention weights extracted from BERT and its variants (e.g., RoBERTa) have been transformed to both directed and undirected graphs, on top of which different TDA features are extracted for the *text classification task* such as deepfake text detection [79], LLM hallucination detection [14], Code-LLM hallucination detection [162], robustness evaluation of TDA features [108], authorship attribution of Japanese texts [127], out-of-distribution detection (OOD) [112, 108], and vulnerability detection in code [141]. Additionally, this framework is applied to the *sentiment & semantic analysis task*, specifically on human linguistic competence (i.e., grammatical acceptability judgment) [27, 116, 108], dialog term extraction [164], and document coherence [72]. Similarly, with the same framework, we observe a *speech processing* application [153]. Finally, researchers attempt the *model interpretation & analysis task*, where Kostenok et al. [75] estimates uncertainty in encoder models; Samaga et al. [128] characterizes the occurrence of hallucination in LLMs; Tsai et al. [152] performs the same experiments to investigate the effects of sandbagging and code-injections in language model latent space; Varadarajan et al. [160] examines why GPT-2 significantly misrepresents gender and race identity categories by identifying which attention heads are responsible for the misclassification of specific identity groups; and Proskura et al. [115] performs dynamic weighting for building ensemble models.

ELMo Embeddings. ELMo embeddings are a type of word representation that captures both the meaning of words and their usage in context [109]. Similar to other embeddings, ELMo has also been leveraged to extract topological features. Tymochko et al. [155] performs *text classification* to detect fraudulent papers by examining their titles and abstracts. This is done in comparison of other embeddings (Word2Vec, GloVe, FastText, and Frequency Time Series) to determine the best embeddings to extract strong topological features. Similarly, Alimpiev et al. [5] compares using ELMo, GloVe, Word2Vec, and BERT embeddings to perform semantic analysis on news articles and poems.

5.2.4 Symbolic Representations

Symbolic representations in the context of AI and cognitive science refer to the use of symbols such as letters, numbers, tokens, or abstract entities to represent concepts, objects, relationships, and rules within a system. These symbols can be manipulated according to predefined rules to perform reasoning, problem-solving, and decision-making. Symbolic representation contrasts with sub-symbolic representations, such as neural network-based embeddings, which do not explicitly use symbols or rules. This section then discusses the creation of symbolic representations by using *principles of letter coding (PLC)*, *principles of speech sound coding (PSSC)*, and one-hot encoding, of which topological features are then extracted.

PLC refers to rules and methods used to encode letters that fuel various communication systems, cryptography techniques, or linguistic analyses. Letter coding transforms letters or characters into different symbols, numbers, or other forms. PSSC is similar to PLC but for extracting topological features from speech sounds. One particular application of PLC and PSSC is the study of Ukrainian tongue twisters [179, 76]. These applications attempt two tasks,

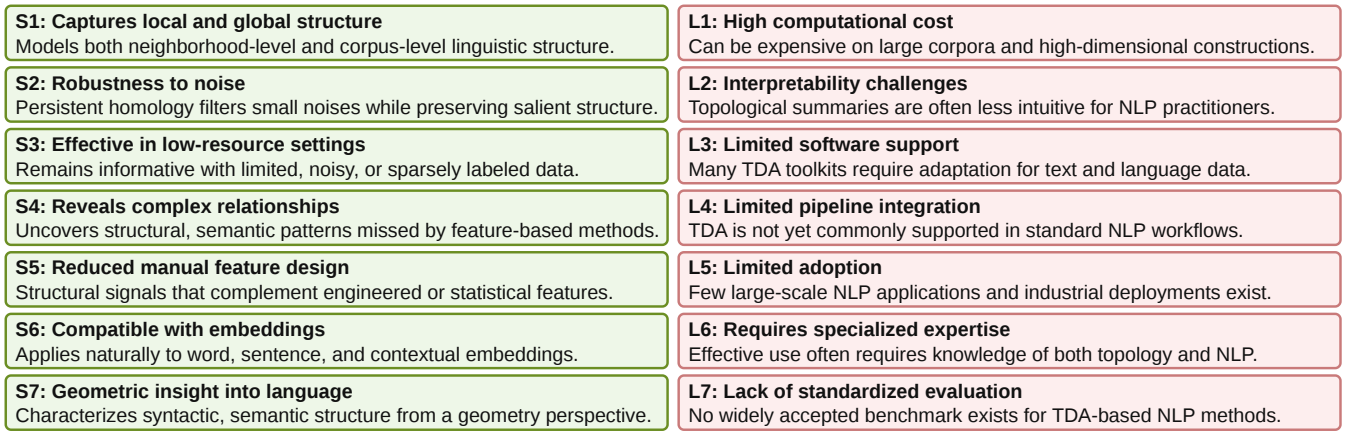

Figure 9: Strengths (**S**) and limitations (**L**) of applying TDA in NLP.

Category*	Method	Description
Linear	PCA	Reduces dimensionality while preserving variance through orthogonal transformations.
Linear	MDS	Projects high-dimensional data into lower dimensions by preserving pairwise distances.
Nonlinear	t-SNE	Projects high-dimensional data into 2D or 3D while preserving local relationships.
Nonlinear	UMAP	Similar to t-SNE, but faster and often better at preserving global structure.
Nonlinear	Isomap, LLE	Captures intrinsic structure in high-dimensional data using graph-based techniques.
Neural	Autoencoders	Learn compressed data representations through encoding and decoding.
Neural	Geometric Deep Learning	Applies neural networks to non-Euclidean spaces such as graphs and manifolds.
Graph	Spectral Clustering, GNNs	Uses graphs to model relationships and structure within data.
Clustering	DBSCAN, K-Means, HDBSCAN	Groups similar data points based on distance or density.
Kernel	SVM, Kernel PCA	Uses non-linear mappings to extract complex structures in data.
Geometric	Delaunay Triangulation, Convex Hull	Uses geometric techniques to extract data characteristics by analyzing spatial boundaries.

(*) **Linear**: Linear Projection; **Nonlinear**: Nonlinear Projection; **Neural**: Neural Network-based Methods; **Graph**: Graph-based Methods; **Clustering**: Clustering Methods; **Kernel**: Kernel Methods; **Geometric**: Geometric Methods.

Table 2: Alternatives to Topological Data Analysis in NLP

sentiment & semantic analysis and *speech processing*, respectively. Yurchuk et al. [179] uses the PLC to create word embeddings for Ukrainian tongue twisters and extract topological features from such embeddings with persistent homology. This is to distinguish tongue twister from a simple narrative sentence using support vector machine and decision tree classifiers. Similarly, Kovaliuk et al. [76] uses PSSC for classifying spoken Ukrainian tongue twisters. Additionally, Escolar et al. [41] uses one-hot encoding to represent cooking recipes numerically. Using these symbolic representations of recipes, they use the persistent homology's concept of holes to create new recipes, which were implemented and confirmed to be acceptable by a sensory evaluation study [41].

5.2.5 Multi-Modal Representations

TDA features have also been extracted from other representations of NLP-related features, including multimedia data, such as audio and video. In this section, the most popular task performed by researchers is *speech processing*, where applications include - studying human vowels and infant vocalizations [19, 18], speech recognition [178, 81], emotion recognition from audio speech [55, 119] and audio in videos [104], depression detection from audio clips [148], recognizing voiced and voiceless consonants in speech [188].

Next, we observe several applications with Vision-Language Models (VLMs), which also attempt the *model interpretation & analysis task* - integrating Representation Topology Divergence (RTD) with the loss function to align the topological structures of image and text representations during tuning [67]; implementing a topological approach to align the image and text latent manifolds in VLMs [182, 117, 172]; assessing the adversarial robustness of VLMs by measuring topological consistency [163]; assessing the topological alignment in VLMs between images and multi-lingual text (i.e., French, Spanish, Russian, etc.) [177]; and multimodal recommendations improvements [9].

6. BENEFITS AND CHALLENGES OF TDA

The use of TDA in NLP presents both clear opportunities and important challenges, many of which also extend to adjacent topological approaches. Figure 9 summarizes the main strengths and limitations of applying TDA across NLP tasks. Among its most salient strengths is its effectiveness in low-resource settings, which makes it particularly attractive for noisy, limited, and heterogeneous data. Although TDA has been shown to be a powerful mathematical technique by the numerous applications discussed above, there are alternatives to TDA (Table 2). While these alternatives have shown great utility in a variety of tasks, such as dimension reduction [16], TDA is the only technique that can extract not only local but global features. More broadly, TDA provides a principled way of capturing local and global textual structure, uncovering complex relationships, and maintaining robustness to noise. At the same time, its wider adoption remains constrained by several practical barriers, including high computational cost, limited interpretability, relatively immature software support, and the need for specialized ex-

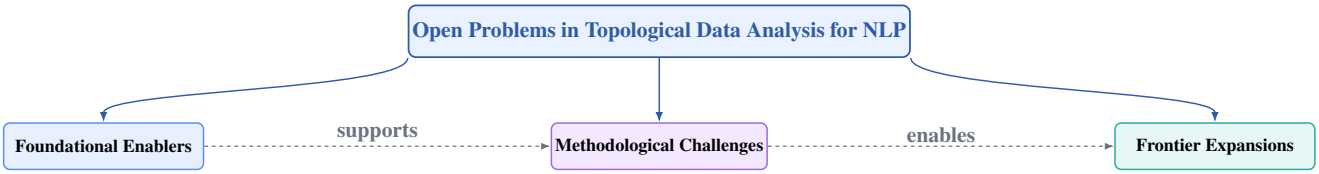

Figure 10: A high-level taxonomy of open problems in applying topological data analysis to NLP, organized into foundational enablers, methodological challenges, and frontier expansions.

pertise.

However, while persistence diagrams are not intuitively interpretable, in the context of NLP, 0D features (connected components) corresponds to clusters of semantically similar documents (topics), while 1D features (loops) can capture transitional or overlapping themes where documents form continuums rather than discrete groups. These structures help identify stable topic groupings, semantic relationships, and potential outliers, with robustness to noise. Compared to PCA or t-SNE, persistence diagrams provide complementary insights. PCA captures linear variance and may miss nonlinear structure, while t-SNE emphasizes local clustering but can distort global relationships. In contrast, persistence diagrams analyze the data in its original space and quantify both local and global structure across scales.

These strengths are further reflected in a broader body of work on related topological approaches in NLP. Beyond Persistent Homology and Mapper, researchers have explored methods grounded in simplicial homology, morse theory, homotopy, and other notions of connectedness and shape to extract structural information from language data. Although these methods are often less formalized than Persistent Homology, they similarly demonstrate the value of topological thinking for modeling semantic structure, discourse organization, stylistic variation, and model behavior. For example, topological notions such as the connected component dimension (β_0) have been used to assess document coherence through semantic connectedness [28], while more recent work has extended these ideas to dialogue semantics [130] and to topological semantic spaces for studying sociolinguistic phenomena in LLMs [70]. Other studies treat text as an object with shape, using topology to characterize stylistic and linguistic patterns [90] or to quantify semantic differences between real and fake news through thematic complexity and connectedness [135]. More recently, topological ideas have also appeared in safety-oriented LLM research, where homotopy-based methods are used to obfuscate malicious prompts or optimize jailbreak attacks [84, 167].

Taken together, these studies suggest that the appeal of TDA in NLP is part of a broader methodological advantage of topological approaches: their ability to reveal structural regularities in language that are often difficult to capture through conventional feature-based or purely predictive methods alone. Similarly, limitations of TDA, such as computational expense, interpretability challenges, and limited software, highlight the practical obstacles that must be addressed before topological methods can see broader adoption in NLP. While ongoing advances in algorithms, software, and computing infrastructure offer reasons for optimism, these strengths and challenges together point to several promising directions for future work, which we discuss in Section 7.

7. OPEN PROBLEMS

Although TDA has shown growing promise in NLP, many challenges remain before its full potential can be realized. In this section, we organize the main open problems and future directions into three broad categories: foundational enablers, methodological challenges, and frontier expansions. Together, these categories highlight how researchers can better leverage the strengths of TDA while addressing its current limitations and risks (Figure 10).

7.1 Foundational Enablers

LLM-Assisted TDA Code Generation. One of the major challenges in applying TDA to NLP tasks is the steep learning curve associated with its mathematical foundations, which are often accessible only to expert audiences. Moreover, theorists who develop and understand these advanced concepts do not always collaborate with computational scientists to translate them into executable code. To address this gap, Liu et al. [87] proposes using ChatGPT to generate Python code for TDA concepts by training it on these mathematical foundations. Their findings suggest that ChatGPT can alleviate this bottleneck, particularly for complex TDA concepts like hypergraphs, digraphs, and persistent harmonic space, which have not been as heavily explored as the Vietoris-Rips complex [87]. Similarly, experts can develop specialized code generators, such as fine-tuning models like Code-Llama [124] on TDA concepts. In addition, Li et al. [86] proposes to benchmark LLM as topological thinkers by giving them tasks to implement persistent homology on. The idea is to create *LLM4PH*, an LLM for persistent homology. Therefore, we observe that we are closer to creating a dedicated LLM for TDA code generation which could significantly lower the barrier to entry, encouraging the NLP community to explore TDA applications more innovatively.

Topology-Linguistics Alignment. TDA can come across as not intuitive. Even though it has the potential to be applied to interpret different behaviors of modern NLP models, there is still a need for theoretical approaches that better tie TDA features to linguistic phenomena but intuitively. For example, Draganov et al. [36] investigates the shape of words and their embeddings in Indo-European languages and find similar conclusions to Port et al. [113, 114], which investigate the syntactic topological space of such languages. They find that TDA features represent historical facts, such that languages clustered closely together are similar or influenced by each other. These applications show how TDA can be used to reveal and confirm linguistic phenomena. However, we currently observe only 13 theoretical TDA works in NLP, compared to over 100 non-theoretical ones. Thus, we need more theoretical TDA approaches, as it is impractical to achieve the depth and understanding of performance from a topological perspective without further investigations.

7.2 Methodological Challenges

Interpretability of TDA Results. Interpreting TDA features in NLP problems, given its non-intuitive nature is very challenging.

This is evident in the fact that most TDA for explainability applications is mostly in Computer Vision [132], where the structure is clearly apparent. Consequently, the interpretation of TDA features for text or speech data remains an open problem. There are currently two main tasks in this space - (1) explain model performance by interpreting TDA features extracted from the model; and (2) explain model performance by using TDA to probe the prediction space or data. Either task requires a deeper understanding of TDA such that intuitive explanations can be used to tie topology to linguistic phenomena. Specifically, we need novel approaches that link TDA features to linguistic phenomena, for instance, disentangling β_0, and β_1's representations to different properties of natural texts such as coherency, and writing style. This can be done either through visualizing the latent space of a model [120, 174, 128] or probing the latent space of models with TDA [144, 171, 142, 174].

TDA for Understanding NLP Model Behavior. Existing studies suggest that TDA is promising for interpreting the behavior of NLP models; however, current efforts remain scattered across tasks, representations, and model settings. In particular, TDA has been used to probe neural representations and make language models less opaque by characterizing the geometry and topology of their learned hidden space. For example, existing work spans representation geometry and alignment across modalities, including text embeddings, sentence semantics, text–speech alignment, adversarial versus non-adversarial latent spaces, and image–text alignment [62, 145, 175, 176, 43, 163, 177, 152]. Other studies use TDA to probe hidden states and model weights, revealing structural properties of language model latent spaces and uncovering grammatical or representational patterns [26, 77, 47]. TDA has also been applied to training and optimization dynamics, model-level explanation and diagnostics, model efficiency & safety, reliability, and reasoning-related behavior [120, 49, 69, 98, 185, 144, 57, 174, 115, 50, 11, 180, 46, 75, 128, 160, 44, 118, 146, 85, 71]. Beyond these works, how can TDA move beyond case-specific analysis to become a principled and generalizable framework for interpreting complex model behavior in NLP is of great value for future work.

Improved TDA Feature Extraction and Representation Selection. Unlike some other data modalities that possess an intrinsic geometric structure, texts acquire their "shape" only through their numerical representations. In other words, the topology we observe is not an inherent property of the text itself, but of the embedding method used to encode it. A corpus represented with TF–IDF will therefore exhibit a geometry characteristic of sparse lexical weighting, whereas Transformer-based pre-trained embeddings induce a different shape driven by contextual semantic structure. These differences are partly explained by the distinct linguistic features each representation captures - lexical frequency in the case of TF–IDF, and richer semantic and syntactic information in contextual embeddings. However, the resulting geometric and topological variations are often unintuitive. As a result, it becomes difficult to determine which numerical representation is most appropriate for extracting meaningful TDA features for a given task. To address this challenge, we must develop principled methods for selecting representations that align with the objectives of the analysis. This includes exploring new forms of numerical encoding, such as *symbolic representations* that capture the diversity of textual phenomena across tasks. Equally important is the development of improved strategies for leveraging existing embeddings in ways that enhance the stability, interpretability, and task-relevance of the extracted topological features.

