Delve: A Dataset-Driven Scholarly Search and Analysis System

Uchenna Akujuobi, Xiangliang Zhang
Computer, Electrical and Mathematical Sciences and Engineering Division
King Abdullah University of Science and Technology (KAUST)
Saudi Arabia
{uchenna.akujuobi,xiangliang.zhang}@kaust.edu.sa

ABSTRACT

Research and experimentation in various scientific fields are based on the observation, analysis and benchmarking on datasets. The advancement of research and development has thus, strengthened the importance of dataset access. However, without enough knowledge of relevant datasets, researchers usually have to go through a process which we term "manual dataset retrieval". With the accelerated rate of scholarly publications, manually finding the relevant dataset for a given research area based on its usage or popularity is increasingly becoming more and more difficult and tedious. In this paper, we present Delve, a web-based dataset retrieval and document analysis system. Unlike traditional academic search engines and dataset repositories, Delve is dataset driven and provides a medium for dataset retrieval based on the suitability or usage in a given field. It also visualizes dataset and document citation relationship, and enables users to analyze a scientific document by uploading its full PDF. In this paper, we first discuss the reasons why the scientific community needs a system like Delve. We then proceed to introduce its internal design and explain how Delve works and how it is beneficial to researchers of all levels.

1. INTRODUCTION

The word "Data" according to the Webster's English dictionary [35], is defined as "a collection of facts, observations, or other information related to a particular question or problem". Based on the above definition, data (physical or digital) can be attributed to being a "cornerstone" of various scientific researches which have led to the advancement of science and technology. In various scientific fields, the research process involves exploration, analysis and evaluation based on a set of **data**. For instance, various computer science fields (machine learning, data mining, database, computer vision, pattern recognition, etc.) usually evaluate the effectiveness of a proposed approach based on experiments conducted on a set of benchmark datasets. In several other scientific fields like the environmental and biological sciences, the credibility of proposed models designed from data and available knowledge in consort with end-users and simulations is usually critically analyzed and reviewed based on the model's performance on a particular range of data spectrum [22; 16; 5; 2].

The amelioration of science and technology has made it possible not just to approach problems that could have never been solvable in the past but to also improve upon the performance of previous methods. Data is essential to both cases. However, scholarly search based on dataset usage even when familiar with the research field might require a significant amount of time and effort due to the unprecedented rate of scholarly publications [23]. For example, consider the query: "Find all papers using the MOA datasets and working on relational learning". Typically, a two-step manual method for answering this query is to 1) query the academic search engines for papers on relational learning, and 2) spend a lot of time reading through the searched papers to filter out the papers using the MOA datasets. This process can be quite tedious and becomes more complex as a paper "A" might specify it used the same dataset as in paper "B", in which case a researcher also needs to search and go through paper "B" to determine what dataset was used. Another relevant query can be: "Find popular datasets used in relational learning". It will also require a vast amount of time to be dedicated to reading many articles.

It is vital to have a data-driven search engine to exploit the rich semantics of dataset information available in academic documents, which current scholarly search engines fail to provide. With the availability of different academic web search engines and databases (e.g., Google scholar[1], Microsoft Academic[2], Semantic scholar[3]), information on papers or authors in different fields, topics, can be easily accessed. Also, dataset portals and repositories like Opendata[4] and UCI repository [21] provide a medium where users can search for datasets. However, having these two systems independently, even though each individually performs its respective functions, offers only a meager benefaction in finding relevant papers working on a given dataset or finding relevant datasets for a given problem. To the best of our knowledge, only one academic search engine[3] currently integrates the use of dataset in academic document search. It uses dataset as a filter medium to their search results, rather than allowing datasets as a search query, i.e., not answering a simple query like "Find all the papers using MOA dataset".

In this paper, we present Delve, an online dataset-driven system that provides a medium for dataset or document search, visual analysis of the citation relations among documents and datasets, and online document analysis. More

[1]https://scholar.google.com
[2]https://academic.microsoft.com
[3]https://www.semanticscholar.org
[4]https://data.opendatasoft.com

specifically, Delve offers users a simple and easy-to-use interface for

- Finding a set of benchmark datasets for a research topic/field interest;
- Finding a set of research papers that used the same datasets;
- Visually analyzing the citation relations of academic documents and datasets;
- Instantly online analyzing an academic document and showing its citation relations w.r.t. other documents and datasets, when a PDF of the document is provided.

With these above-mentioned features, Delve is useful for different purposes, e.g., finding relevant papers for literature review, finding appropriate datasets for a specific research interest, understanding document and dataset citation relationships.

The rest of the paper is organized as follows. In section 2, we discuss the importance of data availability and access for scientific research. Section 3 presents a brief overview of the previous works done in the area of document and dataset search. We explicitly introduce the Delve system in section 4 and show how the Delve system works in section 5 by presenting some use case scenarios. We then provide some possible extension of the system in section 6 and conclude in section 7.

2. DATA ACCESS IN SCIENCE

Ancient civilizations like the Egyptians, Babylonians, Indians, and Chinese; practiced what many could refer to today as applied science and mathematics [15]. Using the knowledge obtained from recording and studying the stars and heavenly bodies, they were able to predict seasons and develop principles of direction that they then applied to agriculture and navigation. The availability of the recorded data about the stars and heavenly bodies played a significant role in the study of seasons and navigation. Data have always been a driving force in the evolution of science and technology. With the current progression of science, the crucial need for data becomes more and more pronounced. In this section, we will discuss the importance of dataset access and analysis in scientific research.

2.1 Empirical Evaluations

Empirical evaluations are one of the fundamental procedures in scientific research, for validating and analyzing the performance of different methods. Usually, the evaluations provide a comparison showing which method is superior in a given problem setting [12]. As commonly noted by several authors [12; 8; 28; 7], evaluations can sometimes be seriously misleading, and need to be made in a fair and objective way. Also, as noted by Keogh [19], most empirical evaluations are data biased because the choice of dataset has a substantial effect on the results reported in many scholarly papers. For instance, let us consider a researcher with a prior research interest in Natural Language Processing (NLP) who might be interested in developing a new method or extending a previous method in a new research area of interest (e.g., *image annotation*). To show the performance of her method, the researcher would need to compare the performance of her method with that of some prior methods on the different datasets used by the prior works. Without the availability and prior knowledge of such datasets, a fair and objective comparison cannot be made. Easy access to information about datasets and how they have been used will reduce the data biases in empirical evaluations and curtail the dilemma of choosing the wrong datasets for this step of scientific research. This information would improve the quality and validity of empirical evaluations and increase the efficiency of researchers.