7.3 Frontier Expansions

Adversarial Robustness of TDA Features. Robustness to noise, particularly adversarial perturbations, has been an important research topic in NLP. While such robustness of TDA features is promising, there have been only a few works in this direction [108, 26, 43, 163]. For instance, Perez et al. [108] shows that their topologically-augmented BERT model is more robust than the vanilla BERT model when tested against perturbations generated by TextAttack [99]. Chauhan et al. [26] also show that there are some weak correlations between persistent homology features of a trained BERT model and its adversarial robustness against several state-of-the-art attackers. Recently, we have observed interesting applications that use topology to track the latent space of LLMs before and after adversarial perturbations are introduced [43, 163]. These studies highlight the emerging application of topology in robustness evaluations, which will benefit the NLP, ML, and security communities.

Novel Applications of TDA. When we have more theoretical approaches of TDA and issues barring the application of TDA on interpretable NLP tasks are mitigated, we can hope that TDA can be applied to even more novel, diverse, and important tasks. From Section 5.1, we can see that TDA has been applied to seven non-theoretical NLP tasks. While many of the tasks are interesting, especially the speech processing and health applications, there are still nuanced niche fields that could benefit from TDA. One glaring application is on multi-lingual tasks [62, 49, 177]. Due to the benefits of TDA, which include performing robustly on heterogeneous, imbalanced, and noisy data, its application to multi-lingual tasks is necessary. Other applications include: *Topology-aware neural networks, Topological interpretability, semantic and syntactic structural analysis, forensic authorship, multi-modal (e.g., VLMs) analysis,* etc.

Topological Deep Learning for NLP. Due to the benefits of TDA and deep learning, a new niche field is born - Topological Deep Learning (TDL) as "the collection of ideas and methods related to the use of topological concepts in deep learning" [107]. Initially, TDL is described as an ensemble of topological features extracted by TDA techniques such as persistent homology and deep learning features. In this setting, TDL is a traditional deep learning model with extra features (i.e., TDA-extracted features). However, as the field has advanced, a new definition for TDL has emerged - "the collection of ideas and methods related to the use of topological concepts in deep learning" [107]. TDL allows a deep learning model to be integrated more deeply with concepts of algebraic topology, such as the introduction of simplicial neural networks (NNs) [37, 106], which are NNs with layers made up of simplicial complexes. This deeper integration of TDA into NNs makes TDL particularly useful for the explosion of high-dimensional data. These high-dimensional data require better tools for processing as the current tools shrink the dimension, resulting in information loss. In NLP, one particular approach to integrate TDA with high-dimensional NLP embeddings has been the utilization of text in graphical forms, which have been shown to yield better results than directly using texts as a sequence of tokens [88, 111, 85, 79]. Nevertheless, more research is still needed to validate such an approach.

8. CONCLUSION

Our world is currently experiencing an explosion of data and an explosion of computational techniques to process such data. Machine Learning (ML) is the most popular of these computational methods. However, while its benefits are numerous, it has a few limita-

tions. The biggest of the limitations of ML is its inability to sufficiently process data that is high-dimensional, imbalanced, noisy, and scarce. Therefore, a small community of NLP researchers emerged to tackle this limitation by proposing using TDA to tackle difficult NLP tasks. These researchers employ two TDA techniques - Persistent Homology and Mapper to solve NLP tasks using theoretical and non-theoretical approaches. This yielded 137 papers, which we comprehensively surveyed in this paper. Finally, we conclude that while the applications of TDA in NLP have improved greatly since 2012, there is still room for improvement, specifically in reducing the barrier to entry for non-TDA experts to apply it to their NLP tasks.

9. ETHICAL STATEMENT

This survey highlights emerging applications of Topological Data Analysis (TDA) in Natural Language Processing. While our primary goal is to synthesize existing work, we recognize that several use cases carry important ethical considerations and dual-use risks. Topological methods can inadvertently expose latent sensitive attributes (e.g., dialect, health cues, authorship), enabling reidentification or profiling even when data is anonymized. Applications in speech, emotion, and health-related domains further raise fairness, consent, and equity concerns, particularly for minority groups and low-resource languages. Therefore, we emphasize the need for bias and robustness audits, careful data governance and licensing, and privacy-preserving mechanisms when sharing derived features. Responsible release practices, such as restricting code that enables circumvention, conducting red-team evaluations, and requiring IRB or ethics review for clinical or surveillance-adjacent uses, are essential. Finally, given the computational demands of TDA pipelines, their environmental impact should be considered as well.

10. ACKNOWLEDGMENTS

The authors thank Dr. Charlie Dagli for his encouragement, reading the paper drafts, and providing invaluable recommendations.

References

[1] H. Adams, T. Emerson, M. Kirby, R. Neville, C. Peterson, P. Shipman, S. Chepushtanova, E. Hanson, F. Motta, and L. Ziegelmeier. Persistence images: A stable vector representation of persistent homology. *Journal of Machine Learning Research*, 18(8):1–35, 2017.

[2] M. Alagappan, D. Jiang, N. Denko, and A. C. Koong. A multimodal data analysis approach for targeted drug discovery involving topological data analysis (tda). In *Tumor Microenvironment: Study Protocols*, pages 253–268. Springer, 2016.

[3] L. Alanís-López, J. P. B. Lafarga, L. R. G. Sánchez, E. I. L. Otañez, A. Ramirez-Cabello, and A. Ucan-Puc. Detection of ai generated texts using deep learning and topological data analysis. In *Mexican Congress on Artificial Intelligence*, pages 69–82. Springer, 2025.

[4] A. W. Alexander and H. Wang. Topological data mapping of online hate speech, misinformation, and general mental health: A large language model based study. *PLOS Digital Health*, 4(7):e0000935, 2025.

[5] E. Alimpiev and V. Myers. Plot holes and text topology. *Stanford CS224N Custom Project*, 2020.

[6] A. Arun, K. K. Chandra, A. Sinha, B. Velayutham, J. Arora, M. Jain, and P. Kumaraguru. Topo goes political: Tda-based controversy detection in imbalanced reddit political data. In *Companion Proceedings of the ACM on Web Conference 2025*, pages 2616–2625, 2025.

[7] E. Asriani, I. Muchtadi-Alamsyah, and A. Purwarianti. Topological data analysis for transformer nmt: Exploring the use of cohomology-based persistence landscapes as a representation of global context. In *2025 12th International Conference on Advanced Informatics: Concept, Theory and Application (ICAICTA)*, pages 1–6, 2025.

[8] S. M. Athreya and P. Rosen. Hole: Homological observation of latent embeddings for neural network interpretability. *arXiv preprint arXiv:2512.07988*, 2025.

[9] K. Bachiri, A. Yahyaouy, M. Malek, and N. Rogovschi. Topological data analysis and graph-based learning for multimodal recommendation. *IEEE Access*, 2025.

[10] M. M. Bailey and M. I. Heiligman. Detecting narrative shifts through persistent structures: A topological analysis of media discourse. *arXiv preprint arXiv:2506.14836*, 2025.

[11] L. Balderas, M. Lastra, and J. M. Benítez. A green ai methodology based on persistent homology for compressing bert. *Applied Sciences*, 15(1):390, 2025.

[12] U. Bauer. Ripser: efficient computation of vietoris–rips persistence barcodes. *Journal of Applied and Computational Topology*, 5(3):391–423, 2021.

[13] U. Bauer, M. Kerber, J. Reininghaus, and H. Wagner. Phat–persistent homology algorithms toolbox. *Journal of symbolic computation*, 78:76–90, 2017.

[14] A. Bazarova, A. Yugay, A. Shulga, A. Ermilova, A. Volodichev, K. Polev, J. Belikova, R. Parchiev, D. Simakov, M. Savchenko, et al. Hallucination detection in llms via topological divergence on attention graphs. *arXiv preprint arXiv:2504.10063*, 2025.

[15] A. Bensalem, M. A. Bensalem, and M. A. Chadli. Detecting token-level sentiment change in text translation through zig zag persistent homology. *https://hal.science/hal-05139630/*, 2025.

[16] J. A. D. Binnie, P. Dłotko, J. Harvey, J. Malinowski, and K. M. Yim. A survey of dimension estimation methods, 2025.

[17] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5:135–146, 2017.

[18] G. Bonafos, C. Bourot, P. Pudlo, J.-M. Freyermuth, L. Reboul, S. Tronçon, and A. Rey. Dirichlet process mixture model based on topologically augmented signal representation for clustering infant vocalizations. In *Proc. Interspeech 2024*, pages 3605–3609, 2024.

[19] G. Bonafos, J.-M. Freyermuth, P. Pudlo, S. Tronçon, and A. Rey. Topological data analysis of human vowels: Persistent homologies across representation spaces. *arXiv preprint arXiv:2310.06508*, 2023.

[20] H. Bouazzaoui, M. A. Elomary, and M. I. Mamouni. An application of persistent homology and the graph theory to linguistics: The case of tifinagh and phoenician scripts. *Statistics in Transition new series*, 22(3):141–156, 2021.

[21] M. Byers. *The Hidden Shape of Data: Topological Data Analysis for Anxiety Detection in Text*. PhD thesis, Texas State University, 2021.

[22] C. Byrne, D. Horak, K. Moilanen, and A. Mabona. Topic modeling with topological data analysis. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11514–11533, 2022.

[23] G. Carlsson. Topological methods for data modelling. *Nature Reviews Physics*, 2(12):697–708, 2020.

[24] D. Cavaliere, S. Senatore, and V. Loia. Context-aware profiling of concepts from a semantic topological space. *Knowledge-Based Systems*, 130:102–115, 2017.

[25] J. Chandra, S. K. Navneet, and Y. Zhang. The topology of recovery: Using persistent homology to map individual mental health journeys in online communities. *arXiv preprint arXiv:2602.23886*, 2026.

[26] J. Chauhan and M. Kaul. Bertops: Studying bert representations under a topological lens. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2022.

[27] D. Cherniavskii, E. Tulchinskii, V. Mikhailov, I. Proskurina, L. Kushnareva, E. Artemova, S. Barannikov, I. Piontkovskaya, D. Piontkovski, and E. Burnaev. Acceptability judgements via examining the topology of attention maps. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 88–107, 2022.

[28] I.-J. Chiang. Discover the semantic topology in high-dimensional data. *Expert Systems with Applications*, 33(1):256–262, 2007.

[29] N. H. Christianson, A. Sizemore Blevins, and D. S. Bassett. Architecture and evolution of semantic networks in mathematics texts. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 476(2239), 2020.

[30] F. Cornell. An explainable topological search engine with giotto-tda. In *gtda-challenge-2020*, 2020.

[31] S. Das, S. A. Haque, and M. I. Tanveer. Persistence homology of tedtalk: Do sentence embeddings have a topological shape? *arXiv preprint arXiv:2103.14131*, 2021.

[32] R. Deng and F. Duzhin. Topological data analysis helps to improve accuracy of deep learning models for fake news detection trained on very small training sets. *Big Data and Cognitive Computing*, 6(3):74, 2022.

[33] T. K. Dey and S. Mandal. Protein classification with improved topological data analysis. In *18th International Workshop on Algorithms in Bioinformatics (WABI 2018)*. Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2018.

[34] R. Dong. Linguistics from a topological viewpoint. *arXiv preprint arXiv:2403.15440*, 2024.

[35] P. Doshi and W. Zadrozny. Movie genre detection using topological data analysis. In *Statistical Language and Speech Processing: 6th International Conference, SLSP 2018, Mons, Belgium, October 15–16, 2018, Proceedings 6*, pages 117–128. Springer, 2018.

[36] O. Draganov and S. Skiena. The shape of word embeddings: Quantifying non-isometry with topological data analysis. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 12080–12099, 2024.

[37] S. Ebli, M. Defferrard, and G. Spreemann. Simplicial neural networks. In *TDA & Beyond*, 2020.

[38] H. Edelsbrunner, J. Harer, et al. Persistent homology-a survey. *Contemporary mathematics*, 453(26):257–282, 2008.

[39] B. Effah. *Topological data analysis of open-ended responses*. PhD thesis, University of Cape Coast, 2017.

[40] N. Elyasi and M. H. Moghadam. An introduction to a new text classification and visualization for natural language processing using topological data analysis. *arXiv preprint arXiv:1906.01726*, 2019.

[41] E. G. Escolar, Y. Shimada, and M. Yuasa. A topological analysis of the space of recipes. *International Journal of Gastronomy and Food Science*, 39:101088, 2025.

[42] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231, 1996.

[43] A. Fay, I. García-Redondo, Q. Wang, H. Dubossarsky, and A. Monod. The shape of adversarial influence: Characterizing LLM latent spaces with persistent homology. In *The Fourteenth International Conference on Learning Representations*, 2026.

[44] S. J. Feng, E. M. Lai, and W. Li. Geometry of textual data augmentation: Insights from large language models. *Electronics*, 13(18):3781, 2024.

[45] S. Fitz. The shape of words-topological structure in natural language data. In *Topological, Algebraic and Geometric Learning Workshops 2022*, pages 116–123. PMLR, 2022.

[46] S. Fitz. Do large gpt models discover moral dimensions in language representations? a topological study of sentence embeddings. *arXiv preprint arXiv:2309.09397*, 2023.

[47] S. Fitz, P. Romero, and J. J. Schneider. Hidden holes - topological aspects of language models. In *NeurIPS 2024 Workshop on Symmetry and Geometry in Neural Representations*, 2024.

[48] J. S. Garcia. *Applications of topological data analysis to natural language processing and computer vision*. PhD thesis, Colorado State University, 2022.

[49] A. García-Castellanos, G. L. Marchetti, D. Kragic, and M. Scolamiero. Relative representations: Topological and geometric perspectives. In *UniReps: 2nd Edition of the Workshop on Unifying Representations in Neural Models*, 2024.

[50] Y. Gardinazzi, K. Viswanathan, G. Panerai, A. Cazzaniga, M. Biagetti, et al. Persistent topological features in large language models. In *Forty-second International Conference on Machine Learning*, 2025.

[51] S. Gholizadeh, A. Seyeditabari, and W. Zadrozny. Topological signature of 19th century novelists: Persistent homology in text mining. *big data and cognitive computing*, 2(4):33, 2018.

[52] S. Gholizadeh, A. Seyeditabari, and W. Zadrozny. A novel method of extracting topological features from word embeddings. *arXiv preprint arXiv:2003.13074*, 2020.

[53] S. Gholizadeh and W. Zadrozny. A short survey of topological data analysis in time series and systems analysis. *arXiv preprint arXiv:1809.10745*, 2018.

[54] R. Ghrist. Barcodes: the persistent topology of data. *Bulletin of the American Mathematical Society*, 45(1):61–75, 2008.

[55] R. Gonzalez-Diaz, E. Paluzo-Hidalgo, and J. F. Quesada. Towards emotion recognition: a persistent entropy application. In *Computational Topology in Image Context: 7th International Workshop, CTIC 2019, Málaga, Spain, January 24-25, 2019, Proceedings 7*, pages 96–109. Springer, 2019.

[56] I. Goshev, P. Sekuloski, I. Chorbev, D. Kitanovski, and V. D. Ristovska. Topology as a lens for semantic organization in transformer embeddings. *INTERNATIONAL SCIENTIFIC JOURNAL "MATHEMATICAL MODELING"*, 2025.

[57] K. Gourgoulias, N. Ghalyan, M. Labonne, S. Moran, J. Sabelja, et al. Estimating class separability of text embeddings with persistent homology. *Transactions on Machine Learning Research*, 2024.

[58] V. A. Gromov, Q. N. Dang, and A. S. Erbolova. A language and its holes: The first-order homology of the large-scale geometrical structure of a natural language. *Complexity*, 2025(1):9659172, 2025.

[59] H. Guan, W. Tang, H. Krim, J. Keiser, A. Rindos, and R. Sazdanovic. A topological collapse for document summarization. In *2016 IEEE 17th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 1–5. IEEE, 2016.

[60] A. Guilinger, E. Best, and V. Awasthi. Topological data analysis for distinguishing human written and ai generated abstracts. *preprints.org*, 2025.

[61] P. Haghighatkhah, A. Fokkens, P. Sommerauer, B. Speckmann, and K. Verbeek. Story trees: Representing documents using topological persistence. In *Proceedings of the Thirteenth LREC 2022*, pages 2413–2429, 2022.

[62] S. Haim Meirom and O. Bobrowski. Unsupervised geometric and topological approaches for cross-lingual sentence representation and comparison. In S. Gella, H. He, B. P. Majumder, B. Can, E. Giunchiglia, S. Cahyawijaya, S. Min, M. Mozes, X. L. Li, I. Augenstein, A. Rogers, K. Cho, E. Grefenstette, L. Rimell, and C. Dyer, editors, *Proceedings of the 7th Workshop on Representation Learning for NLP*, pages 173–183, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[63] F. Hensel, M. Moor, and B. Rieck. A survey of topological machine learning methods. *Frontiers in Artificial Intelligence*, 4:681108, 2021.

[64] W. J. Holmes. Topological analysis of averaged sentence embeddings. Master's thesis, Wright State University, 2020.

[65] M. D. Hopp, V. Labatut, A. Amalvy, R. Dufour, H. Stone, H. K. Jach, and K. Murayama. Persistent homology of topic networks for the prediction of reader curiosity. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 28121–28132, 2025.

[66] M. Horn, E. De Brouwer, M. Moor, Y. Moreau, B. Rieck, and K. Borgwardt. Topological graph neural networks. In *International Conference on Learning Representations*, 2022.