2.2 Reproducibility and Data Analysis

A scientific work needs to be repeatable given the same procedure, parameters, and data. Reproducibility makes a research work easier to understand by other researchers both to verify the reported results or to extend the work. However, the lack of reproducibility has continued to be a significant problem in science, and several authors [14; 19; 27] have warned against it. The availability of dataset used in a scientific work is quite crucial for the reproducibility of the work, as methods perform differently with different datasets [12]. Providing information and easy access to datasets used in various scientific research works would enhance the reproducibility of these works, and thus enable an objective analysis and validation of research works, for promoting the quality of scientific research.

Another advantage of providing access to dataset information is that the value of data can be better explored by more researchers. This data exploration could lead to further insights and observations, bringing about more knowledge discoveries from the dataset by using it for different purposes, which might not be the initial intention of gathering the dataset. A valid example of this, as mentioned by Vanschoren et al. [33], is the Sloan Digital Sky Survey (SDSS) data. The SDSS project commissioned to take spectra and images of about 35% of the night sky has so far created the most comprehensive astrophysical catalog in the world [36]. This collected data, which was initially confined only to the members of the project, was opened up to the public [30] and has since been used in different research studies. Due to the availability of the data, scientists were able to ask different questions from the dataset [29; 24; 25; 13; 4], leading to a vast number of discoveries. An example of a significant discovery from the SDSS data is the discovery of the emission light galaxy known as "Green peas" via the Galaxy Zoo project. The Galaxy Zoo project employs the help of astronomy enthusiasts to classify millions of galaxies in data obtained from different sources including the SDSS. Volunteers studying the SDSS data provided by the Galaxy Zoo project discovered the emission light galaxy by noting their peculiarity which was then unresolved in Sloan Digital Sky Survey imaging [4].

3. PREVIOUS WORK

Over the years, the importance of access to scholarly documents and datasets has been increasingly recognized by researchers. There have been works directed towards making information to scholarly documents and datasets easily accessible. However, these efforts have mostly been disjoint.

3.1 Scholarly Search Engines

Citeseer[5], an open-source scholarly search engine, was introduced in 1997 by Giles et al. [10] as an automatic citation indexing system. Citeseer later became CiteseerX in 2007, which is a scholarly search engine, digital library, and repository for scientific and academic papers with a focus on scholarly papers in computer science [1]. Citeseer is considered to be the first academic search engine and only indexes publicly available documents. In 2000, Scirus[6] was launched as a joint work between FAST, a Norwegian search engine company, and the Elsevier Science publishing group to address the problem of access to scholarly documents from both authoritative sources like publishers and non-authoritative sources like university websites. Scirus has since been retired in 2014 and replaced with Scopus[7].

Two of the more recent scholarly search engines are Google Scholar[1] and Microsoft Academic[2]. Google scholar was initially launched in 2004 as a way to improve the efficiency of researchers by providing access to scholarly literature information and knowledge [20]. Over the years more features have been added to the search engine including saving search results and organizing author citations. Microsoft Academic was initially introduced as Windows Live Academic Search in 2006 to compete with Google's scholarly search engine, and then was retired after two years. In 2016, Microsoft Academic, a relaunch of Microsoft Academic Search, was introduced as a free public scholarly search engine, which essentially has the same function as other scholarly search engines.

Some systems extend the idea of academic search engines by applying machine learning techniques on the academic documents to retrieve other information from the scholarly materials. AMiner[8] (former Arnetminer) was created in 2006 to search and analyze researcher networks [31]. In 2015, Semantic Scholar[3] was created to provide a smart search service for journals by applying some machine learning techniques and a layer of semantic analysis. Semantic Scholar incorporates the use of datasets as a filter parameter in generating their results. Currently, there is a considerable amount of scholarly search engines available on the web, each with its features. However, none of the currently available academic search engines is dataset driven.

3.2 Dataset Repositories and Portals

The creation of standard collections of datasets has made the reproduction and empirical evaluations of scientific work easier and fair [12]. There are a lot of dataset repositories and data portals currently available. Some of these like the UCI repository [21], KDD achive[9], Mldata[10], OpenDataSoft[4], Data.Gov[11], SDSS[12] and LDC[13] are openly accessible to the public. There are also the commercial dataset collections including Datamarket[14], Xignite[15], and IEEEDataPort[16]. With the increasing advocacy of open data, more and more closed datasets are being made public. Open data has been shown to benefit both the academic community and the data owner [33; 14; 12; 17].

Vast number of dataset repositories and data portals mean more available datasets to use, but also mean more difficulties for researchers to find appropriate datasets and relevant references. It is often that researchers end up using datasets they have heard or read about, even though there might be better datasets available and more suitable for their research problem. Having a platform possessing information on datasets from different dataset repository and data portals, ranking them by relevancy to a search keyword or phrase, and providing the relevant datasets to researchers will not only provide researchers with better dataset choices but also provide exposure to various good dataset repositories and data portals.

4. DELVE

Delve[17] is a dataset-driven system with a focus on dataset retrieval and document analysis [3]. The advanced features Delve has over other scholarly search engines are

- Delve can be used to **retrieve dataset-driven results**. For example, in Delve homepage (shown in Figure 8a), a user can give a query word, which can be a dataset name or a research topic, e.g., *image/video annotaiton*. Matched datasets will be returned and ranked by their relevance (shown in the middle part of Figure 6), as well as relevant research documents (given in the right part of Figure 6). This feature is very useful for finding relevant datasets and surveying relevant research documents.
- Delve can be used to **understand the relationship** between papers and dataset. This relationship can be paper-to-paper, paper-to-dataset, or dataset-to-dataset. For example, the graph in Figure 7b and Figure 8b show the citation relationship among papers and datasets. The visualization of the relationship among papers and datasets can help on easily getting more insights about a paper or dataset.
- Delve can **analyze PDFs** of academic documents. This feature can be used in analyzing a document even before submitting it to a journal or conference to evaluate its relationship to other published papers. With this, a researcher will be able to detect a paper that might be of advisable to cite or read through. Figure 8 demonstrates one example of this feature.

Detailed works behind each feature will be discussed in next subsections. At the time of writing this paper, we have to admit some disadvantages of Delve:

- The size of the Delve database (currently including 2 million scholarly documents) is limited when compared with the database of other popularly used academic search engines. This limitation would be less

[5]http://citeseerx.ist.psu.edu
[6]http://www.scirus.com
[7]https://www.scopus.com
[8]https://aminer.org
[9]https://kdd.ics.uci.edu
[10]http://mldata.org
[11]https://www.data.gov
[12]http://www.sdss.org
[13]http://www.ldc.upenn.edu
[14]http://www.qlik.com/us/products/qlik-data-market
[15]http://www.xignite.com/
[16]https://ieee-dataport.org/
[17]https://delve.kaust.edu.sa

of an issue as the Delve database will be continuously expanded with more collections.

- There are false citation relationships displayed on Delve. This issue is due to the limited document collection size and the simple algorithm Delve currently uses for citation relationship prediction. We expect this to be curtailed with improvements to the applied algorithms and the addition of more document collections.

4.1 The Delve Database

The Delve database was initialized by collecting papers published in 17 different conferences and journals between 2001 to 2016, including AAAI, IJCAL, TKDD, NIPS, CIKM, VLDB, ICML, ICDM, PKDD, WSDM, SDM, ICDE, KDD, DMKD, KAIS, WWW, and TKDE. Using the Microsoft graph dataset[18], the Delve database was extended to include references and the references of their references (up to 2 hops away) of the papers in the initially selected conferences and journals. This extension thus enlarged our database. At the time of writing this paper, the Delve database includes more than 2 million scholarly documents from more than 1000 different sources including conferences, journals, and books.

Documents and datasets are treated as nodes in Delve. A large citation graph is then built by linking papers and papers, papers and datasets, datasets and datasets if there exist citation/usage relations among them. For supporting dataset-driven search, Delve explores not only the **node content** (text of scholarly documents or datasets), but also the **edge labels** (positive labels indicating the dataset relevance, e.g., paper A used dataset D, or paper A citing paper B because of the common datasets they used). The initial labeling work was conducted by crowd-sourcing on papers and datasets cited by these papers published in ICDE, KDD, ICDM, SDM, and TKDE from 2001 to 2014. These labels (accounting for 5% of the whole graph edges) have been manually verified to be correct by three qualified participants. Due to the high cost of crowd-sourcing, it is infeasible to label the remaining 95% of edges manually. Therefore, one of the principal challenges that arise in Delve is to develop an efficient and effective method to assign labels to a large number of unlabeled edges. To tackle this issue, we developed a semi-supervised learning method using ideas adopted from the label propagation algorithm [9] for edge label inference, which will be discussed later in section 4.3.

4.2 Document Parsing

For preparing node content, we acquired publicly available PDF documents of papers in the Delve dataset when accessible. For nodes when PDFs are not accessible, we acquired their other content information such as title, authors, abstract, publication venue, publication year, URLs, etc. The documents were collected through web crawling from different sources. The web crawler was designed to go through a list of scholarly document URLs to locate and download PDF files that are openly available. This URL list of papers was obtained from the Microsoft graph dataset and by crawling the web. We were able to collect about 680,000 PDFs, according for 32% of the nodes in the whole graph.

[18]https://academicgraph.blob.core.windows.net/graph-2015-11-06/index.htm

The downloaded PDFs are converted into text using the Linux pdf2text tool. Then using methods proposed in [6; 32], we sectionize the text. We extract the following sections from each document:

Header: This is composed of the title of the paper, the author(s) information, and the keywords when available;

Abstract: We extract the paper abstract when available;

Paper body text: This is composed of the document text excluding the header information, abstract and references;

Citation: We extract the document references made in the paper, which is then parsed further to separate the different parts of the citation: author, title, year of publication, and page numbers;

Citation context: These are the sentences encompassing a citation reference in the body of the document;

Cited links: These are the URLs cited in the paper. These URLs could be links relevant to the research work, link to datasets used, or link to the implementation codes.

Due to the variety of the documents we currently have in our database, we still experience some of the parsing issues due to variations in formatting as noted in [10]. We expect in time for this problem to be reduced with the improvement to the parsing algorithm.

The citation information of the papers is extracted from the paper text, Microsoft dataset, and the web. We proceed to identify and merge the different citations to the same article, and then build the citation network. The apparent difficulty in dealing with citations made in different conferences and journals is the variations in the formatting of documents and their citation methods, such as the MLA, APA, Chicago, Harvard, and other formats. There exist also papers that do not follow any particular guideline citation format, even include typos in citation.

In order to split each cited paper in *References* into sections such as *authors*, *title*, *publication year* and so on, we developed a rule-based method and combined it with the method proposed in [6]. The heuristics in the constructed rules have considered the variation of reference styles in different documents. For instance, the author section normally appear first, and often separated by comma from each other. The publication year, a double quote or a full-stop usually separates the authors and the title sections. However, these observations do not present a generalization over all the citation syntaxes which we incorporated in our splitting method. After the splitting, a reference paper appearing in different styles are merged as a single one. Then, we proceed to create the citation network of the system by building the links between papers and datasets based on their citation relationship. Next, we discuss the citation network building and labeling.

4.3 Delve System Design

The Delve system is made up of two main modes of operation. A high-level view of the system architecture is shown in Figure 1. The offline processing module includes the remote structure, framework, and design of the system to ensure

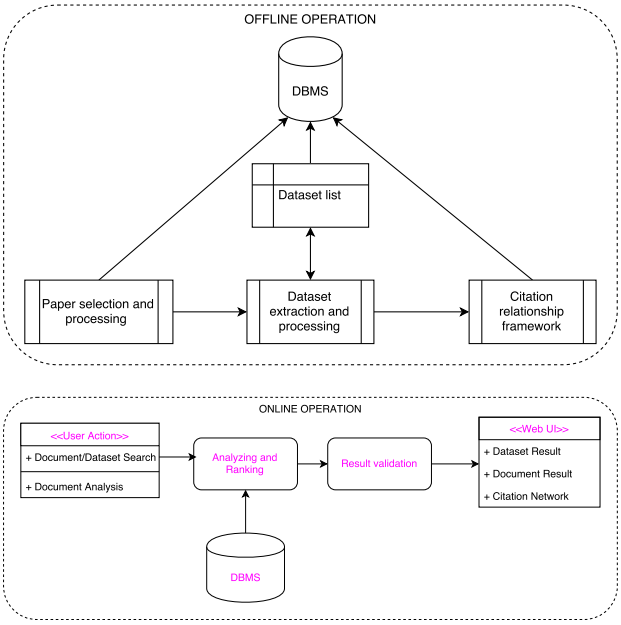


Figure 1: Delve Offline and Online Process

its functionality. New improvements and updates are regularly made to the system. In online processing, Delve web interface accepts user query (search phrase or file upload). Delve analyzes and executes the query, launching different processes that perform the execution, ranking, and result analysis.