[67] D. Huang. Topology-aware clip few-shot learning. *arXiv preprint arXiv:2505.01694*, 2025.

[68] J. Huang. A tda approach of analyzing election speeches with nlp techniques. *Journal of Computing Sciences in Colleges*, 38(3):215–215, 2022.

[69] Z. Huang, C. Zhang, H. Bian, S. Zhang, C.-l. A. Tai, J. Zhang, C. Qin, J. Qu, Y. Ye, Y. Yang, et al. Optimizing soft prompt tuning via structural evolution. *arXiv preprint arXiv:2602.16500*, 2026.

[70] I. Ionescu, J. Leung, Y. Siglidis, et al. Generative topolinguistics: Bidirectional interfaces for emergent language topologies. *Antikythera: Journal for the Philosophy of Planetary Computation*, 2025.

[71] T. Ishimtsev. The hole-in-the-bagel theory of consciousness-a phenomenological study and topodynamic model for emergent coherence in llms. *https://zenodo.org/records/18346699*, 2026.

[72] S. Jain, R. Singhal, S. Krishna, Y. K. Singla, and R. R. Shah. Beyond words: A topological exploration of coherence in text documents. In *The Second Tiny Papers Track at ICLR 2024*, 2024.

[73] A. Jakubowski, M. Gasic, and M. Zibrowius. Topology of word embeddings: Singularities reflect polysemy. In I. Gurevych, M. Apidianaki, and M. Faruqui, editors, *Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics*, pages 103–113, Barcelona, Spain (Online), Dec. 2020. Association for Computational Linguistics.

[74] J. Karlgren, M. Bohman, A. Ekgren, G. Isheden, E. Kullmann, and D. Nilsson. Semantic topology. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 1939–1942, 2014.

[75] E. Kostenok, D. Cherniavskii, and A. Zaytsev. Uncertainty estimation of transformers' predictions via topological analysis of the attention matrices. *arXiv preprint arXiv:2308.11295*, 2023.

[76] T. Kovaliuk, I. Yurchuk, and O. Gurnik. Topological structure of ukrainian tongue twisters based on speech sound analysis. *6th International Workshop on Modern Data Science Technologies*, 2024.

[77] S. Kudriashov, V. Zykova, A. Stepanova, J. Raskind, and E. Klyshinsky. The more polypersonal the better-a short look on space geometry of fine-tuned layers. In *International Conference on Neuroinformatics*, pages 13–22. Springer, 2024.

[78] A. Kumar and A. Sarkar. Extractive text summarization using topological features. In *International Workshop on Combinatorial Image Analysis*, pages 105–121. Springer, 2022.

[79] L. Kushnareva, D. Cherniavskii, V. Mikhailov, E. Artemova, S. Barannikov, A. Bernstein, I. Piontkovskaya, D. Piontkovski, and E. Burnaev. Artificial text detection via examining the topology of attention maps. In *Proceedings of the 2021 EMNLP*, pages 635–649, 2021.

[80] L. Kushnareva, T. Gaintseva, G. Magai, S. Barannikov, D. Abulkhanov, K. Kuznetsov, E. Tulchinskii, I. Piontkovskaya, and S. Nikolenko. Ai-generated text boundary detection with roft. In *1st Conference on Language Modeling (COLM)*, volume 2024, 2024.

[81] C. Laméris. *Topological Featurization of Speech Data for Speech Recognition*. PhD thesis, University of Groningen, 2024.

[82] Z. Lamine, M. I. Mamouni, and M. W. Mansouri. A topological data analysis of the protein structure. *International Journal of Analysis and Applications*, 21:136–136, 2023.

[83] R. Lavery, A. Jurek-Loughrey, and L. Bai. Combining topological signature with text embeddings: Multi-modal approach to fake news detection. In *2024 35th Irish Signals and Systems Conference (ISSC)*, pages 1–6. IEEE, 2024.

[84] L. Lazo, H. Jelodar, and R. Razavi-Far. Llm security and safety: Insights from homotopy-inspired prompt obfuscation. *arXiv preprint arXiv:2601.14528*, 2026.

[85] C. Li, C. Zhang, Y. Lu, S. Chen, X. Wang, J. Zhang, Z. Wang, Z. Jin, K. Liu, S.-H. Bae, et al. Understanding chain-of-thought in large language models via topological data analysis. *arXiv preprint arXiv:2512.19135*, 2025.

[86] H. Li, H. Wan, Y. Huang, Y. Chen, Y. Gel, and H. Jiang. Large language models as topological thinkers: A benchmark on graph persistent homology, 2026.

[87] J. Liu, L. Shen, and G.-W. Wei. Chatgpt for computational topology. *Foundations of data science (Springfield, Mo.)*, 6(2):221, 2024.

[88] X. Liu, Z. Zhang, Y. Wang, H. Pu, Y. Lan, and C. Shen. Coco: Coherence-enhanced machine-generated text detection under low resource with contrastive learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16167–16188, 2023.

[89] B. Løvlie. Text classification via topological data analysis. Master's thesis, Norwegian University of Science and Technology (NTNU), 2023.

[90] X. Luong, M. Juillard, S. Mellet, and D. Longrée. Trees and after: The concept of text topology. *Literary and Linguistic Computing*, 22(2):167–186, 2007.

[91] L. Maadarani and S. G. Hajra. The shape of poems. In *Fall Poster Forum*, 2020.

[92] K. MacDonald, A. Ruparelia, B. Rogers, A. Bovell, and B. Stone. Binary malware attribution using llm embeddings and topological data analysis. *Conference on Applied Machine Learning for Information Security*, 2024.

[93] A. Maćkiewicz and W. Ratajczak. Principal components analysis (pca). *Computers & Geosciences*, 19(3):303–342, 1993.

[94] L. McInnes, J. Healy, N. Saul, and L. Großberger. Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29):861, 2018.

[95] Y. Meng, R.-H. Li, H. Qin, X. Wu, H. Duan, Y. Lu, and G. Wang. Encoding group interests with persistent homology for personalized search. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 54(9):5606–5616, 2024.

[96] P. Michel, A. Ravichander, and S. Rijhwani. Does the geometry of word embeddings help document classification? a case study on persistent homology-based representations. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 235–240, 2017.

[97] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.

[98] A. More, A. Zhang, N. Bonilla, A. Vivekan, K. Zhu, P. Sharafoleslami, and M. Chaudhary. Optimizing chain-of-thought confidence via topological and dirichlet risk analysis. *arXiv preprint arXiv:2511.06437*, 2025.

[99] J. Morris, E. Lifland, J. Y. Yoo, J. Grigsby, D. Jin, and Y. Qi. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 119–126, 2020.

[100] E. Munch. A user's guide to topological data analysis. *Journal of Learning Analytics*, 4(2):47–61, 2017.

[101] J. Murugan and D. Robertson. An introduction to topological data analysis for physicists: From lgm to frbs. *arXiv preprint arXiv:1904.11044*, 2019.

[102] J. L. Nielson, J. Paquette, A. W. Liu, C. F. Guandique, C. A. Tovar, T. Inoue, K.-A. Irvine, J. C. Gensel, J. Kloke, T. C. Petrossian, et al. Topological data analysis for discovery in preclinical spinal cord injury and traumatic brain injury. *Nature communications*, 6(1):8581, 2015.

[103] L. Novak. Network and topological analysis of scholarly metadata: A platform to model and predict collaboration. Master's thesis, Purdue University, 2019.

[104] E. Paluzo-Hidalgo, R. Gonzalez-Diaz, and G. Aguirre-Carrazana. Emotion recognition in talking-face videos using persistent entropy and neural networks. *Electronic Research Archive*, 30(2):644–660, 2022.

[105] E. Paluzo Hidalgo, R. González Díaz, and M. Á. Gutiérrez Naranjo. Summary and distance between sets of texts based on topological data analysis. *arXiv preprint arXiv:1912.09253*, 2019.

[106] E. Paluzo-Hidalgo, R. Gonzalez-Diaz, and M. A. Gutierrez-Naranjo. Trainable and explainable simplicial map neural networks. *Information Sciences*, 667:120474, 2024.

[107] T. Papamarkou, T. Birdal, M. M. Bronstein, G. E. Carlsson, J. Curry, Y. Gao, M. Hajij, R. Kwitt, P. Lio, P. Di Lorenzo, et al. Position: Topological deep learning is the new frontier for relational learning. In *Forty-first International Conference on Machine Learning*, 2024.

[108] I. Perez and R. Reinauer. The topological bert: Transforming attention into topology for natural language processing. *arXiv preprint arXiv:2206.15195*, 2022.

[109] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, 2018.

[110] G. Petri and A. Leitao. Prediction of disease type from topological features of time series. In *gtda-challenge-2020*, 2020.

[111] H. T. Phan, N. T. Nguyen, and D. Hwang. Fake news detection: A survey of graph neural network methods. *Applied Soft Computing*, 139:110235, 2023.

[112] A. Pollano, A. Chaudhuri, and A. Simmons. Detecting out-of-distribution text using topological features of transformer-based language models. *The IJCAI-2024 AISafety Workshop*, 2024.

[113] A. Port, I. Gheorghita, D. Guth, J. M. Clark, C. Liang, S. Dasu, and M. Marcolli. Persistent topology of syntax. *Mathematics in Computer Science*, 12(1):33–50, 2018.

[114] A. Port, T. Karidi, and M. Marcolli. Topological analysis of syntactic structures. *Mathematics in Computer Science*, 16(1):2, 2022.

[115] P. Proskura and A. Zaytsev. Beyond simple averaging: Improving nlp ensemble performance with topological-data-analysis-based weighting. In *2024 IEEE 11th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 1–8. IEEE, 2024.

[116] I. Proskurina, E. Artemova, and I. Piontkovskaya. Can bert eat rucola? topological data analysis to explain. In *Proceedings of the 9th Workshop on Slavic Natural Language Processing 2023 (SlavicNLP 2023)*, pages 123–137, 2023.

[117] A. A. Rahim et al. Topological perspectives on optimal multimodal embedding spaces. *arXiv preprint arXiv:2405.18867*, 2024.

[118] N. Rair, A. Goupil, V. Vrabie, and E. Chochoy. When annotators disagree, topology explains: Mapper, a topological tool for exploring text embedding geometry and ambiguity. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 8468–8491, 2025.

[119] M. B. Raskin. The shape of emotion in speech: A topological data analysis of speech emotion recognition. *IEEE Access*, 14:52964–52978, 2026.

[120] A. Rathore, Y. Zhou, V. Srikumar, and B. Wang. Topobert: Exploring the topology of fine-tuned word representations. *Information Visualization*, 22(3):186–208, 2023.

[121] M. Rawson, S. Dooley, M. Bharadwaj, and R. Choudhary. Topological data analysis for word sense disambiguation. *arXiv preprint arXiv:2203.00565*, 2022.

[122] R. Rejimoan. Detection of machine-generated text by integrating roberta embeddings with topological features. *International Journal of Applied Mathematics*, 38(6s):17–30, 2025.

[123] B. Rieck. Topological data analysis for machine learning. *ECML-PKDD 2020 (Tutorial)*, 2020.

[124] B. Roziere, J. Gehring, F. Gloeckle, S. Sootla, I. Gat, X. E. Tan, Y. Adi, J. Liu, R. Sauvestre, T. Remez, et al. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*, 2023.

[125] B. M. Ruppik, M. Heck, C. van Niekerk, R. Vukovic, H.-c. Lin, S. Feng, M. Zibrowius, and M. Gašić. Local topology measures of contextual language model latent spaces with applications to dialogue term extraction. *Proceedings of the 25th Meeting of the Special Interest Group on Discourse and Dialogue*, 2024.

[126] S. M. N. Sakib. S M Nazmuz Sakib Affix Isometry Index, 2025. researchgate.net.

[127] W. Sakurai, M. Asano, D. Imoto, M. Honma, and K. Kurosawa. Authorship attribution by attention pattern of bert with topological data analysis and umap. In *2025 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pages 0296–0301. IEEE, 2025.

[128] S. N. Samaga, G. G. Arroyo, and T. K. Dey. Halluzig: Hallucination detection using zigzag persistence. *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics*, 2026.

[129] I. R. Sami and K. Farrahi. A simplified topological representation of text for local and global context. In *Proceedings of the 25th ACM International Conference on Multimedia*, pages 1451–1456, 2017.

[130] A. B. Santacana. A topological semantics of dialogue: Nerve structures and logical extraction. *arXiv preprint arXiv:2506.00615*, 2025.

[131] L. Sassone, M. Manetti, M. G. Bergomi, and M. Ferri. *Bridging Topological Persistence and Machine Learning for Music Information Retrieval*. PhD thesis, Sapienza – University of Rome, 2022.

[132] N. Saul and D. L. Arendt. Machine learning explanations with topological data analysis. In *Demo at the Workshop on Visualization for AI explainability (VISxAI)*, 2018.

[133] N. Saul and C. Tralie. Scikit-tda: Topological data analysis for python. *URL https://doi. org/10.5281/zenodo*, 2533369, 2019.

[134] K. Savle, W. Zadrozny, and M. Lee. Topological data analysis for discourse semantics? In *Proceedings of the 13th International Conference on Computational Semantics-Student Papers*, pages 34–43, 2019.

[135] B. Scalvini and A. Mashaghi. Semantic topology: a new perspective for communication style characterization. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 9223–9233, 2025.

[136] D. Shehu. An analysis of the effect of polysemy on the topology of the latent manifold. Master's thesis, Eindhoven University of Technology, 2024.

[137] K. Shin. Genre classification: A topological data analysis approach. *kevin-shin.com*, 2019.

[138] G. Singh, F. Mémoli, G. E. Carlsson, et al. Topological methods for the analysis of high dimensional data sets and 3d object recognition. *PBG@ Eurographics*, 2:091–100, 2007.

[139] Y. Singh, C. M. Farrelly, Q. A. Hathaway, T. Leiner, J. Jagtap, G. E. Carlsson, and B. J. Erickson. Topological data analysis in medical imaging: current state of the art. *Insights into Imaging*, 14(1):58, 2023.

[140] Y. Skaf and R. Laubenbacher. Topological data analysis in biomedicine: A review. *Journal of Biomedical Informatics*, 130:104082, 2022.

[141] P. Snopov and A. N. Golubinskiy. Vulnerability detection via topological analysis of attention maps. *arXiv preprint arXiv:2410.03470*, 2024.

[142] P. Solunke, V. Guardieiro, J. Rulff, P. Xenopoulos, G. Y.-Y. Chan, B. Barr, L. G. Nonato, and C. Silva. Mountaineer: Topology-driven visual analytics for comparing local explanations. *IEEE Transactions on Visualization and Computer Graphics*, 2024.

[143] B. Sovdat. Text mining via homology. Master's thesis, UNIVERSITY OF LJUBLJANA, 2016.

[144] A. Spannaus, H. A. Hanson, G. Tourassi, and L. Penberthy. Topological interpretability for deep learning. In *Proceedings of the Platform for Advanced Scientific Computing Conference*, pages 1–11, 2024.

[145] T. Sun and B. Nelson. Topological interpretations of gpt-3. *arXiv preprint arXiv:2308.03565*, 2023.

[146] X. W. Tan, N. Tan, G. Lee, and S. Kok. The shape of reasoning: Topological analysis of reasoning traces in large language models. *arXiv preprint arXiv:2510.20665*, 2025.

[147] T. Temčinas. Local homology of word embeddings. *arXiv preprint arXiv:1810.10136*, 2018.

[148] M. Tlachac, A. Sargent, E. Toto, R. Paffenroth, and E. Rundensteiner. Topological data analysis to engineer features from audio signals for depression detection. In *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 302–307. IEEE, 2020.

[149] P. Torres-Tramón, H. Hromic, and B. R. Heravi. Topic detection in twitter using topology data analysis. *Current Trends in Web Engineering*, pages 186–197, 2015.

[150] J. F. Triki. Analysis of word embeddings: A clustering and topological approach. Master's thesis, The University of Bergen, 2021.

[151] I. Trofimov, D. Cherniavskii, E. Tulchinskii, N. Balabin, E. Burnaev, and S. Barannikov. Learning topology-preserving data representations. In *ICLR 2023 International Conference on Learning Representations*, 2023.

[152] A. Tsai, S. Subramanian, C. Liu, K. Lopez, L. Zinn-Brooks, A. Shulz, and A. Uchendu. The shape of vulnerability: How adversarial perturbations reshape the topology of language model latent spaces. In *ACL 2026 Student Research Workshop*, 2026.

[153] E. Tulchinskii, K. Kuznetsov, D. Cherniavskii, S. Barannikov, S. Nikolenko, and E. Burnaev. Topological data analysis for speech processing. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 311–315, 2023.

[154] E. Tulchinskii, K. Kuznetsov, L. Kushnareva, D. Cherniavskii, S. Nikolenko, E. Burnaev, S. Barannikov, and I. Piontkovskaya. Intrinsic dimension estimation for robust detection of ai-generated texts. *Advances in Neural Information Processing Systems*, 36, 2024.

[155] S. Tymochko, J. Chaput, T. Doster, E. Purvine, J. Warley, and T. Emerson. Con connections: Detecting fraud from abstracts using topological data analysis. In *2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 403–408. IEEE, 2021.

[156] S. Tymochko, Z. New, L. Bynum, E. Purvine, T. Doster, J. Chaput, and T. Emerson. Argumentative topology: Finding loop (holes) in logic. *arXiv preprint arXiv:2011.08952*, 2020.

[157] A. Uchendu, T. Le, and D. Lee. Topformer: Topology-aware authorship attribution of deepfake texts with diverse writing styles. *ECAI 2024*, 2024.

[158] L. Valeriani, D. Doimo, F. Cuturello, A. Laio, A. Ansuini, and A. Cazzaniga. The geometry of hidden representations of large transformer models. *Advances in Neural Information Processing Systems*, 36, 2024.

[159] H. J. van Veen. Novel topological shapes of model interpretability. In *TDA and Beyond at the 34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, 2020.

[160] K. Varadarajan and T. Songdechakraiwut. Augmenting bias detection in llms using topological data analysis, 2025.

[161] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[162] D. Voronkova, I. Trofimov, A. Dmitriev, E. Tulchinskii, E. Burnaev, and S. Barannikov. Topology of attention detects hallucinations in code LLMs, 2026.

[163] M. Vu, G. Zollicoffer, H. Mai, B. Nebgen, B. Alexandrov, and M. Bhattarai. Topological signatures of adversaries in multimodal alignments. In *Forty-second International Conference on Machine Learning*, 2025.

[164] R. Vukovic, M. Heck, B. Ruppik, C. van Niekerk, M. Zibrowius, and M. Gasic. Dialogue term extraction using transfer learning and topological data analysis. In *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 564–581, 2022.

[165] H. Wagner, P. Dłotko, and M. Mrozek. Computational topology in text mining. In *Computational Topology in Image Context: 4th International Workshop, CTIC 2012, Bertinoro, Italy, May 28-30, 2012. Proceedings*, pages 68–78. Springer, 2012.

[166] M. Wamil, A. Hassaine, S. Rao, Y. Li, M. Mamouei, D. Canoy, M. Nazarzadeh, Z. Bidel, E. Copland, K. Rahimi, et al. Stratification of diabetes in the context of comorbidities, using representation learning and topological data analysis. *Scientific Reports*, 13(1):11478, 2023.

[167] Z. Wang, D. Anshumaan, A. Hooda, Y. Chen, and S. Jha. Functional homotopy: Smoothing discrete optimization via continuous parameters for llm jailbreak attacks. In *The Thirteenth International Conference on Learning Representations*, 2025.

[168] D. Wei, M. Mao, X. Fang, and M. Chau. Short-phd: Detecting short llm-generated text with topological data analysis after off-topic content insertion. In *2nd Conference on Language Modeling (COLM)*, volume 2025, 2025.

[169] M. Wright and X. Zheng. Topological data analysis on simple english wikipedia articles. *The PUMP Journal of Undergraduate Research*, 3:308–328, 2020.

[170] X. Wu, X. Niu, and R. Rahman. Topological analysis of contradictions in text. In *Proceedings of the 45th International ACM SIGIR*, pages 2478–2483, 2022.

[171] P. Xenopoulos, G. Chan, H. Doraiswamy, L. G. Nonato, B. Barr, and C. Silva. Gale: Globally assessing local explanations. In *Topological, Algebraic and Geometric Learning Workshops 2022*, pages 322–331. PMLR, 2022.

[172] M. Xu, Q. Hu, X. Hu, S. Abousamra, X. Yu, W. Lyu, K. Qi, D. Samaras, and C. Chen. Topo-r1: Detecting topological anomalies via vision-language models. *arXiv preprint arXiv:2603.13054*, 2026.

[173] H. Yadav, T. B. Smith, P. Bubenik, and C. McCarty. What is missing from this picture? persistent homology and mixup barcodes as a means of investigating negative embedding space. *arXiv preprint arXiv:2510.14327*, 2025.

[174] X. Yan, R. Sevastjanova, S. van der Ben, M. El-Assady, and B. Wang. Explainable mapper: Charting llm embedding spaces using perturbation-based explanation and verification agents, 2025.

[175] Z. Yessenbayev and Z. Kozhirbayev. Comparison of word embeddings of unaligned audio and text data using persistent homology. In *International Conference on Speech and Computer*, pages 700–711. Springer, 2022.

[176] Z. Yessenbayev and Z. Kozhirbayev. Use of riemannian distance metric to verify topological similarity of acoustic and text domains. In *International Conference on Artificial Neural Networks*, pages 368–380. Springer, 2024.

[177] J. You, K. Dasol, and J.-H. Jung. Topological alignment of shared vision-language embedding space. In *UniReps: 3rd Edition of the Workshop on Unifying Representations in Neural Models*, 2025.

[178] Z. Yu, P. Feng, Q. Qu, H. Zhang, and Y. Zhu. Topological deep learning for speech data, 2025.

[179] I. Yurchuk and O. Gurnik. Tongue twisters detection in ukrainian by using tda. In *CEUR Workshop Proceedings*, pages 163–172, 2023.

[180] W. W. Zadrozny. Abstraction, reasoning and deep learning: A study of the" look and say" sequence. *arXiv preprint arXiv:2109.12755*, 2021.

[181] W. W. Zadrozny. A note on argumentative topology: Circularity and syllogisms as unsolved problems. *arXiv preprint arXiv:2102.03874*, 2021.

[182] H. Zhang, L. Zhang, Y. Zhang, and Z. Mao. Homology consistency constrained efficient tuning for vision-language models. *Advances in Neural Information Processing Systems*, 37:93011–93032, 2024.

[183] J. Zhang, Q. Sun, C. Zhang, X. Wang, Z. Huang, Y. Zhou, P. Zheng, C.-l. A. Tai, S.-H. Bae, Z. Ma, et al. Tda-rc: Task-driven alignment for knowledge-based reasoning chains in large language models. *arXiv preprint arXiv:2604.04942*, 2026.

[184] J. Zhang, C. Zhang, S. Chen, Y. Liu, C. Li, Q. Sun, S. Yuan, F. D. Puspitasari, D. Han, G. Wang, et al. Text summarization via global structure awareness. *arXiv preprint arXiv:2602.09821*, 2026.

[185] J. Zhang, C. Zhang, S. Chen, X. Wang, Z. Huang, P. Zheng, S. Yuan, S. Zheng, Q. Sun, J. Zou, L.-H. LEE, and Y. Yang. Learning global hypothesis space for enhancing synergistic reasoning chain. In *The Fourteenth International Conference on Learning Representations*, 2026.

[186] P. Zheng, C. Zhang, Y. Wen, W. Liu, Q. Sun, J. Mo, J. Zhang, J. Lee, T.-H. Kim, K. Liu, et al. Topology-aware layer pruning for large vision-language models. *arXiv preprint arXiv:2604.16502*, 2026.

[187] X. Zhu. Persistent homology: An introduction and a new text representation for natural language processing. In *IJCAI*, pages 1953–1959, 2013.

[188] Y. Zhu, P. Feng, S. Yi, Q. Qu, and Z. Yu. Topology-enhanced machine learning for consonant recognition. *Research Square*, 2024.

APPENDIX

Table 3: Non-theoretical applications of TDA in NLP. For the TDA techniques - PH: Persistent Homology and M: Mapper. Task categories - cl: classification, C & TM: clustering & topic modeling, S & SA: sentiment & semantic analysis, S & V: structure & visualization, H,S,& SA: health, social, & scholarly analysis, S: speech processing, MI & A: model interpretation & analysis (**PART I**)

Name	Task	Problem	Numerical Representation	Learning Type	Modality	TDA
SIFT [187]	cl	child vs. adolescent writing detection	TF-IDF	Supervised	Text	PH
[89]	cl	deepfake text detection	TF-IDF, GloVe	Supervised	Text	PH
[68]	cl	election speech feature extraction	TF-IDF	Supervised	Text	PH
[35, 137]	cl	movie genre	TF-IDF	Supervised	Text	PH
[143]	cl	distinguishing between languages	TF-IDF	Supervised	Text	PH
[134]	S & SA	legal document entailment	TF-IDF	Supervised	Text	PH
[96]	S & SA	clustering and sentiment analysis	TF-IDF, GloVe	Supervised	Text	PH
DoCollapse [59]	C & TM	keyphrase extraction	TF-IDF	Unsupervised	Text	PH
TOPOL [149]	C & TM	Twitter topic detection	TF-IDF	Supervised	Text	PH
[78]	C & TM	text summarization	TF-IDF	Unsupervised	Text	PH
[159]	cl	Propaganda tweet	TF-IDF	Unsupervised	Text	M
[40]	cl	classify Persian poems	TF-IDF	Supervised	Text	PH & M
[39]	cl	age group categorization of lonely people	TF-IDF	Supervised	Text	PH & M
[91]	cl	nursery rhyme classification from different continents - Australia, Asia, Africa, Europe, and North America	TF-IDF	Supervised	Text	PH
[61]	S & V	building document structure	Pre-trained (Word2Vec)	Unsupervised	Text	PH
BERT+TDA [170]	S & SA	contradiction detection	Pre-trained (Word2Vec)	Supervised	Text	PH
[175]	MI & A	speaker recognition & text processing	Pre-trained (Word2Vec)	Unsupervised	Speech	PH
[176]	MI & A	speaker recognition & text processing	Pre-trained (Word2Vec)	Unsupervised	Speech	PH
[30]	S & SA	building a topological search engine	Pre-trained (Word2Vec)	Unsupervised	Text	M
[64]	C & TM	document clustering and topic modeling tasks	Pre-trained (Word2Vec)	Supervised	Text	M
[121]	S & SA	word sense induction and disambiguation	Pre-trained (Word2Vec)	Unsupervised	Text	PH
[147]	S & SA	word sense induction and disambiguation	Pre-trained (Word2Vec, GloVe)	Unsupervised	Text	PH
[44]	MI & A	Geometry of textual data augmentation	Pre-trained (Word2Vec)	Supervised	Text	PH
[155]	cl	fraudulent paper detection	Pre-trained (Word2Vec, GloVe, ElMo)	Supervised	Text	PH
[110]	H,S,& SA	disease epidemic prediction	Pre-trained (Word2Vec)	Supervised	Text	PH
[173]	H,S,& SA	publication analysis	Pre-trained (Word2Vec)	Unsupervised	Text	PH
[156]	cl	finding topological loops in logical statements	Pre-trained (Word2Vec, GloVe)	Unsupervised	Text	PH
[169]	C & TM	distinguish subsets in data	Pre-trained (Word2Vec)	Unsupervised	Text	PH
[105]	S & SA	measuring the distance between the literary style of Spanish poets	Pre-trained (Word2Vec)	Supervised	Text	PH
[10]	S & SA	detecting narrative shifts	Pre-trained (Word2Vec)	Unsupervised	Text	PH
[5]	S & SA	distinguishing news articles & poems	Pre-trained (Word2Vec, GloVe)	Unsupervised	Text	PH
[51]	cl	extract the topological signatures of novelists	Pre-trained (GloVe)	Supervised	Text	PH
TIES [52]	S & SA	document categorization & sentiment analysis	Pre-trained (GloVe)	Unsupervised	Text	PH
[144]	MI & A	phenotype prediction and news group categorization	Pre-trained (GloVe)	Unsupervised	Text	M
[181]	S & SA	finding topological loops in logical statements	Pre-trained (GloVe)	Unsupervised	Text	PH
[21]	H,S,& SA	social anxiety detection	Pre-trained (GloVe)	Supervised	Text	PH
[62]	MI & A	compare cross-lingual sentence representations	Pre-trained (GloVe)	Unsupervised	Text	PH
[3]	cl	deepfake text detection	Pre-trained (GloVe)	Supervised	Text	PH
[180]	MI & A	investigates the manifestations of intelligence and understanding in neural networks	Pre-trained (GloVe)	Supervised	Text	PH
[32]	cl	fake news detection	Pre-trained (GloVe, BERT)	Supervised	Text	PH
[103]	H,S,& SA	analyzing scholarly network	Pre-trained (GloVe, BERT)	Unsupervised	Text	PH
[73]	S & SA	(1) polysemy word classification, and (2) word sense induction & disambiguation	Pre-trained (FastText)	Unsupervised	Text	PH
[150, 136]	S & SA	polysemy word	Pre-trained (FastText)	Supervised	Text	PH
PHD [154]	cl	deepfake text detection	Pre-trained (Transformers - CLS)	Supervised	Text	PH
Short-PHD [168]	cl	deepfake text detection (for short-text)	Pre-trained (Transformers - CLS)	Supervised	Text	PH
[60]	cl	deepfake text detection (for academic abstracts)	Pre-trained (Transformers - CLS)	Supervised	Text	PH
[80]	cl	deepfake text detection	Pre-trained (Transformers - CLS)	Supervised	Text	PH
[57]	MI & A	class separability estimation	Pre-trained (Transformers - CLS)	Unsupervised	Text	PH
TopoBERT [120]	S & V and MI & A	visually analyzing the fine-tuning process of a Transformer-based model	Pre-trained (Transformers - CLS)	Unsupervised	Text	M
[83]	cl	fake news detection	Pre-trained (Transformers - CLS)	Supervised	Text	PH
[31]	cl	classification of public speaking ratings from TED talks	Pre-trained (Transformers - CLS)	Supervised	Text	PH
[25]	H,S,& SA and S & V	tracking individual mental health journeys	Pre-trained (Transformers - cls)	Supervised	Text	M
[22, 65]	C & TM	topic modeling	Pre-trained (Transformers - CLS)	Unsupervised	Text	M
TOPFORMER [157]	cl	deepfake text detection	Pre-trained (Transformers - Hidden)	Supervised	Text	PH
TDA-BERTa [122]	cl	deepfake text detection	Pre-trained (Transformers - Hidden)	Supervised	Text	PH
[7]	cl	Portuguese-English language translation	Pre-trained (Transformers - Hidden)	Supervised	Text	PH

Table 4: Non-theoretical applications of TDA in NLP. For the TDA techniques - PH: Persistent Homology and M: Mapper. Task categories - cl: classification, C & TM: clustering & topic modeling, S & SA: sentiment & semantic analysis, S & V: structure & visualization, H,S,& SA: health, social, & scholarly analysis, S: speech processing, MI & A: model interpretation & analysis (**PART II**)

Name	Task	Problem	Numerical Representation	Learning Type	Modality	TDA
[48]	S & SA	polysemy word	Pre-trained (Transformers - Hidden)	Supervised	Text	M
[15]	S & SA	polysemy word	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
[56]	S & SA	semantic analysis of embeddings	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
[184]	S & SA	text summarization	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
[4]	H,S,& SA and S & V	Hate speech, Misinformation & Psychiatric disorder	Pre-trained (Transformers - Hidden)	Supervised	Text	M
PBCE [11]	MI & A	Model compression	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
Persistent Similarity [50]	MI & A	Probing layers in LLMs	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
[145]	MI & A	Correlation between sentence vectors	Pre-trained (Word2Vec, Transformers - Hidden)	Unsupervised	Text	PH
Persistence Scoring Function [26]	MI & A	captures the homology of the high-dimensional hidden representations	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
Topological Densification [49]	MI & A	zero-shot model stitching	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
TSLoss [69]	MI & A	topology loss function for prompt tuning	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
HOLE [8]	MI & A	Homological Observation of Latent Embeddings for Neural Network Interpretability	Pre-trained (Transformers - Hidden)	Supervised	Text	PH
[46]	MI & A	topology of fairness	Pre-trained (Transformers - Hidden)	Unsupervised	Text	M
[43]	MI & A	adversarial vs. non-adversarial text representation in LLMs	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
[146]	MI & A	shape of reasoning process of LLMs	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
[85, 183]	MI & A	reasoning process of LLMs using chain-of-thought prompts	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
EDTR [98]	MI & A	measures LLM confidence	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
GHS-TDA [185]	MI & A	improving reasoning process of chain-of-thought	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
[71]	MI & A	tracing reasoning process of chain-of-thought	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
[125]	C & TM	dialogue term extraction	Pre-trained (Transformers - Hidden)	Supervised	Text	PH
[77]	MI & A	polypersonality	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
[174]	MI & A	explainability of latent space	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
[47]	MI & A	topological complexity of LLM hidden space	Pre-trained (Transformers - Hidden)	Unsupervised	Text	PH
[14, 162]	cl	LLM Hallucination detection	Pre-trained (Transformers - Attention)	Supervised	Text	PH
[92]	cl	Code attribution	Pre-trained (Transformers - Attention)	Supervised	Text	PH
[79]	cl	deepfake text detection	Pre-trained (Transformers - Attention)	Supervised	Text	PH
[27, 116, 72]	S & SA	grammatical acceptability judgment	Pre-trained (Transformers - Attention)	Supervised	Text	PH
[75]	MI & A	Uncertainty estimation of model predictions	Pre-trained (Transformers - Attention)	Supervised	Text	PH
[128]	MI & A	hallucination detection in LLMs	Pre-trained (Transformers - Attention)	Supervised	Text	PH
[152]	MI & A	investigation of adversarial influence in the latent space	Pre-trained (Transformers - Attention)	Supervised	Text	PH
[160]	MI & A	bias detection	Pre-trained (Transformers - Attention)	Supervised	Text	PH
[108]	cl	spam detection, grammatical acceptability judgment, and movie sentiment analysis	Pre-trained (Transformers - Attention)	Supervised	Text	PH
[127]	cl	Authorship attribution of Japanese texts	Pre-trained (Transformers - Attention)	Supervised	Text	PH
[112]	cl	out-of-distribution detection	Pre-trained (Transformers - Attention, CLS)	Supervised	Text	PH
[115]	MI & A	estimation of weights for ensembles of classification models	Pre-trained (Transformers - Attention, CLS)	Supervised	Text	PH
[118]	MI & A	visualizes fine-tuning process to see when annotators disagree	Pre-trained (Transformers - Attention, CLS)	Supervised	Text	PH
[6]	S & V and S & SA	controversial vs. non-controversial political discourse detection	Pre-trained (Transformers - CLS)	Supervised	Text	PH
PHPS [95]	S & V and S & SA	personalized web search	Pre-trained (Transformers - CLS)	Supervised	Text	PH
[141]	cl	vulnerability detection in code	Pre-trained (Transformers - Attention)	Supervised	Text	PH
TopoHuBERT [153]	S	speaker recognition	Pre-trained (Transformers - Attention)	Supervised	Speech	PH
[164]	S & SA	dialogue term extraction	Pre-trained (Transformers - Attention)	Supervised	Text	PH
[179]	S & SA	Ukrainian tongue twisters	Symbolic Representations	Supervised	Text	PH
[76]	S	Ukrainian tongue twisters	Symbolic Representations	Supervised	Speech	PH
[41]	H,S,& SA	recipe discovery	Symbolic Representations	Unsupervised	text	PH
[19]	S	human vowel	Multi-Modal	Supervised	Speech	PH
[18]	S	infant vocalization	Multi-Modal	Unsupervised	Speech	PH
[178, 81]	S	speech recognition	Multi-Modal	Unsupervised	Speech	PH
[55, 104, 119]	S	emotion recognition	Multi-Modal	Supervised	Speech	PH
[148]	S	depression detection	Multi-Modal	Supervised	Speech	PH
[188]	S	consonants recognition	Multi-Modal	Supervised	Speech	PH
[163]	MI & A	multi-modal adversarial robustness assessment	Multi-Modal	Supervised	Text	PH
[67, 182, 117, 172]	MI & A	aligning the topological structures of image and text representations during tuning	Multi-Modal	Supervised	Text	PH
[9]	MI & A	aligning the topological structures of image and text representations during tuning	Multi-Modal	Supervised	Text	PH
[177]	MI & A	multi-modal multi-lingual topological alignment	Multi-Modal	Supervised	Text	PH