We express a paper or dataset source in our system database to an *entity*. Each *entity* is made up of a set of attributes (title, authors, papers cited, papers citing it, its citation information, etc). From this information, a citation network $G = \{V, E\}$ is built through linking two entities if one cites the other. An edge between v_i and v_j are labeled **positive** (dataset related) if v_i cites v_j because v_i uses the dataset in v_j, or **negative** (not dataset related) otherwise. As discussed previously, the data-driven search will explore nodes that are linked by positive edges. A significant challenge in Delve is the edge label assignment in a large citation network with only 5% known labels, which are crowd-sourced.

4.3.1 Edge Label Assignment

Based on the logical assumption that a highly cited dataset entity will most likely gain more positive citations than negative citations in the future, we see that for our problem, entity citations labels are interrelated. We adopted two methods (label propagation [37] and PageRank [26]) and modified them to infer labels for the edges with unknown labels. They are selected due to their advantages in the amount of priori needed, better run time, and fitness our problem, considering that we are working with a large network with missing information.

Before using the inferred edge labels in query answering, we conducted extensive experiments to evaluate the developed methods on a total of 101,503 labeled edges. The results are reported in Table 1. The studied network is overwhelmingly unbalanced with most of the nodes having very low in-degree. Therefore, to ease the performance assessment,

Table 1: Performance results of edge label inference using modified PageRank and label propagation

	Pagerank	Label Propagation
AUC	0.8231	0.8979
Precision	0.9933	1.0
Recall	0.6391	0.6260

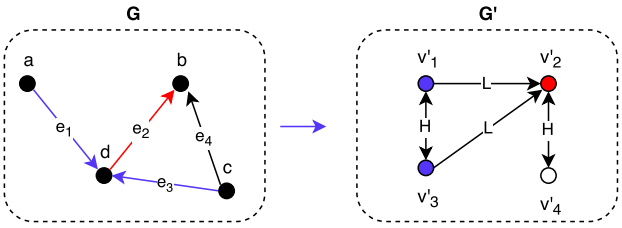


Figure 2: Graph G is reconstructed into G', where $e_i = v'_i$, $L = 0.5 + Sim_{ij}$, $H = 1 + Sim_{ij}$. In G, different types of link relationship are illustrated by colors, blue links are positive, red links are negative, while a black edge (e_4) shows a link with a missing label. In G', these edges are represented by nodes in corresponding colors, while the white node v'_4 signifies the black edges e_4 with a missing label.

we randomly sample 10% of incoming edges to nodes with high in-degree as test datasets. The results are obtained from running the evaluation five times with random samples and then taking the average. We measure the performance of these methods based on the precision, recall, and AUC value. The performance of both methods are comparable. However, label propagation is a little better and more stable than RageRank w.r.t. parameter settings. Thus, the adoption of the label propagation method as the main labeling algorithm for the Delve system. The details of these two approaches are given next.

4.3.1.1 Label Propagation.

Label propagation is a popular graph-based semi-supervised learning framework which is efficient in large graphs. The classic problem setup is defined as follows: given a graph $G = \{V, E\}$, a set of nodes V_l have known labels, while the remaining nodes V_{ul} have unknown labels. Label propagation infers the labels of V_{ul} by progagating the known labels from V_l to V_{ul}.

We aim to label the edges, and thus restructure our graph to $G' = \{V', E', W'\}$, where the set of nodes V' is the set of edges E in graph G and E' is the set of generated edges whose weight W' show the calculated similarities between two edges corresponding to nodes $(v'_i, v'_j), \forall v'_i, v'_j \in V'$. The edges E' are generated by linking each edge e_i in the original G (v'_i in G') to the top 20 similar edges e_j (v'_j in the original G') that have the same target node as e_i or where the target node of e_i is the source node of e_j.

A simple example of this reconstruction is shown in Figure 2. Since this is not an undirected graph, using the reconstructed graph G' as described above is not enough to ensure convergence. A general way to ensure convergence is to disregard the directions or make the graph matrix stochastic and irreducible [26]. The problem with these solutions w.r.t. our work is that they implicitly make a vague assumption that all papers and dataset are somewhat related.

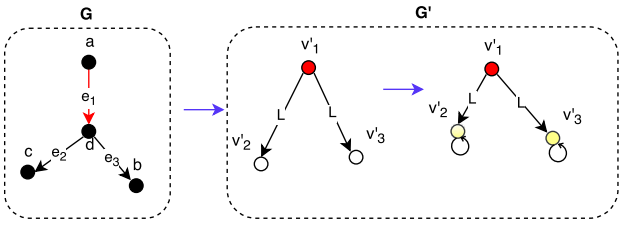


Figure 3: A simple case of a dead-end. A preprocessing step is applied on the reconstructed graph - assigning an "unknown" label to each node with a dead-end (in this case nodes v'_2 and v'_3; shown in yellow).

We, therefore, opt for a different solution and introduce a new label "unknown". Before running the propagation algorithm, we scan through the reconstructed graph, searching and assigning the label "unknown" and append a self-loop to the nodes $v'_d \in V'$ with a dead-end. Figure 3 presents a simple example of a dead-end preprocessing.

To define the similarity between v'_i and v'_j (two edges e_i and e_j in the original graph G), we extract the number of dataset keywords[19] from each citation context (i.e., the sentences which encompass the citations). We then defined a Gaussian similarity score between pairs of edges (e_i and e_j in G)

$$Sim_{ij} = \exp(-\frac{||d_i - d_j||^2}{2\sigma^2}) \quad (1)$$

where $d_i = \frac{n_d}{n_c}$, n_d is the number of dataset related words in the sentences which encompass the citation depicted as d_i, and n_c is the number of such sentences in the source papers.We then assign a weight:

$$W'_{ij} = \begin{cases} 1 + Sim_{ij} & \text{if } v'_i \text{ and } v'_j \text{ have the same target } v_t \in G, \\ 0.5 + Sim_{ij} & \text{otherwise.} \end{cases} \quad (2)$$

With the constructed graph $G' = \{V', E', W'\}$ where a small portion of V' have verified labels, label propagation algorithm is run to propagate the given labels to unlabeled V'. After the labeling step, the graph G' is reconstructed back to the original graph G.