Beyond Simulate-Then-Optimize: Geothermal AI for Geothermal Dynamics Prediction, Design, and Discovery

Kunpeng Liu[1], Nori Nakata[2], Jinghan Zhang[1], Guodong Chen[2,3],
Rui Liu[4], Tao Zhe[4], Dongjie Wang[4], Xinyuan Wang[5], Hongyu Cao[5], Yanjie Fu[5,†]

[1]Clemson University, [2]Lawrence Berkeley National Laboratory,
[3]University of California Berkeley, [4]University of Kansas, [5]Arizona State University

ABSTRACT

The central bottleneck in computational geothermal science is not simulator fidelity or data scarcity—it is the abstraction itself. Geothermal energy is increasingly important to the clean energy transition, yet its computational core still follows a legacy simulate-then-optimize paradigm: a deterministic simulator is calibrated to sparse observations and then used to optimize decisions within a fixed model. Hidden inside this pipeline are three commitments—one predicted future, one mostly static operating strategy, and one fitted model per site. We argue that, for next-generation enhanced geothermal systems, the subsurface is partially observed, heterogeneous, and intervention-sensitive, and the information available to characterize it is limited. As a result, forecasting and decision-making must reason over multiple physically plausible futures under uncertainty. Our central claim is that geothermal should be reframed as an adaptive problem of inference, intervention, and discovery. Under this view, simulation becomes conditional generation over plausible reservoir futures rather than point prediction of one trajectory. Operation becomes adaptive decision making over belief states rather than offline scheduling under a presumed known state. Calibration becomes the separation of transferable physical structure from site-specific corrections rather than repeated fitting within a fixed equation class. These are not three independent engineering problems; they are three phases of a single inference cycle. This reframing matters because, in geothermal, uncertainty is not merely something to quantify; it is something operations act upon and reshape. Likewise, persistent model mismatch is not merely an engineering nuisance to suppress; it is the primary scientific signal from which missing or site-modulated physics can be discovered. We therefore organize the paper around three consequences of this reframing: generative world models of reservoir evolution, belief-state policy learning for sustainable operation, and data-to-equation discovery for transferable geophysics. Taken together, these directions define a new agenda for geothermal AI beyond faster surrogate prediction toward adaptive subsurface intelligence where inference, intervention, and discovery are intrinsically coupled.

1. INTRODUCTION

Geothermal energy is emerging as a strategically important pillar of the clean energy transition. Unlike solar and wind, geothermal can provide *firm*, 24/7 carbon-free power with a small land footprint, making it valuable for stabilizing electricity systems dominated by intermittent renewables. Beyond grid decarbonization, geothermal can boost energy security, support industrial growth, enable resilient power for data centers [41], and, in some regions, co-produce critical minerals from subsurface fluids [26].

Recent advances in horizontal drilling, high-rate completions, distributed sensing, and closed-loop field operations are turning geothermal into a more scalable and manufacturable technology stack [21, 39, 54]. Yet the computational logic used to reason about geothermal systems has changed far less. Most development workflows still follow a legacy *simulate-then-optimize* paradigm: a deterministic physics-based simulator is calibrated to sparse observations via history matching and then used to optimize well placement, injection rates, and operating schedules within a fixed model. Hidden inside this pipeline are three commitments: one predicted future, one largely static control strategy, and one fitted model per site. For next-generation enhanced geothermal systems (EGS), these commitments are increasingly untenable. The subsurface does not admit a single inevitable future: sparse observations, heterogeneous rock properties, and tightly coupled thermal–hydraulic–mechanical–chemical processes yield multiple physically plausible reservoir evolutions consistent with the same initial conditions. Operation is not a stationary optimization problem either. Injection and production decisions reshape the reservoir over time, creating path dependence, delayed feedback, and trade-offs among energy yield, reservoir longevity, and safety. Nor is calibration simply parameter fitting inside a universal model class. Geological variability entangles site-specific structure with broader governing physics, limiting transferability across locations [8, 40]. An AI agent has been developed for seamless connection to the knowledge base to digital twins, but subsurface reservoir evolution is currently a missing piece [22, 24].

We argue that geothermal should be reframed not as a forward simulation problem followed by downstream optimization, but as an integrated problem in which subsurface futures are inferred, interventions are adaptively chosen under uncertainty, and persistent mismatch is converted into scientific insight. This shift changes the computational object itself. The goal is no longer to estimate one best future under a fixed model, but to reason over plausible futures, act under partial observability, and update physical understanding as operations unfold. Under this framing, simulation becomes conditional

[†]Corresponding author: yanjie.fu@asu.edu

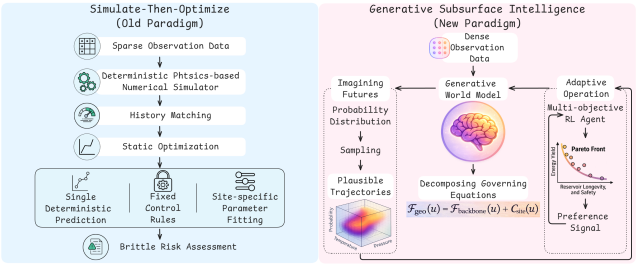

Figure 1: Contrasting geothermal AI paradigms: (A) the traditional simulate-then-optimize pipeline, based on deterministic solvers, history matching, and static control rules; and (B) the generative subsurface intelligence paradigm, which treats the subsurface as a partially observed, uncertain system and integrates generative world models, adaptive decision making, and transferable physics discovery.

generation over reservoir trajectories rather than deterministic point prediction; operation becomes belief-state policy learning rather than static scheduling; and calibration becomes the discovery of transferable structure and site-specific corrections rather than repeated parameter fitting within a fixed equation class.

Figure 1 contrasts these paradigms. The traditional approach compresses uncertainty into a single forecast and then optimizes against that forecast as if the relevant physics were already known. By contrast, the proposed paradigm models reservoir evolution through probabilistic representations induced by partial observability and unresolved heterogeneity in which inference, intervention, and discovery continually update one another. Uncertainty in reservoir evolution shapes operational decisions; operational decisions alter the latent system from which future knowledge must be inferred; and persistent mismatch between predicted and observed behavior can reveal missing or site-modulated physics.

Geothermal is a particularly revealing testbed for this broader challenge because prediction, control, and scientific discovery are inseparable in practice: uncertainty is not merely something to quantify but something operations act upon and reshape. Geothermal AI will therefore not be transformed by improved deterministic surrogates alone. As long as the field remains organized around prediction within a fixed simulator followed by offline optimization, progress will remain local, brittle, and difficult to transfer. The central limitation is not insufficient data, compute, or simulator fidelity. It is that the dominant abstraction no longer matches the geothermal systems we seek to model and operate.

The remainder of this paper develops this argument in four steps. We first show that the dominant simulate-then-optimize paradigm is insufficient. We then propose a closed-loop reframing centered on inference, intervention, and discovery. From this reframing, three research frontiers follow naturally: generative world models of reservoir evolution, belief-state policy learning for adaptive operation, and equation discovery for transferable geothermal physics. We conclude by discussing how evaluation must change if this new framing is taken seriously, and by outlining the broader scientific and deployment implications of that shift.

2. WHY THE CURRENT FRAMING IS INSUFFICIENT

The *simulate-then-optimize* paradigm has long underpinned computational geothermal science. Its limits are now often described as matters of simulator fidelity, data scarcity, or computational expense. We argue that this diagnosis is too shallow. The deeper problem is that the paradigm encodes the hidden assumptions about what geothermal systems are: it presumes one forecastable future, one mostly fixed control logic, and one locally fitted model per site. No amount of incremental model improvement fully resolves a framing mismatch of this kind. To make that claim concrete, we identify three structural mismatches that are not merely engineering bottlenecks, but conceptual failures in how the problem itself is posed.

2.1 Single-Trajectory Prediction vs. Multiple Plausible Futures

The first hidden assumption concerns prediction. At its core lies a single-trajectory worldview: given initial conditions, boundary conditions, and calibrated parameters, the simulator should return one best estimate of the future. That abstraction is reasonable when the subsurface is sufficiently characterized and uncertainty is limited. In such settings, the central scientific question is: *what will happen next?*

However, this is no longer the right question for many EGS settings. Real geothermal reservoirs are not only complex; under sparse observations and unresolved heterogeneity, they admit multiple physically plausible future evolutions. The same observable starting state can correspond to multiple physically plausible futures, as key subsurface properties are uncertain, sparsely measured, and heterogeneous. Permeability varies across fractured rock volumes, fracture connectivity is only partly known, and thermo–hydro–mechanical–chemical (THMC) processes interact nonlinearly over time [48, 51]. Small differences in local conditions can trigger qualitatively different system responses: a slight thermal perturbation may induce phase transitions, causing abrupt pressure changes; mechanical deformation can open or close flow paths, altering transport in ways that are path-dependent and partly irreversible.

A concrete example illustrates the point. At The Geysers in Northern California, the world's largest geothermal complex, injection-induced seismicity remains notoriously difficult to predict. Identical injection protocols applied to neighboring wells can produce qualitatively different seismic responses, because subsurface fracture connectivity and stress conditions differ in ways that no deterministic model can resolve from sparse surface observations [38]. This is not a failure of calibration; it is a failure of the single-trajectory abstraction. More broadly, ensemble simulations with TOUGH2 under different permeability realizations at EGS sites routinely show trajectory divergence exceeding an order of magnitude in predicted flow rates over decadal horizons [7,44], underscoring that the deterministic framing systematically understates subsurface ambiguity [4, 61].

As a result, no single trajectory is sufficient. Even if a deterministic simulator is accurate on average, it maps a structured set of plausible futures into a single forecast. Deterministic AI surrogates inherit the same limitation: they may emulate simulators efficiently, but still return point predictions, whereas the underlying scientific object is a *distribution* over possible futures [30, 60]. This distinction is critical because geothermal decisions are high-stakes and long-horizon. Operators must reason not only about what

one model predicts, but about what *could* happen: which futures are plausible, which are risky, and how uncertainty propagates over decades of operation.

The core limitation, therefore, is not any specific simulator, but the deterministic prediction abstraction itself. When the system admits many physically plausible evolutions, the computational task should be reframed from predicting a single trajectory to describing a conditional family of futures. The cost of retaining the deterministic abstraction is not merely reduced realism; it is a systematic compression of risk, in which structurally different but plausible reservoir futures are collapsed into a single forecast.

2.2 Static Control vs. Adaptive, Multi-Objective Operation

The second hidden assumption concerns control. The dominant paradigm treats operation as a scheduling problem: devise a plan largely offline, perhaps revise it occasionally, and then execute it through fixed schedules, threshold rules, or limited re-optimization. That abstraction is reasonable when reservoir conditions evolve slowly, observations are sufficiently informative, and objectives are narrow and stable. In such settings, the central control question is: *what is the best schedule under the current model?*

However, EGS operation is not a one-time scheduling problem. It is a long-horizon decision problem in which actions reshape the system being controlled. Injection and production decisions alter pressure fields, temperature gradients, fracture behavior, and long-term reservoir sustainability.

Moreover, the objective is inherently multi-dimensional [12]. Operators must balance energy yield, reservoir longevity, pressure stability, and mechanical integrity, often under evolving economic, regulatory, and safety constraints [42]. These trade-offs evolve over time: strategies that maximize short-term heat extraction may accelerate thermal depletion, destabilize pressure, or increase downstream risk over decades [40].

Compounding this challenge, the system is only *partially observed*. Unlike many engineered systems, the geothermal reservoir state is not directly observable. Measurements are sparse, noisy, and indirect, and key variables must be inferred rather than directly measured. As a result, geothermal control is fundamentally a problem of acting under uncertainty about the current subsurface state, not merely optimizing over a known state. Existing strategies such as fixed injection rates or threshold-based adjustment rules are poorly suited to this setting [18, 25]: they do not explicitly reason about uncertainty, they do not adapt as the reservoir evolves, and they often encode implicit short-term objectives at the expense of long-term system health.

The limitation, therefore, is not that current controllers are insufficiently tuned, but the current control is the inappropriate abstraction for a partially observed, evolving, multi-objective system. What is needed is a formulation in which policies adapt continuously to uncertain and changing reservoir conditions, while explicitly trading off competing objectives over long operational horizons. The cost of retaining the static-control abstraction is not merely suboptimal scheduling; it is a failure to recognize that operational decisions are epistemic as well as engineering actions, because they reshape the latent system from which future decisions must be made.

2.3 Site-Specific Calibration vs. Transferable Physics

The third hidden assumption concerns knowledge accumulation. The dominant paradigm treats calibration as a local fitting problem: adjust model parameters until simulated outputs match observations at one site. That approach is reasonable if each reservoir is treated as an isolated engineering project and if sufficient local data are available. In that framing, the central question is: *how can we fit this model to this site?*

However, this framing creates a scalability problem. Each new geothermal site requires costly recalibration, often with limited data, because fitted parameters absorb multiple sources of variation at once [16]. They reflect not only universal physical structure, such as conservation laws and Darcy-type flow, but also local geological idiosyncrasies, such as fracture geometry, stress sensitivity, and site-specific permeability corrections. Thus, calibration entangles what should transfer across sites and what should remain site-specific.

This entanglement is costly both scientifically and operationally. A model calibrated at Utah FORGE may not transfer to sites in Nevada or Texas, even when the underlying physics is largely shared. The issue is not that one site obeys different laws of nature than another, but that the current calibration pipeline lacks a mechanism to separate reusable physical structure from local corrections. Consequently, each site is treated almost as a new problem. Progress becomes site-locked: knowledge accumulates locally with weak transfer across sites.

For geothermal to scale as a science and an industry, calibration must become more than parameter estimation. The computational goal is to decompose governing behavior into two components: a transferable physical backbone that captures shared structure, and site-specific components that can be learned or discovered independently. Without such a separation, AI-for-geothermal remains trapped in a cycle of local fitting rather than cumulative scientific learning. The cost of retaining the site-specific calibration abstraction is not merely repeated labor; it is the inability of the field to accumulate knowledge across projects, because reusable physical structure remains entangled with local corrective fitting.

These three hidden assumptions define the ceiling of the current paradigm. The central bottleneck is not the fidelity of any individual simulator, controller, or calibration routine, but the abstraction that organizes them. Next-generation geothermal systems require distributional reasoning, adaptive decision-making under partial observability, and separable, transferable physics. To move beyond that ceiling, the field needs not just better tools, but a different computational formulation. We turn to that formulation next.

3. OUR PERSPECTIVE: GEOTHERMAL AS AN ADAPTIVE, COUPLED PROBLEM OF INFERENCE, INTERVENTION, AND DISCOVERY

3.1 Core Reframing

The structural mismatches identified above point to a deeper conclusion: geothermal is not best understood as a pipeline of simulation, optimization, and calibration, but as a coupled

and iterative system in which these functions continually inform and update one another. Uncertainty in subsurface evolution shapes operational decisions; operational decisions alter the reservoir state and the information subsequently observed; and persistent mismatch between predicted and observed behavior can reveal missing or site-modulated physics. In this sense, geothermal is fundamentally a problem of inference, intervention, and discovery. Simulation, control, and calibration should therefore be understood not as three independent engineering problems, but as three phases of a single inference cycle.

The value of AI is therefore not simply to accelerate existing modules, but to support a different computational object altogether. Rather than predicting one best future and optimizing against it, the field should reason over families of plausible futures, choose interventions under partial observability, and treat residual mismatch as a source of scientific learning. This reframing leads to three coupled consequences: (1) simulation should be treated as conditional generation over physically plausible reservoir trajectories; (2) operation should be treated as adaptive policy learning over belief states under multiple objectives; and (3) calibration should be treated as the discovery of transferable physical structure plus site-specific corrections, rather than repeated fitting within a fixed equation class.

3.2 Conceptual Mapping: Old Objects, New Roles

Under this perspective, every core object in geothermal computational science acquires a new role:

- **Simulation** is no longer a deterministic forward solve of a numerical PDE system, but *conditional distribution learning* over feasible spatiotemporal trajectories, $p_\theta(\mathbf{u}_{0:T}|\mathbf{c})$, where $\mathbf{u}_{0:T}$ denotes the evolution of temperature, pressure, phase saturation, and fluid velocity, and $\mathbf{c}$ encodes geological conditions and boundary controls.
- **Operation/control** is no longer the application of fixed injection schedules or threshold-based rules, but *adaptive multi-objective policy learning* under latent-state uncertainty, $\pi_\phi(\mathbf{a}_t|\mathbf{b}_t)$, where $\mathbf{b}_t = p(\mathbf{z}_t|o_{0:t}, \mathbf{a}_{0:t-1})$ is a belief distribution over latent reservoir states $\mathbf{z}_t$ inferred from sparse, noisy observations.
- **Calibration/history matching** is no longer parameter fitting within a fixed governing equation, but *automated equation discovery* via a backbone–calibration decomposition, $\mathcal{F}_{\text{geo}}(\mathbf{u}) = \mathcal{F}_{\text{backbone}}(\mathbf{u}) + \mathcal{C}_{\text{site}}(\mathbf{u})$, where the backbone captures universal conservation laws and the site-specific term is discovered through symbolic regression.

These remappings carry immediate corollaries: uncertainty quantification becomes intrinsic to the generative framework rather than a post hoc add-on; site characterization becomes representation learning over multimodal evidence; and cross-site transfer becomes systematic comparison of discovered $\mathcal{C}_{\text{site}}$ terms rather than ad hoc expert judgment (Table 1). The three core mappings are the most consequential because, together, they instantiate the closed loop of inference, intervention, and discovery.

3.2.1 Simulation as conditional distribution learning

The central insight is that high-fidelity geothermal simulation is computationally prohibitive, real-world data are sparse, and subsurface physics admits multiple physically plausible evolutions under the same conditions, because subsurface structure and state are only partially characterized, the governing system admits multiple physically plausible evolutions consistent with the available evidence [7, 13, 48].

Diffusion-based generative models provide a natural solution: they treat generation as a stochastic search over a probability landscape, learning to reverse a noise-corruption process to sample from a conditional trajectory distribution. Such generative models has demonstrated their capabilities to extract subsurface elastic properties [5, 46]. Just as diffusion models in computer vision generate diverse, high-quality images from noise, a geothermal diffusion model generates diverse, physically plausible reservoir trajectories conditioned on geological parameters and boundary controls. Each trajectory represents a distinct hypothesis of subsurface evolution rather than a noisy variant of a single prediction. This reframes uncertainty quantification from a separate analytical step into the generation process itself.

3.2.2 Operation as belief-state policy learning

Geothermal operation is a long-horizon decision process with delayed and often irreversible consequences. Injection decisions alter temperature gradients, fracture permeability, and mechanical stress, with effects that may only manifest years later [40]. Under this framing, we model the system as a partially observable Markov decision process (POMDP), where policies act on *belief states*—probability distributions over latent reservoir conditions—rather than fully observed states. Multi-objective learning makes trade-offs between energy yield, pressure stability, thermal longevity, and mechanical integrity explicit, yielding families of Pareto-efficient policies rather than a single operating strategy. Operators can then select among these policies based on real-time constraints and risk tolerance, shifting from reactive, site-specific heuristics to proactive, generalizable, and uncertainty-aware control.

3.2.3 Calibration as equation discovery

Traditional calibration adjusts parameters within a fixed set of governing equations, conflating universal physical laws (e.g., conservation of energy, Darcy flow) with site-specific geological effects (e.g., fracture geometry and stress-dependent permeability). Under the new framing, we explicitly decompose the governing equations:

$$\mathcal{F}_{\text{geo}}(\mathbf{u}) = \mathcal{F}_{\text{backbone}}(\mathbf{u}) + \mathcal{C}_{\text{site}}(\mathbf{u}), \tag{1}$$

where $\mathcal{F}_{\text{backbone}}$ encodes universal laws shared across sites, and $\mathcal{C}_{\text{site}}$ is a *learnable* site-specific correction discovered via data-driven equation discovery (e.g., symbolic regression). The resulting $\mathcal{C}_{\text{site}}$ terms are interpretable expressions rather than opaque neural weights, enabling comparison across sites, identification of shared mechanisms, and construction of transferable geothermal knowledge. This shift elevates calibration from numerical fitting to scientific discovery.

3.3 Why This Is a Paradigm Shift, Not a Better Tool

This is not merely a stronger surrogate model applied to the same task. It changes the unit of modeling—from a single trajectory to a distribution; the role of control—from static

Dimension	Simulate-then-Optimize	Generative SubsurfaceIntelligence
Problem formulation	Single deterministic prediction	Conditional distribution over trajectory space
Representation	Fixed PDE discretization	Learned latent dynamics with physics structure
Optimization target	Maximize heat extra under fixed mode	Multi-objective policy over belief states
Uncertainty handling	Post hoc sensitivity analysis	Intrinsic: each sample is a hypothesis
Calibration	Parameter fitting in fixed equations	Equation discovery: backbone + site terms
Cross-site transfer	Re-calibrate from scratch	Compare discovered calibration terms
Evaluation criteria	Prediction error at one site	Adaptation, robustness, physical consistency, transfer

Table 1: Comparison between the legacy simulate-then-optimize pipeline and the proposed closed-loop subsurface intelligence framing for geothermal systems.

optimization to belief-state-conditioned policy learning; and the interface between learning and physics—from parameter fitting to equation discovery.

Under the old paradigm, the computational challenge was "solve this PDE faster." Under our perspective, it becomes "learn the distribution of what the subsurface *could do*, reason about what it *should do*, and discover *why* it behaves differently across sites."

This distinction reshapes the research agenda for geothermal AI: not faster solvers for fixed equations, but new formulations, learning architectures, and evaluation criteria.

A concrete litmus test clarifies the stakes: under the old framing, a perfectly calibrated deterministic simulator at one site would be considered a solved problem. Under ours, it would be considered a *failure mode*—because it has absorbed site-specific effects into opaque parameters, foreclosed distributional reasoning, and produced knowledge that cannot transfer to the next site. If this claim is correct, then the community's default measure of success (single-site prediction error) is not merely incomplete but actively misleading, because minimizing it drives the field deeper into the paradigm we argue should be replaced.

3.4 What This Perspective Is Not Claiming

This perspective does not argue that numerical simulators should be discarded, or that geothermal can be treated as a purely data-driven problem. Physics-based simulators remain essential sources of structure, supervision, and validation. Nor do we claim that geological variation can be removed through AI alone. The challenge is not to eliminate site specificity, but to represent uncertainty, intervention dependence, and transferable physical structure more faithfully than the current pipeline allows.

3.5 The Structural Opportunity: Why Now

This reframing is now possible due to the convergence of three developments. *First*, advances in generative AI—particularly diffusion models, score-based generative modeling, and physics-informed neural operators—have demonstrated the ability to learn complex, high-dimensional distributions over spatiotemporal fields, from weather prediction to molecular dynamics [23, 29, 43, 50, 66]. These tools can be adapted to geothermal systems, provided appropriate inductive biases (e.g., physical constraints and coupling-aware architectures) are incorporated. *Second*, the geothermal sensor ecosystem is undergoing rapid transformation. Dense fiber-optic sensing (distributed temperature, acoustic, and strain sensing), microseismic monitoring, and downhole geochemical sampling now generate high-resolution spatiotemporal data streams at sites such as Utah FORGE [54]. Data density is approaching the threshold at which learning-driven approaches become practical rather than aspirational. *Third*, the urgency of clean energy deployment creates both economic pull and policy support. The DOE Enhanced Geothermal Shot, targeting $45/MWh by 2035, demands not incremental improvements in drilling or simulation but a step change in how geothermal systems are computationally understood, operated, and scaled [52]. The transition from oil and gas workforces to geothermal industries further amplifies the need for AI-enabled, transferable operational knowledge. These developments make it both possible and necessary to move from deterministic, site-locked, and static computation to generative, transferable, and adaptive subsurface intelligence.

4. RESEARCH FRONTIERS OPENED BY THIS REFRAMING

The three directions below are not parallel wish lists or loosely related technical opportunities. They are the three necessary research consequences of the coupled reframing above. Once geothermal is treated as a problem of inference, intervention, and discovery, the field must learn to: (1) generate plausible reservoir futures rather than point forecasts; (2) choose actions over belief states rather than fixed schedules; and (3) convert persistent residual mismatch into interpretable physical insight rather than absorb it into opaque calibration. Taken together, they form a closed-loop intelligence architecture in which each component changes the object that the next component must reason about, aligning naturally with the recent shift toward agentic AI systems that couple reasoning, planning, and iterative feedback [17, 32, 35, 36, 56, 57, 71, 75, 76]. Read this section not as a menu of tools, but as a decomposition of that architecture.

4.1 World Models That Generate Reservoir Futures

Once geothermal simulation is reframed as reasoning over plausible futures rather than forecasting a single trajectory, the first technical frontier is the construction of *generative world models* for subsurface systems. Their role is not merely to emulate a simulator faster, but to represent the conditional distribution of what the reservoir *could do* under uncertain geology and chosen controls. Figure 2 illustrates this generative world-modeling perspective, in which sparse site information and control conditions define a conditional distribution over physically plausible reservoir trajectories. This matters because failing to reason over the *distribution* of reservoir trajectories, rather than a single prediction, produces brittle risk assessments, overconfident operational

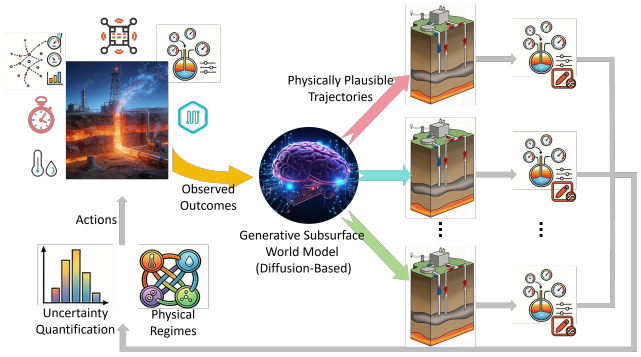

Figure 2: A generative world model for subsurface geothermal systems. Given initial conditions, control parameters, and sparse site data, a diffusion-based model generates 100+ physically plausible trajectories within a fractured subsurface volume, constrained by conservation laws and thermodynamic consistency to enable reliable uncertainty quantification.

planning, and a false sense of certainty about long-horizon reservoir behavior.

The core question is therefore not "how can we speed up simulation?" but: *how can we learn conditional generative models* $p_\theta(\mathbf{u}_{0:T}|\mathbf{c})$ *over high-dimensional, multi-physics geothermal trajectories that are diverse, physically consistent, and computationally useful for downstream decisions?*

Why this frontier is now technically plausible. Diffusion models have shown a remarkable ability to learn complex, high-dimensional distributions [23,50]. Recent work on latent diffusion [3,47] and physics-informed score matching suggests that operating in a learned latent space can significantly reduce computational cost while preserving physical fidelity. Closely related work demonstrates that diffusion-based generation can be coupled with causal stability constraints to yield robust selections under distribution shift [55]. Preliminary work on Brownian Bridge-augmented frameworks for CO_2 storage simulations demonstrates that generative models can produce physically consistent trajectories with higher fidelity than deterministic surrogates [1]. Work on the supply chain also demonstrates the importance of the combination of simulation and generative models [2, 9, 10]. Neural operators (Fourier Neural Operators, DeepONet) provide resolution-independent function mappings and can serve as efficient backbone architectures [29, 37].

Open Questions. Despite recent progress, three core challenges remain unresolved. The first concerns how to construct latent representations that respect the heterogeneous coupling structure of THMC processes, where slow thermal diffusion, fast pressure propagation, and discrete phase transitions interact across scales [11, 65]. Recent advances in RL-guided Transformer feature construction [20] and graph-walk-based feature–variable alignment [19] illustrate complementary strategies for representing such heterogeneous coupling within learned latent spaces. Closely related is the question of how physical constraints—including conservation laws, thermodynamic consistency, and stress limits—can be built directly into generative dynamics, rather than imposed as external corrections. More broadly, it remains unclear how generative models can represent regime-dependent behavior (e.g., liquid, two-phase, and steam systems) without blurring physically distinct modes or collapsing diversity across operating conditions. Analogous challenges in language-model-driven generation have been addressed through diversity-controlled augmentation [59] and structured exploration of under-covered hypothesis regions [68], suggesting transferable strategies for preventing mode collapse in physical trajectory generation. One promising formulation is a regime-conditioned mixture:

$$p_\theta(\mathbf{u}_{0:T}|\mathbf{c}) = \sum_r p_\theta(\mathbf{u}_{0:T}|r,\mathbf{c})\, p_\theta(r|\mathbf{c}), \tag{2}$$

where $r \in \{\text{liquid, two-phase, steam}\}$ indexes thermodynamic regimes and $p_\theta(r|\mathbf{c})$ is a learned regime classifier conditioned on site context. This decomposes the generation problem into regime identification and within-regime trajectory sampling, preventing cross-regime mode collapse while preserving the ability to reason about regime transitions under changing operational conditions.

What would count as a real shift. Success would not simply be faster simulators, but a qualitative shift in how geothermal systems are modeled and used in practice. Instead of producing a single trajectory over days of computation, models would generate diverse, physically consistent futures on demand, each reflecting different plausible evolutions of the reservoir under uncertainty. Engineers could interrogate these trajectory ensembles to assess risk, compare intervention strategies, and reason about system behavior across a range of operating conditions. In this setting, simulation becomes a tool for exploring distributions of outcomes, rather than committing to a single deterministic forecast.

4.2 Belief-State Policy Learning for Sustainable Reservoir Management

Once geothermal operation is reframed as acting under partial observability rather than executing a fixed schedule, the second frontier is the development of *belief-state policies* that adapt to evolving reservoir conditions over decades-long horizons. Their role is not merely to optimize extraction, but to decide what the system *should do* when the state is uncertain, objectives conflict, and today's action changes tomorrow's reservoir. Figure 3 illustrates this shift from static scheduling to belief-state policy learning, emphasizing adaptive action under partial observability, multi-objective trade-offs, and long-horizon reservoir outcomes. This matters because current practice—static injection rules, threshold heuristics, and single-objective optimization—systematically sacrifices long-term reservoir sustainability for short-term energy yield, a trade-off that becomes more costly as EGS deployments scale.

The core question is: *how can we learn families of Pareto-efficient policies* $\pi_\phi(\mathbf{a}_t|\mathbf{b}_t)$ *that reason over belief states, balance energy yield, reservoir longevity, pressure stability, and mechanical safety, and generalize across geological settings?*

Why this frontier is now technically plausible. Multi-objective reinforcement learning (MORL) provides frameworks for learning Pareto-optimal policy families parameterized by preference vectors [15, 31, 33, 34, 58, 62, 63]. Complementary techniques control exploration depth based on belief uncertainty itself, deciding when to deepen, expand, or terminate trajectory exploration [70]. Generative world models from Direction 1 can serve as interactive environments for policy training, enabling efficient evaluation of long-horizon outcomes without costly numerical simulations. Belief-state

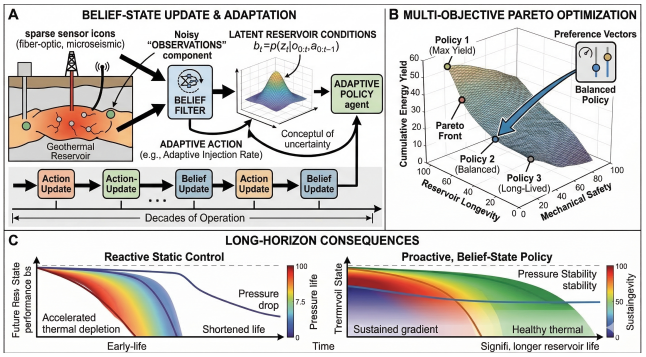

Figure 3: Belief-state policy learning for geothermal operation. (A) A belief filter updates latent reservoir states (b_t) from sparse, noisy observations to produce adaptive actions. (B) Multi-objective optimization identifies Pareto-efficient policies balancing energy yield, reservoir longevity, and mechanical safety. (C) Over long horizons, proactive belief-state policies maintain thermal and pressure stability, achieving longer reservoir life than reactive static control.