4.3.1.2 PageRank.

PageRank algorithm [26] determines the importance of a web page based on the importance of other web pages with which it has in-links. A web page that has more in-links will have higher importance (measured as PageRank score). In-links can be considered as weighted votings, where the weight of a link depends on the importance of the source page and also the number of out-links the source page has. In our built network, there is a general observation: if a paper node or dataset node is highly cited with positive edges, the likelihood of a new unknown citation to this node to be also positive is high. This observation conforms to the mechanism of PageRank: web page with many in-links are good, and in-links from a "good" web page are better than in-links from a "bad" web page. We correspond dataset citation links to "good" links and others to "bad" links. We

[19]We manually compiled a list of dataset related words and phrases, such as: "used dataset from", "gene banks","copora", etc.

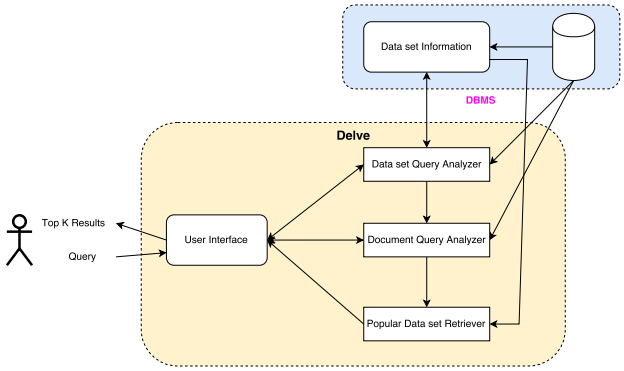


Figure 4: Delve search schema

then, apply PageRank to rank the nodes, with an expectation that highly cited nodes relevant to dataset are ranked higher than others.

To do this, we construct a Markov model M that represents the graph G as a sparse matrix whose element $M_{u,v}$ is the probability of moving from node u to node v in one-time step. We then compute the adjustments to make our matrix stochastic and irreducible (see [26]). The PageRank scores are then calculated using a modified version of the quadratic extrapolation algorithm, which accelerates the convergence of the power method [18]. In the original PageRank algorithm, the PageRank score r_u for node $u \in V$ can be recursively defined as:

$$r_u = \Sigma_{v \in L_u} \frac{r_v}{N_v}, \quad \forall u \in V. \quad (3)$$

where N_v is the number of out-links from node v, and L_u is the set of nodes that are connected to u.

To include the known positive labels of edges, we modified the algorithm such that the PageRank score is recursively defined as:

$$r_u = \Sigma_{v \in L_u} \begin{cases} \frac{r_u}{N_v} & \text{if edge } (v,u) \text{ is 1,} \\ -\frac{r_u}{N_v} & \text{otherwise,} \end{cases} \quad (4)$$

Equation (4) is set such that incoming negative links will decrease the rank score of a node, while incoming positive links will increase the rank score of a node. An initial rank score of $\frac{1}{N}$ is assigned to each node (N is the total number of nodes). After converged ranking scores are obtained for each node, a threshold is applied to nodes, whose incoming citations are labeled as positive if its score is above the threshold, and negative otherwise. It is worth noting that the threshold has a high impact on the inference accuracy. We set it be the 85th percentile after cross-validation.

4.3.2 Delve Search

Figure 4 presents a scenario where a user is searching for a dataset to use as a benchmark dataset for her research project. She enters her research topic of interest as a *query* (input). Delve analyzes this query and presents the user with results (outputs) ranging from matched datasets to papers that used these datasets, which all ranked by relevance. An example of the search result page is shown in Figure 6. To save users' time on filtering out unusable data sets, we separate the invalid datasets (datasets that are no longer available) from the valid ones. Although unavailable (see

the third tab in the middle of Figure 6), the information of these datasets is still presented for the literature review and survey purpose. It is worth mentioning that a dataset node can actually be a paper if it contains direction or descriptions of the dataset used in other papers.

Delve is also capable of handling queries based on snippets of the dataset name. For example, the user might have a dataset in mind, e.g., the PTB Diagnostic ECG Database "http://physionet.org/physiobank/database/ptbdb/". However, the user only knows that the dataset is from *physionet* and is called *ptb*. Entering "*physionet ptb*" will return the correct result.

When a user inputs a query using the user interface, the search phrase is parsed and sent to the dataset query analyzer and the document query analyzer for processing and analysis the search result results. The search schema is made up of three main layers namely: Dataset Query Analyzer, Document Query Analyzer, and Popular Dataset Retriever.

Dataset Query Analyzer (DaQA): Given an input search phrase, DaQA converts the search phrase into a dataset query to search the Delve database for dataset items that match the user search phrase. These dataset items are the nodes associated with positive incoming edges. They can be dataset entities, or paper entities containing dataset information. The matched entities are validated and ranked according to their relevance scores. The result is sent both as output to the user and as input to the Document Query Analyzer to retrieve documents that use the returned dataset items.

Document Query Analyzer (DoQA): The document query analyzer receives as input the user search phrase and the dataset items matching the search phrase. The DoQA converts the search phrase into a document search query, queries the database for documents matching the query, and returns a ranked result. Papers citing the matched datasets items are assigned a boosted weight in the relevance ranking algorithm. The returned results are in turn sent as output to the user and its indexes sent as input to the *Popular dataset retriever* to get the prevalent datasets used by papers matching the search query.

Popular Dataset Retriever (PoDR): The work of the popular dataset retriever is to query the database for the popular dataset items cited by the papers returned by the DoQA. More specifically, it retrieves dataset nodes that have incoming positive links from nodes presenting papers returned by the document query analyzer. These datasets are then ranked according to their citation count.

After the query processing and analysis, Delve returns to the user the result from the different stages of analysis. Figure 6 presents a sample of the result returned by the different stages. The dataset list displayed in the middle panel of Figure 6 shows the results from the DaQA (matched datasets). The result from the first (DaQA) and third stage (PoDR) can be seen under the "matched" and "popular" tabs respectively. Results from the second stage (DoQA) are displayed on the rightmost panel having a list of documents with their metadata including the document title, authors, venue, and abstracts.