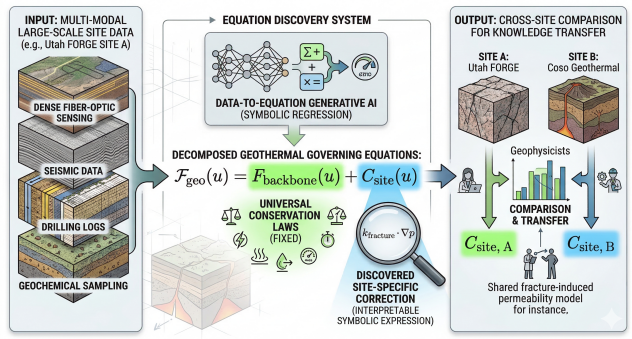

Figure 4: Data-to-equation scientific discovery workflow. Multi-modal site data (e.g., Utah FORGE) are processed by symbolic regression-based generative AI to decompose geothermal governing equations into fixed universal backbone laws ($F_{backbone}$) and site-specific corrections (C_{site}). This interpretability enables cross-site knowledge transfer by allowing geophysicists to compare discovered terms and identify shared geological mechanisms across distinct reservoirs.

methods from the POMDP literature provide principled mechanisms for decision-making under partial observability [28]. In data-sparse regimes, prototype-based reward modeling reduces sample complexity of policy alignment while preserving fidelity to preference signals [69], a property essential when geothermal operational data are too scarce to support fully online reward estimation. Distributionally robust optimization provides tools to ensure policy transfer across sites by optimizing worst-case performance over subsurface distributions [45]. Adaptive weighting strategies allow value-based policies to track non-stationary environments online, a property essential under decadal reservoir drift [75].

Open Questions. Despite recent progress, key challenges emerge when moving from modeling to decision-making. A central difficulty is maintaining coherent belief representations under partial observability, where geothermal measurements are sparse, indirect, and temporally irregular, making state estimation inherently uncertain and history-dependent. At the same time, policies must remain reliable under substantial variation across reservoirs, raising the question of how control strategies can generalize despite differences in permeability structure, fracture geometry, and stress conditions. Finally, geothermal operation unfolds over decades, forcing a tight coupling between learning and control: actions not only extract energy but also shape future system knowledge, making it unclear how to balance information acquisition with long-term productivity. For cross-site transfer specifically, a minimax robust formulation provides a concrete starting point:

$$\max_{\pi} \min_{k \in \mathcal{K}} J^{(k)}(\pi), \tag{3}$$

where $\mathcal{K}$ indexes a family of site-specific world models and $J^{(k)}(\pi)$ is the multi-objective return under model k. This transforms the vague desideratum of "robustness" into a well-defined optimization problem whose solution is a policy that performs acceptably across geological settings, even if it sacrifices peak performance at any single site.

What would count as a real shift. In practice, success would be reflected in a shift from a set of fixed operating rules to adaptive, state-aware decision-making. Operators would no longer rely on predetermined injection schedules, but instead adjust actions in response to evolving reservoir beliefs, with an explicit understanding of the trade-offs between energy production, reservoir longevity, and safety. Rather than committing to a single operating strategy, they could navigate a spectrum of policies, selecting or adapting strategies as new information becomes available. Over time, such policies would not only improve immediate performance but also steer the reservoir toward more stable and sustainable operating regimes.

4.3 From Calibration to Discovery: Data-to-Equation Generative Physics

Once calibration is reframed as the interpretation of structured residuals rather than the repeated fitting of local parameters, the central scientific challenge becomes extracting reusable physical insight from the prediction–reality gap. The third frontier is therefore developing AI systems that can *automatically discover interpretable expressions* for site-specific geothermal physics while preserving a shared physical backbone. Their role is not merely to fit one site better, but to explain *why* a reservoir behaves differently and to turn residual mismatch into transferable scientific knowledge. Figure 4 summarizes this data-to-equation workflow, where structured residuals are used to discover interpretable site-specific corrections on top of a shared physical backbone. This matters because current practice—site-by-site parameter fitting—absorbs missing structure into opaque local parameters and thereby blocks cumulative learning across geothermal projects.

The core question is: *how can we decompose geothermal governing equations into backbone physics $\mathcal{F}_{backbone}$ and site-specific terms $\mathcal{C}_{site}$, discover $\mathcal{C}_{site}$ as interpretable symbolic expressions from data, and use cross-site comparison to extract generalizable geophysical insights?*

Why this frontier is now technically plausible. Symbolic regression has advanced rapidly, with methods from ge-

netic programming to neural-guided search and transformer-based equation generation [6,14,27]. Recent data-to-equation approaches suggest that foundation models can be adapted to low-data symbolic regression settings, while reinforcement feedback can further align equation generation with downstream numerical fitness and domain-specific structure [64,67]. Beyond raw equation search, retrieval- and LLM-augmented feature generation pipelines provide structured priors that constrain the discovery space to interpretable, domain-meaningful terms [73, 74]. The backbone–calibration decomposition $\mathcal{F}_{\text{geo}}(\mathbf{u}) = \mathcal{F}_{\text{backbone}}(\mathbf{u}) + \mathcal{C}_{\text{site}}(\mathbf{u})$ provides a structural prior that constrains search space: the backbone is fixed by conservation laws, and only the residual site-specific term must be discovered. Physics-informed residuals from the generative simulator (Direction 1) and policy-highlighted anomalies from the adaptive controller (Direction 2) guide discovery to physically meaningful regions of equation space.

Open Questions. What distinguishes equation discovery from conventional calibration is not only expressiveness, but the need for interpretability and scientific validity. In practice, current symbolic methods struggle to move beyond simple or weakly coupled forms, raising the question of how complex, multi-term interactions across the THMC state space can be discovered without losing tractability. Even when candidate expressions are found, their status remains ambiguous: fitting observational data is insufficient, yet there is no clear criterion for when a discovered equation should be regarded as physically meaningful rather than incidental. A further complication is that these expressions are inherently site-specific; without a systematic way to relate them across reservoirs, it is unclear how individual discoveries accumulate into broader geological understanding.

A key insight, largely absent from current symbolic regression practice, is that calibration terms are not unique: multiple functional forms may explain the same observations equally well. This means equation discovery should itself be *generative*—learning a distribution over candidate equations $p(\mathcal{C}_{\text{site}}|\text{data})$ rather than returning a single best-fit expression. A distributional treatment would quantify epistemic uncertainty over governing physics, enable model averaging for more robust prediction, and expose structural degeneracies that point to which additional measurements would most effectively disambiguate competing hypotheses.

What would count as a real shift. Success would be evident not in improved predictive accuracy alone, but in how results are used and interpreted. Instead of treating each site as an isolated calibration problem, practitioners would obtain explicit mathematical descriptions of site-specific behavior that can be interrogated, compared, and debated. These expressions would serve as hypotheses about underlying mechanisms, guiding further analysis rather than acting as fixed outputs. Over time, patterns across sites will reveal recurring structures—shared functional forms for stress-dependent permeability, common fracture-flow corrections—enabling domain experts to move from empirical fitting toward a unified, transferable understanding of subsurface physics.

4.4 Why These Three Cannot Be Separated: The Residual-as-Signal Principle

The three directions above are not independent research programs that happen to share a domain. They are successive phases of one feedback-coupled intelligence cycle, linked by a principle we call *residual-as-signal*:

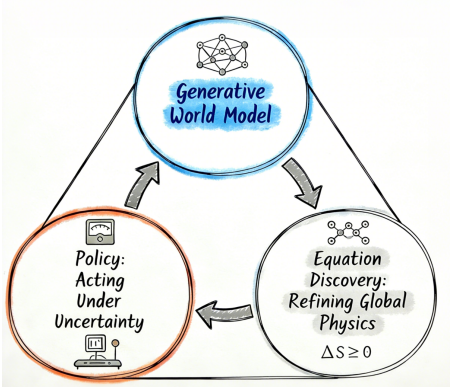

Figure 5: A Coupled and Adaptive Paradigm: from Model to Policy to Equation Discovery.

> *The residual between a world model's predicted distribution and operationally observed outcomes is not merely an error to be minimized—it is the primary scientific signal from which missing site-specific physics can be discovered.*

This is the paper's deepest claim. Residuals are not merely evidence of imperfect fitting; they are structured, decision-dependent traces of what the current representation fails to capture—and they are the only place where scientific discovery enters the computational loop.

This principle creates an irreducible information flow among the three components (Figure 5):

1. **World model → Policy.** The generative world model produces a trajectory distribution $p_\theta(\mathbf{u}_{0:T}|\mathbf{c})$ that serves as the policy's training environment. Without distributional simulation, the policy has no uncertainty-aware sandbox in which to learn.

2. **Policy → Equation discovery.** When the learned policy is deployed, the actions it takes expose a *prediction–reality gap*: systematic discrepancies between the world model's anticipated trajectories and the reservoir's actual response. These residuals are not noise, but structured signatures of physics missing from the world model.

3. **Equation discovery → World model.** Discovered site-specific terms $\mathcal{C}_{\text{site}}$ feed back into the world model, refining its generative distribution. A more accurate world model, in turn, produces better-calibrated training environments for the policy and exposes *subtler* residuals for the next round of discovery.

This closed loop has a concrete consequence: *each component improves the others.* A world model trained in isolation will plateau because it cannot access the operationally induced residuals that reveal missing physics. A policy trained on a static simulator will degrade under distribution shift because it has no mechanism for model refinement. Equation discovery without policy-driven exploration will find only the most obvious corrections, missing the subtle site-specific effects that only emerge under active reservoir management.

The residual-as-signal principle is what distinguishes this agenda from three parallel mini-surveys. It is also the paper's

core differentiator relative to existing geothermal AI reviews: we do not merely propose that AI can help with simulation, control, and calibration separately, but that these three tasks are *informationally coupled* in a way that demands joint treatment.

5. RETHINKING EVALUATION: TOWARD GEOGENBENCH

If the closed-loop perspective is correct, then conventional evaluation protocols for computational geothermal science are not just incomplete; they optimize for the success criterion. Current benchmarks primarily measure *prediction error at a single site*—how closely a simulator or surrogate matches observed data under fixed conditions. But under the new paradigm, prediction accuracy at one site under one condition is necessary and still insufficient. The real question is not whether a model reconstructs one historical trajectory, but whether a coupled system of world modeling, decision making, and discovery produces better uncertainty, better interventions, and more reusable knowledge. We therefore propose **GeoGenBench**, a community benchmark designed to evaluate the full integrated intelligence stack. GeoGenBench is organized around five evaluation dimensions, each with concrete metrics:

- **Distributional fidelity.** Does the generative model produce a diverse, calibrated distribution of trajectories? *Metric:* Continuous Ranked Probability Score (CRPS), which jointly penalizes miscalibration and lack of sharpness, evaluated over held-out THMC trajectories.
- **Physical consistency under sparse data.** Do generated trajectories satisfy conservation laws and coupling constraints even under data scarcity? *Metric:* conservation violation rate (fraction of samples violating energy balance, mass conservation, or thermodynamic monotonicity beyond a tolerance ϵ).
- **Policy robustness and transfer.** Can policies generalize across geological settings? *Metric:* Transferability Score, defined as the ratio of multi-objective return when a policy trained at Site A is deployed at Site B to the return of a policy trained directly at Site B: $\mathrm{TS}(A \to B) = J^{(B)}(\pi_A)/J^{(B)}(\pi_B)$.
- **Multi-objective trade-off quality.** Does the policy produce a well-distributed Pareto front across competing objectives (energy yield, longevity, safety)? *Metric:* Hypervolume indicator, measuring the volume of objective space dominated by the learned Pareto front.
- **Refinement gain from discovery.** Does incorporating discovered $\mathcal{C}_{\text{site}}$ terms improve the world model? *Metric:* Refinement Gain, defined as the reduction in CRPS after augmenting the world model's backbone with discovered site-specific terms: $\mathrm{RG} = 1 - \mathrm{CRPS}_{\text{refined}}/\mathrm{CRPS}_{\text{baseline}}$.

Data and protocol. We envision GeoGenBench built on data from Utah FORGE (the DOE's flagship EGS research site) and The Geysers (the world's largest operating geothermal complex), providing complementary geological settings for transfer evaluation. The evaluation protocol follows the iterative inference–decision–discovery cycle: train a generative world model at Site A → learn a belief-state policy → transfer to Site B → measure transferability score → discover $\mathcal{C}_{\text{site}}$ from the prediction–reality residual → refine the world model → measure refinement gain. This protocol evaluates not only individual components, but their *joint* performance as an integrated system.

Concretely, benchmark design should move beyond single-site reconstruction. One class of tests should evaluate whether a model trained in several fields can adapt to a new site with limited calibration while preserving the quality of the uncertainty. A second class should evaluate whether belief-state policies remain robust under hidden shifts in permeability structure, fracture connectivity, or sensing sparsity. A third class should test whether discovered correction terms remain stable across resampling, data subsets, and nearby sites, indicating that they capture repeatable mechanisms rather than incidental fits.

A named, concrete benchmark with standardized metrics and shared datasets would give the community a common target, accelerating progress across all three research directions.

Trustworthiness in high-stakes deployment. A natural concern about a generative reframing of geothermal AI is whether such models can be made trustworthy enough for safety-critical operational use, where induced seismicity, pressure excursions, and thermal short-circuiting are partially irreversible. Trustworthiness here does not arise from any single safeguard but from four commitments intrinsic to the closed loop. *First*, physical constraints—conservation laws, thermodynamic monotonicity, and stress envelopes—should be built into generative dynamics rather than imposed as post-hoc filters [3], so that every sampled trajectory is feasible by construction. *Second*, the value of a trajectory ensemble lies in calibration rather than diversity: a model that produces many visibly different futures is not trustworthy unless its predicted distribution achieves nominal coverage of held-out outcomes (CRPS, reliability curves), with conformal prediction [49] offering one route to finite-sample coverage guarantees. *Third*, belief-state policies must operate under safety envelopes—e.g., CVaR bounds on induced seismicity and pressure excursions [72], combined with the minimax formulation in Eq. (3)—and route irreversible actions through human-in-the-loop oversight; the Geysers example in Section 2.1 is precisely the regime where bounding the tail matters more than optimizing the mean. *Fourth*, the residual-as-signal principle makes the system auditable: persistent mismatch is surfaced for equation discovery rather than absorbed into opaque weights, and GeoGenBench is designed to falsify the framing itself if it fails to deliver better calibration, transfer, and reusable insight than deterministic baselines. In high-stakes domains, falsifiability is itself a trustworthiness property.

A falsifiability commitment. GeoGenBench is designed not only to measure progress but to test whether the closed-loop framing itself is correct. If systems that win on GeoGenBench do not also yield better transfer, better uncertainty, and more reusable physical insight than systems optimized under the old single-site paradigm, then the perspective advanced in this paper should be revised.

6. BROADER IMPLICATIONS

This shift is not merely of academic interest; it changes what geothermal computation is expected to deliver. The implications are scientific, engineering, and economic, but

all follow from the same claim: once prediction, intervention, and discovery are treated as a coupled process, geothermal systems should no longer be evaluated as static modeling exercises but as adaptive knowledge systems.

Scientific implications. Because persistent residual becomes a signal rather than noise, a new mode of scientific exploration opens in geophysics. Rather than hypothesizing governing equations *a priori* and fitting parameters, we can *discover* site-specific physics directly from data and compare these discoveries across geological settings. This data-to-equation approach can reveal previously unknown coupling mechanisms, identify geological conditions under which standard models systematically fail, and accelerate the development of next-generation geothermal physics. More broadly, it establishes a model paradigm for AI-driven scientific discovery in other subsurface domains, including carbon storage, groundwater management, and mineral extraction.

Engineering and system implications. Because decision support becomes feedback-coupled rather than scenario-based, generative world models and belief-state policies enable a new class of *geothermal digital twins*. These are not static simulations of one scenario, but uncertainty-aware environments in which operators can compare plausible futures, inspect trade-offs, and update decisions as observations arrive. The architecture of geothermal decision support shifts from a sequential workflow—run simulations, inspect outputs, manually adjust—to a closed-loop process in which modeling, intervention, and refinement continuously inform one another.

Economic and deployment implications. Because transferable physical structure reduces startup cost, the most immediate economic impact is faster deployment of new geothermal projects. If backbone models, reusable uncertainty representations, and partially transferable control principles carry knowledge from one site to the next, new deployments may require less bespoke calibration and shorter model-development cycles before operational analysis can begin. The importance of this shift is not that it removes site-specific work, but that it changes how much prior knowledge can be carried from one deployment to the next. This matters for scaling geothermal fast enough to meet ambitious deployment targets such as the DOE's Enhanced Geothermal Shot and longer-horizon U.S. geothermal expansion goals [52, 53].

7. CONCLUDING REMARKS

The next phase of progress in computational geothermal science will likely not come from faster surrogates alone. It will come from changing the computational framing of the problem. We have argued that geothermal is not best understood as deterministic simulation followed by downstream optimization, but as a closed-loop problem in which plausible subsurface futures must be inferred, interventions must be chosen under partial observability, and persistent mismatch must be turned into improved physical understanding. Under this view, simulation, control, and calibration are no longer separate modules. They become coupled parts of one computational loop. This does not make physics-based simulation obsolete, nor does it imply that geothermal can be reduced to generic machine learning. Rather, it suggests that the next generation of geothermal AI should be built around hybrid systems that generate plausible futures, adapt decisions as information evolves, and accumulate transferable scientific knowledge across sites. The residual-as-signal principle is the conceptual center of this agenda. It says that the most informative geothermal errors are not merely discrepancies to be minimized after the fact; they are structured traces of what the current representation fails to capture. If that principle is correct, then the real opportunity is not simply to solve familiar tasks more efficiently. It is to redesign the loop by which geothermal systems are modeled, operated, and scientifically understood. This claim is not specific to geothermal energy. Carbon storage, groundwater management, and nuclear waste disposal share the same defining features: partial observability, multi-physics dynamics, sparse data, and the entanglement of universal laws with site-specific geology. If this adaptive and integrated reframing succeeds in geothermal, it can provide a template for AI-driven scientific discovery across subsurface systems. Recent advances in generative modeling, scientific machine learning, and sequential decision making make this shift newly plausible. The most important step is revision of the abstraction itself. If that revision succeeds, geothermal can become a model domain for AI-driven scientific discovery in partially observed, intervention-sensitive physical systems.

8. ACKNOWLEDGEMENTS

Kunpeng Liu is supported by NSF 2550105, NSF 2550106, and NSF 2242812. Nori Nakata and Guodong Chen are supported by the U.S.Department of Energy under Award Number DE-AC02-5CH11231.

9. REFERENCES

[1] H. Bai, G. Chen, W. Ying, X. Wang, N. Gong, S. Dong, G. Pedrielli, H. Wang, H. Chen, and Y. Fu. Brownian bridge augmented surrogate simulation and injection planning for geological co2 storage. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 14459–14466, 2026.

[2] H. Bai, H. Wang, N. Gong, X. Wang, W. Ying, H. Chen, and Y. Fu. Supply chain optimization via generative simulation and iterative decision policies. In *2025 Winter Simulation Conference (WSC)*, pages 558–569. IEEE, 2025.

[3] J.-H. Bastek, W. Sun, and D. Kochmann. Physics-informed diffusion models. In *International Conference on Learning Representations*, volume 2025, pages 3360–3385, 2025.

[4] Z. Bi and N. Nakata. Learning injection–seismicity coupling for probabilistic multi-horizon forecasting in geothermal systems. 2026.

[5] Z. Bi, N. Nakata, R. Nakata, P. Ren, X. Wu, and M. W. Mahoney. Advancing data-driven broadband seismic wavefield simulation with multiconditional diffusion model. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–9, 2025.

[6] L. Biggio, T. Bendinelli, A. Neitz, A. Lucchi, and G. Parascandolo. Neural symbolic regression that scales. In *International Conference on Machine Learning (ICML)*, 2021.

[7] E. Bjarkason, A. Yeh, J. O'Sullivan, A. Croucher, and M. O'Sullivan. Non-uniqueness of geothermal natural-state simulations. 11 2019.

[8] D. Blackwell, M. Richards, Z. Frone, J. Batir, A. Ruzo, R. Dingwall, and M. Williams. Temperature-at-depth maps for the conterminous us and geothermal resource estimates. Southern Methodist University Geothermal Laboratory, Dallas, TX (United States), 10 2011.

[9] H. Cao, H. Bai, and Y. Fu. Structured memory and role-aware decision making for supply chain transportation. In *2025 IEEE International Conference on Big Data (BigData)*, pages 1837–1846. IEEE, 2025.

[10] H. Cao, J. Zhang, K. Liu, D. Wang, F. Xia, H. Chen, X. Hu, and Y. Fu. Sim2act: Robust simulation-to-decision learning via adversarial calibration and group-relative perturbation. *arXiv preprint arXiv:2603.09053*, 2026.

[11] D. K. Chandra, P. Wang, J. Leopold, and Y. Fu. Collective representation learning on spatiotemporal heterogeneous information networks. In *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 319–328, 2019.

[12] G. Chen, J. J. Jiao, Q. Liu, Z. Wang, and Y. Jin. Machine-learning-accelerated multi-objective design of fractured geothermal systems. *Nexus*, 1(4), 2024.

[13] Y. Chen, D. Voskov, and A. Daniilidis. Open-source simulation study for direct use geothermal systems. 02 2024.

[14] M. Cranmer. Interpretable machine learning for science with pysr and symbolicregression.jl. 2023.

[15] W. Fan, K. Liu, H. Liu, H. Zhu, H. Xiong, and Y. Fu. Feature and instance joint selection: A reinforcement learning perspective. *arXiv preprint arXiv:2205.07867*, 2022.

[16] S. Finsterle. Practical notes on local data-worth analysis. *Water Resources Research*, 51(12):9904–9924, 2015.

[17] Y. Fu, D. Wang, W. Ying, X. Wang, X. Zhang, H. Liu, and J. Pei. Autonomous data agents: A new opportunity for smart data. *arXiv preprint arXiv:2509.18710*, 2025.

[18] R. Gambini, D. W. Waters, F. Sansone, and V. Memmo. Risk and uncertainty in geothermal projects: Characteristics, challenges and application of the novel reverse enthalpy methodology. *Energies*, 18(15), 2025.

[19] W. Gao, J. Gao, Q. Han, H. Pan, and K. Liu. Graph random walk with feature-label space alignment: A multi-label feature selection method. *arXiv preprint arXiv:2505.23228*, 2025.

[20] W. Gao, Z. Man, Z. He, Y. Tang, J. Gao, and K. Liu. Two-stage feature generation with transformer and reinforcement learning. *arXiv preprint arXiv:2505.21978*, 2025.

[21] G. Grubac, W. El-Rabaa, A. Bonneville, I. Ben-Fayed, R. A. Gonzalez, G. Gullickson, and O. Perez. Implementation of the world's first greater than 300 c propped egs reservoir.

[22] R. Harsuko, Z. Bi, and N. Nakata. Gaia: Geothermal analytics and intelligent agent. *arXiv preprint arXiv:2511.03852*, 2025.

[23] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc.

[24] R. Horne, A. Genter, M. McClure, W. Ellsworth, J. Norbeck, and E. Schill. Enhanced geothermal systems for clean firm energy generation. *Nature reviews clean technology*, 1(2):148–160, 2025.

[25] H. Hoteit, X. He, B. Yan, and V. Vahrenkamp. Uncertainty quantification and optimization method applied to time-continuous geothermal energy extraction. *Geothermics*, 110:102675, 2023.

[26] J. Jello and T. Baser. Utilization of existing hydrocarbon wells for geothermal system development: A review. *Applied Energy*, 348:121456, 2023.

[27] P.-A. Kamienny, S. d'Ascoli, G. Lample, and F. Charton. End-to-end symbolic regression with transformers. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc.

[28] H. Kurniawati. Partially observable markov decision processes (pomdps) and robotics, 2021.

[29] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhatt, A. Stuart, and A. Anandkumar. Fourier neural operator for parametric partial differential equations. *International Conference on Learning Representations (ICLR)*, 2021.

[30] Z. Liang, J. Yu, R. Thibaut, F. Zheng, C. Hoiland, and A. Pollack. Surrogate modeling for geothermal systems: Accelerating optimization, history matching, and uncertainty quantification.

[31] C. Liu, X. Xu, and D. Hu. Multiobjective reinforcement learning: A comprehensive overview. volume 45, pages 385–398, 2015.

[32] R. Liu, S. J. Quan, Z.-R. Peng, Z. Yao, H. Wang, Z. Chen, K. Liu, Y. Fu, and D. Wang. City editing: Hierarchical agentic execution for dependency-aware urban geospatial modification, 2026.

[33] R. Liu, R. Xie, Z. Yao, Y. Fu, and D. Wang. Continuous optimization for feature selection with permutation-invariant embedding and policy-guided search. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1857–1866, 2025.

[34] R. Liu, T. Zhe, Y. Fu, F. Xia, T. Senator, and D. Wang. Permutation-invariant representation learning for robust and privacy-preserving feature selection, 2026.

[35] R. Liu, T. Zhe, Z.-R. Peng, N. Catbas, X. Ye, D. Wang, and Y. Fu. Urban planning in the age of agentic ai: Emerging paradigms and prospects. *ACM SIGKDD Explorations Newsletter*, 27(2):35–42, 2026.

[36] R. Liu, T. Zhe, D. Wang, Z. Yao, K. Liu, Y. Fu, H. Liu, and J. Pei. Agentos: From application silos to a natural language-driven data ecosystem, 2026.

[37] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, 2021.

[38] E. L. Majer and J. E. Peterson. The impact of injection on seismicity at the geysers, california geothermal field. *International Journal of Rock Mechanics and Mining Sciences*, 44(8):1079–1090, 2007.

[39] N. Nakata, H. Chang, S. Wu, Z. Bi, L. Chen, F. Soom, H. Gao, A. Titov, and S. Dadi. Fracture characterization and stress communication revealed by induced seismicity at the cape modern geothermal field. *Utah, GRC Transactions*, 49(2025):1191–1202, 2025.

[40] M. I. of Technology. *The Future of Geothermal Energy: Impact of Enhanced Geothermal Systems (EGS) on the United States in the 21st Century : an Assessment*. The Future of Geothermal Energy: Impact of Enhanced Geothermal Systems (EGS) on the United States in the 21st Century : an Assessment by an MIT-led Interdisciplinary Panel. Massachusetts Institute of Technology, 2006.

[41] H. Oh, K. Beckers, and K. McCabe. Geospatial characterization of low-temperature heating and cooling demand in residential, commercial, manufacturing, agricultural, and data center sectors for potential geothermal applications in the united states. *Renewable and Sustainable Energy Reviews*, 206:114875, 2024.

[42] P. Olasolo, M. Juárez, M. Morales, S. D´Amico, and I. Liarte. Enhanced geothermal systems (egs): A review. *Renewable and Sustainable Energy Reviews*, 56:133–144, 2016.

[43] J. Pathak, S. Subramanian, P. Harrington, S. Raja, A. Chattopadhyay, M. Mardani, T. Kurth, D. Hall, Z. Li, K. Azizzadenesheli, P. Hassanzadeh, K. Kashinath, and A. Anandkumar. Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators, 2022.

[44] K. Pruess, C. Oldenburg, and G. Moridis. TOUGH2: A general-purpose numerical simulator for multiphase fluid and heat flow. *Lawrence Berkeley National Laboratory Report*, 1999. LBNL-43134.

[45] H. Rahimian and S. Mehrotra. Distributionally robust optimization: A review. *arXiv preprint arXiv:1908.05659*, 2019.

[46] P. Ren, R. Nakata, M. Lacour, I. Naiman, N. Nakata, J. Song, Z. Bi, O. A. Malik, D. Morozov, O. Azencot, et al. Learning earthquake ground motions via conditional generative modeling. *Nature Communications*, 2026.

[47] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022.

[48] J. Rutqvist and O. Stephansson. The role of hydromechanical coupling in fractured rock engineering. *Hydrogeology Journal*, 11(1):7–40, 2003.

[49] G. Shafer and V. Vovk. A tutorial on conformal prediction. *Journal of machine learning research*, 9(3), 2008.

[50] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

[51] J. Taron, D. Elsworth, and K.-B. Min. Numerical simulation of thermal-hydrologic-mechanical-chemical processes in deformable, fractured porous media. *International Journal of Rock Mechanics and Mining Sciences*, 46(5):842–854, 2009.

[52] U.S. Department of Energy. Doe launches new energy earthshot to slash the cost of geothermal power, September 2022. Enhanced Geothermal Shot aims to reduce EGS cost by 90% to $45/MWh by 2035.

[53] U.S. Department of Energy. Geovision: Harnessing the heat beneath our feet. `https://www.energy.gov/eere/geothermal/geovision`, 2024. U.S. geothermal market analysis and projections.

[54] Utah FORGE. Utah FORGE: Frontier observatory for research in geothermal energy, n.d. Accessed March 22, 2026.

[55] A. Vignesh Malarkkan, X. Wang, K. Liu, D. Zhang, and Y. Fu. Causally-guided diffusion for stable feature selection. *arXiv e-prints*, pages arXiv–2603, 2026.

[56] X. Wang, H. Cao, K. Liu, and Y. Fu. Dataforge: Agentic platform for autonomous data engineering. *arXiv preprint arXiv:2511.06185*, 2025.

[57] X. Wang, D. Wang, W. Ying, H. Bai, N. Gong, S. Dong, K. Liu, and Y. Fu. Efficient post-training refinement of latent reasoning in large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 33692–33700, 2026.

[58] X. Wang, D. Wang, W. Ying, R. Xie, H. Chen, and Y. Fu. Knockoff-guided feature selection via a single pre-trained reinforced agent. *IEEE Transactions on Big Data*, 12(2):625–642, 2026.

[59] Z. Wang, J. Zhang, X. Zhang, K. Liu, P. Wang, and Y. Zhou. Diversity-oriented data augmentation with large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 22265–22283, 2025.

[60] G. Wen, Z. Li, K. Azizzadenesheli, A. Anandkumar, and S. M. Benson. U-FNO: An enhanced Fourier neural operator-based deep-learning model for multiphase flow. *Advances in Water Resources*, 163:104180, 2022.

[61] M. C. White and N. Nakata. Induced seismicity and geothermal energy production in the salton sea geothermal field, california. *Scientific Reports*, 15(1):1638, 2025.

[62] M. Xiao, D. Wang, M. Wu, K. Liu, H. Xiong, Y. Zhou, and Y. Fu. Traceable group-wise self-optimizing feature transformation learning: A dual optimization perspective. *ACM Transactions on Knowledge Discovery from Data*, 18(4):1–22, 2024.

[63] J. Xu, Y. Tian, P. Ma, D. Rus, S. Sueda, and W. Matusik. Prediction-guided multi-objective reinforcement learning for continuous robot control. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

[64] W. Ying, H. Bai, N. Gong, X. Wang, S. Dong, H. Chen, and Y. Fu. Bridging the domain gap in equation distillation with reinforcement feedback. *arXiv preprint arXiv:2505.15572*, 2025.

[65] W. Ying, H. Bai, K. Liu, and Y. Fu. Topology-aware reinforcement feature space reconstruction for graph data. *ACM Transactions on Knowledge Discovery from Data*, 20(1):1–22, 2025.

[66] W. Ying, C. Wei, N. Gong, X. Wang, H. Bai, A. V. Malarkkan, S. Dong, D. Wang, D. Zhang, and Y. Fu. A survey on data-centric ai: Tabular learning from reinforcement learning and generative ai perspective, 2025.

[67] W. Ying, J. Zhang, H. Bai, N. Gong, X. Wang, K. Liu, C. K. Reddy, and Y. Fu. Data-efficient symbolic regression via foundation model distillation. *arXiv preprint arXiv:2508.19487*, 2025.

[68] J. Zhang, F. Mo, T. C. Weerasooriya, X. Ye, D. Wang, Y. Fu, and K. Liu. Blind spot navigation in large language model reasoning with thought space explorer. In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 3691–3707, 2026.

[69] J. Zhang, X. Wang, Y. Jin, C. Chen, X. Zhang, and K. Liu. Prototypical reward network for data-efficient rlhf. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13871–13884, 2024.

[70] J. Zhang, X. Wang, F. Mo, Y. Zhou, W. Gao, and K. Liu. Entropy-based exploration conduction for multi-step reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 3895–3906, 2025.

[71] J. Zhang, X. Wang, W. Ren, L. Jiang, D. Wang, and K. Liu. Ratt: A thought structure for coherent and correct llm reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 26733–26741, 2025.

[72] J. Zhang, H. Xie, X. Zhang, and K. Liu. Enhancing risk assessment in transformers with loss-at-risk functions. In *2024 IEEE International Conference on Knowledge Graph (ICKG)*, pages 477–484. IEEE, 2024.

[73] X. Zhang, J. Zhang, F. Mo, D. K. Chandra, Y.-Z. Chen, F. Xie, and K. Liu. Retrieval-augmented feature generation for domain-specific classification. In *2025 IEEE International Conference on Data Mining (ICDM)*, pages 943–952. IEEE, 2025.

[74] X. Zhang, J. Zhang, F. Mo, D. Wang, Y. Fu, and K. Liu. Leka: Llm-enhanced knowledge augmentation. *arXiv preprint arXiv:2501.17802*, 2025.

[75] X. Zhang, J. Zhang, B. Rekabdar, Y. Zhou, P. Wang, and K. Liu. Dynamic and adaptive feature generation with llm. *arXiv preprint arXiv:2406.03505*, 2024.

[76] T. Zhe, R. Liu, F. Memar, X. Luo, W. Fan, X. Ye, Z. Peng, and D. Wang. Constraint-aware route recommendation from natural language via hierarchical llm agents, 2025.