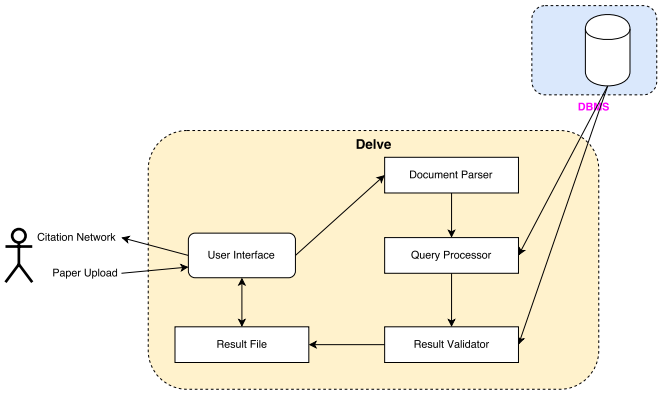


Figure 5: Delve document analysis schema

4.3.3 *Document Analysis*

Delve queries can range from just a keyword search, dataset search, or by uploading a file for on-line analysis. The document analysis provides a medium where researchers can quickly analyze a scholarly document regarding how it is relevant to other documents, without checking the references and searching and reading each of them. This analysis could be for different reasons, like to understand the relationship between a given paper (published or unpublished) with other scholarly papers, to check for other similar papers for further research, or to see the common datasets used in the citation network community of a given paper. Delve allows users to upload the PDF file of the paper for analysis. Delve analyzes the PDF, translates the results into a query, processes the query, and displays the result as a visual citation graph (see Figure 5). The user can then use the Delve citation graph GUI to analyze the paper further (see Figure 8). We plan to provide more information from the document analysis. Further additions will be made to the system later.

The document analysis is organized into three sub-processes namely: document parsing, query processing, and result validation. These sub-processes are explained below:

Document Analyzer: When a document PDF is uploaded, Delve converts the PDF to text using the Linux pdf2text tool. Then using our citation parsing algorithm, we parse and extract the reference list from the academic document. The result is a list of tuples containing the references made in the paper. Each reference tuple is composed of three items - authors, title, and other information.

Query Processor: On receiving the reference list, the query processor sends a query to the DBMS for retrieving the relationship between the document and the items in the Delve database, as well as for the relationship between the references of this paper and items in Delve database. This process is done by converting this paper and each of its references to a database query.

Result Processor: The result processor validates the query result and sets the format to the citation network syntax. The result is written to a temporary file. The file name is returned to the user interface which then reads the file and displays the citation network.

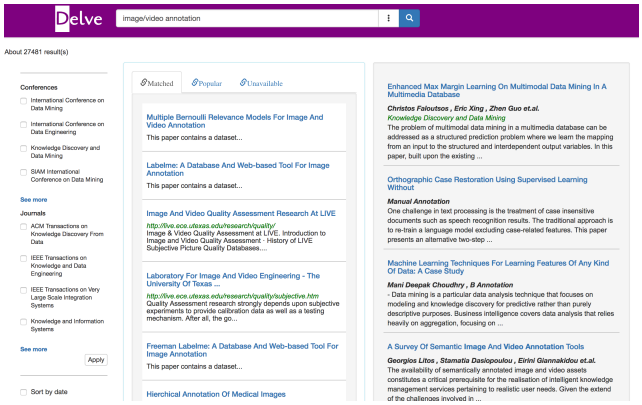


Figure 6: Delve search result page, showing the matched datasets (middle) and documents (right). On the left, there are different filter selections for making advanced search and modifying the returned results.

5. HOW IT WORKS

As discussed before, Delve offers data-driven search and document analysis. In this section, we demonstrate how Delve works in these two modes. For better understanding Delve, please try it at https://delve.kaust.edu.sa to further investigate the interesting results and features provided by Delve.

5.1 Delve Search System

Delve supports two search modes: a normal search and an advanced search. The normal search is structured to be intuitive and simple. The default output includes all the papers in satisfying the query, ranked by relevance as explained in section *4.3.2* above. The advanced search provides an extended medium for querying the system. A user can make use of a combination of different filter selections provided by the Delve web interface to modify the returned results. These filters include searching specific conferences, journals, authors, etc, as shown in the left part of Figure 6.

The search result page area is split into two: the dataset result (displayed in the middle of Figure 6) and document search results (displayed on the right in Figure 6). The dataset results panel is composed of 3 tabs: *Matched*, *Popular* and *Unavailable*. The *Matched* tab contains the datasets matching the search query, the *Popular* tab contains the list of popular dataset used by the documents returned by the document search query, and the *Unavailable* tab contains the temporary or permanently unavailable dataset (whose web links are no longer accessible).

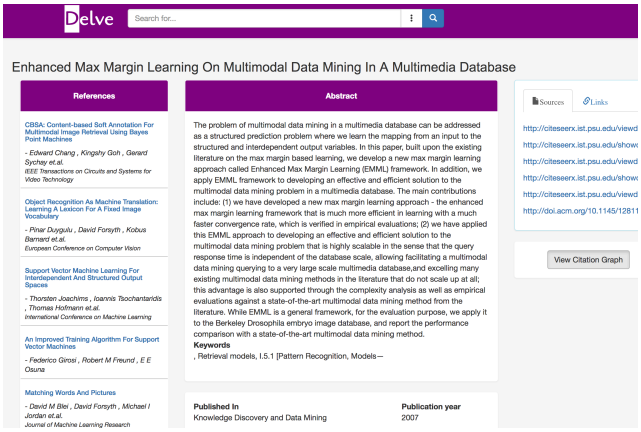
Figure 6 shows the search result of the phrase "image/video annotation". The dataset search result is either a URL (pointing to the web page of the dataset), or a paper (where the dataset is introduced or used). A click on the dataset result, if a link, will take the user to the web page of the dataset. If it is a paper, a click on it will open the information page of the paper. A click on a paper search result item will also open the information page of the selected document, as shown in Figure 7. In this case, it is a paper entitled "Enhanced Max Margin Learning On Multimodal Data Mining In A Multimedia Database" [11]. The information page is composed of several sections :

Document Metadata: including the document abstract, authors, publication year and venue;

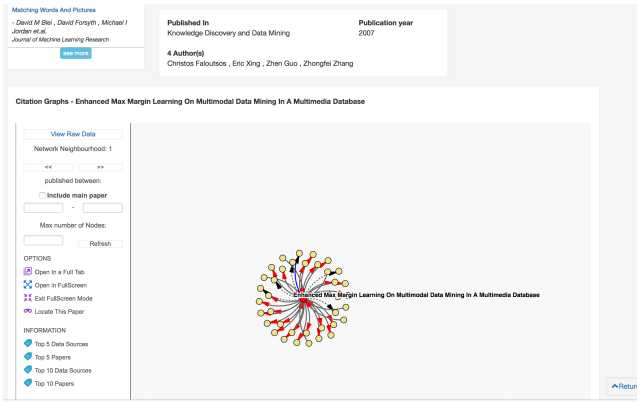
References: this shows the list of references in the paper. Selecting "see more" will open a full list of the document references;

URL section: the section is made up of two subsections - Sources (links to document file) and Links (web links referenced in the document);

Citation Graph: showing the citation network of the paper. More details about this section are presented in Section 5.3.



(a) Information page of the selected item showing the item metadata



(b) Information page of the selected item showing the item citation network

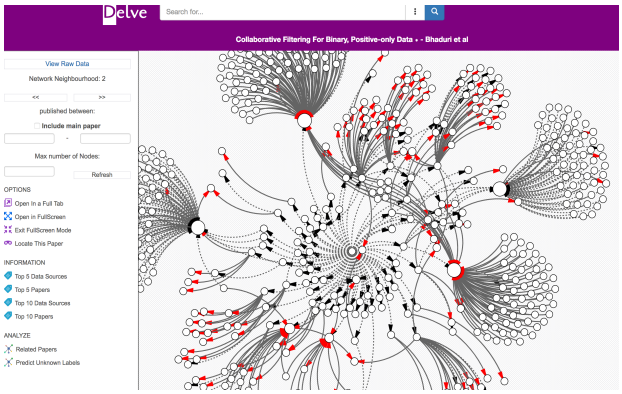
Figure 7: More details about a selected item are provided on the item's information page

5.2 On-line Document Analysis

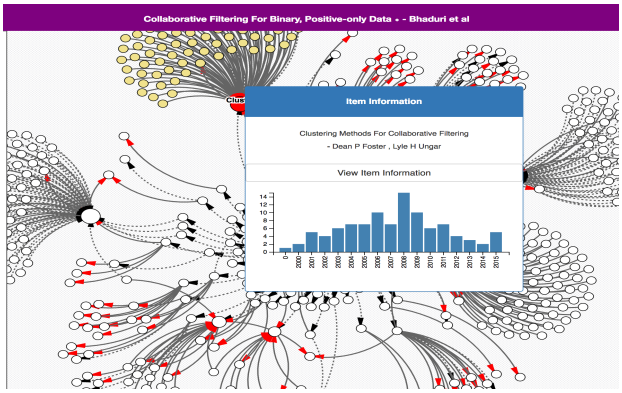
As discussed previously, Delve offers users a document analysis system, which is an important feature of Delve. On the Delve homepage (see Figure 8a), a user can click on "Analyze file", and then select a PDF document to be analyzed, and upload it. The PDF document is then analyzed by the system, and the result is displayed on a new page in the form of a citation graph.

Figure 8b shows the result of analyzing a paper entitled "Collaborative Filtering for Binary, Positively Data" [34], which is recently published at *ACM SIGKDD Explorations*

(a) Delve homepage



(b) Results shown after analyzing a PDF file



(c) Bar chart showing the number of citations per year in the displayed citation network

Figure 8: The citation network produced by the document analysis system after analyzing a given document

Newsletter volume 19 in 2017. It is worth noting that this paper is not included in our system database at the moment of analysis. However, some of its reference papers are included in Delve database. Therefore, we can analyze how this paper and its references are relevant to other papers. The citation network in Figure 8b is centered at this analyzed paper and shows its 2-hop neighbors (by setting *Network Neighborhood: 2* at the top-left corner). Actually, by changing the setting of *Network Neighborhood: k* Delve can display k-hop citation network centered at the analyzed paper. More discussion about the citation network analysis is given next.

5.3 Citation Network Analysis

One of the aims of our system is to provide researchers with a medium to visualize and analyze paper and dataset relationships. Delve provides a simple GUI interface of the citation network of the scholarly documents in its database. Each node represents a paper or a dataset. The color of a node signifies how it is cited. A darker color shows that a node is cited mainly based on dataset. The citation relationship between nodes is shown by a directed link. A blue edge shows a dataset based relationship (with a positive label); a red edge shows a non-dataset based relationship (with a negative label), or a broken edge (with an unknown label). Mouse hovering over a node displays the node title.

The nodes can be interacted with in 3 different way: 1) a left click on a node displays more information about the item - including a bar chart showing the number of citations per year to the selected node based on the displayed citation network (see Figure 8c). Clicking on a bar in the displayed bar chart shows a list of documents citing the selected node document published in the year shown by the bar corresponding to the selected paper. 2) A right-click on a node pops out a list of documents citing or being cited by the node document. And 3) a double-click on a node opens up the information page of the document corresponding to the node.

The tool panel located at the left of the citation network provides some additional tool for analysis. The user can increase the size of the network neighborhood, filter selected papers by year of publication, select the most related papers based on the citation relationship, etc. The raw data used in constructing the citation network can be retrieved by clicking on the "View Data" button.

Another feature of Delve is the online edge inference feature. Currently, Delve uses a modified version of label propagation (see section *4.3.1.1*) to predict the unknown labels in the citation network. When a user clicks on "Infer unknown", Delve reads the citation file, applies the inference algorithm, updates the file with the result and signals the web interface of completion. The web interface reads the updated file and displays the new result.

6. FUTURE WORK

Delve is already launched in public for noncommercial free use. However, it is still young. There are several directions to promote the system. In this section, we present and discuss some future plans for the Delve system.

6.1 Algorithmic Improvement and Database Extension

There are different areas of algorithmic improvement. We plan to improve the document parsing algorithm to improve the Delve database and also the performance of the Delve document analysis system. Another area of improvement is in the citation relationship inference. We plan to apply a more sophisticated inference method. We also need to make all these algorithms efficient for big data.

We plan to extend the Delve database by including papers in conferences and journals out of data mining and machine learning fields, like Bioinformatics, Geology, Biology, Computer vision, etc. With the database extension, we plan to extract more datasets, thus, enriching the Delve dataset database.

6.2 Document Analysis and Citation Network

Currently, Delve shows a binary citation relationship (dataset related and non-dataset related). We plan to extend this to include different types of relationships. For instance, a non-dataset based citation from one paper to another exists probably because of the similar method they used, or because one is the prior work of another, or just because they are from the same authors though having irrelevant content. This feature will improve the document analysis experience as it will provide users with more information about the document.

Another exciting direction is to not only show the citation relationship but also show how citation changes over the years. Knowing how citation changes over time would provide a better understanding of the papers - the significance and impact of the papers to their respective research field and science in general. We also plan to provide more network and document analysis tools. Some features might include the following:

- Recommend uses documents to read, datasets to use, and authors to follow, given the query history;
- Show the top K popular datasets in different research areas;
- Identify influential papers based on the citation network analysis by understanding the roles they played when being cited.

6.3 Structured Document Information

Another plan of our future research direction is to generate structured abstracts from documents texts. A structured abstract is an abstract structured in sections (e.g., objectives, method, results) providing a general summary of the whole document. With this feature, Delve document analysis will provide users with rich and concise information about a given paper, and saving users' time on reading.

7. CONCLUSIONS

The availability and access to dataset have been shown to be a driving factor in several scientific research fields and the advancement of science in general. This paper presents Delve, a system for academic search with a focus on dataset retrieval and document analysis. The Delve search system provides researchers with a medium for data-driven searches. The search result includes datasets and documents ranked by relevance. Delve also presents more information on the documents, the citation network, and useful analysis tools. The Delve document analysis feature allows users to upload the PDF of a scholarly paper and then returns to users a citation graph showing how the given document relates to other documents. Users can further take the citation analysis tools to analyze the results. With additions to the system, we plan to retrieve and show more information from the analyzed paper.

In contrast to prior systems, Delve provides researchers with 1) an easy-to-use medium to locate and retrieve information on relevant documents and dataset, 2) a medium to analyze and visualize the relationship between documents and datasets. This system can answer questions that no scholarly search engine has been able to answer so far. We showed how Delve is beneficial to researchers and for scientific research in general. We believe the Delve system will not just reduce the time required to analyze a paper or find a relevant dataset, benchmark or scholarly document, but will improve the quality of research by providing the user with a platform to understand these entities better and how they interrelate with each other.

8. REFERENCES

[1] About citeseerx. http://citeseerx.ist.psu.edu/about/site.

[2] M. P. Adams, C. J. Collier, S. Uthicke, Y. X. Ow, L. Langlois, and K. R. OBrien. Model fit versus biological relevance: Evaluating photosynthesis-temperature models for three tropical seagrass species. *Scientific reports*, 7, 2017.

[3] U. Akujuobi and X. Zhang. Delve: A data set retrieval and document analysis system. In *ECML-PKDD Demo*, 2017.

[4] C. Cardamone, K. Schawinski, M. Sarzi, S. P. Bamford, N. Bennert, C. Urry, C. Lintott, W. C. Keel, J. Parejko, R. C. Nichol, et al. Galaxy zoo green peas: discovery of a class of compact extremely star-forming galaxies. *Monthly Notices of the Royal Astronomical Society*, 399(3):1191–1205, 2009.

[5] G. Cedersund and J. Roll. Systems biology: model based evaluation and comparison of potential explanations for given biological data. *The FEBS journal*, 276(4):903–922, 2009.

[6] I. G. Councill, C. L. Giles, and M.-Y. Kan. ParsCit: an open-source CRF reference string parsing package. In *LREC*, volume 2008, 2008.

[7] R. P. Duin. A note on comparing classifiers. *Pattern Recognition Letters*, 17(5):529–536, 1996.

[8] B. Efron. [statistical modeling: The two cultures]: Comment. *Statistical Science*, 16(3):218–219, 2001.

[9] Y. Fujiwara and G. Irie. Efficient label propagation. In *Proceedings of the 31st international conference on machine learning (ICML)*, pages 784–792, 2014.

[10] C. L. Giles, K. D. Bollacker, and S. Lawrence. Citeseer: An automatic citation indexing system. In *Proceedings of the third ACM conference on Digital libraries*, pages 89–98. ACM, 1998.

[11] Z. Guo, Z. Zhang, E. Xing, and C. Faloutsos. Enhanced max margin learning on multimodal data mining in a multimedia database. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 340–349. ACM, 2007.

[12] D. J. Hand et al. Classifier technology and the illusion of progress. *Statistical science*, 21(1):1–14, 2006.

[13] H. C. Harris, J. A. Munn, M. Kilic, J. Liebert, K. A. Williams, T. von Hippel, S. E. Levine, D. G. Monet, D. J. Eisenstein, S. Kleinman, et al. The white dwarf luminosity function from sloan digital sky survey imaging data. *The Astronomical Journal*, 131(1):571, 2006.

[14] H. Hirsh. Data mining research: Current status and future opportunities. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 1(2):104–107, 2008.

[15] T. L. Isenhour. *The Evolution of Modern Science*. Bookboon, 2015.

[16] A. J. Jakeman, R. A. Letcher, and J. P. Norton. Ten iterative steps in development and evaluation of environmental models. *Environmental Modelling & Software*, 21(5):602–614, 2006.

[17] M. Janssen, Y. Charalabidis, and A. Zuiderwijk. Benefits, adoption barriers and myths of open data and open government. *Information systems management*, 29(4):258–268, 2012.

[18] S. D. Kamvar, T. H. Haveliwala, C. D. Manning, and G. H. Golub. Extrapolation methods for accelerating pagerank computations. In *Proceedings of the 12th international conference on World Wide Web*, pages 261–270. ACM, 2003.

[19] E. Keogh and S. Kasetty. On the need for time series data mining benchmarks: a survey and empirical demonstration. *Data Mining and knowledge discovery*, 7(4):349–371, 2003.

[20] S. Levy. The gentleman who made scholar, 2015. https://medium.com/backchannel/the-gentleman-who-made-scholar-d71289d9a82d.

[21] M. Lichman. UCI machine learning repository, 2013. http://archive.ics.uci.edu/ml.

[22] National Research Council and others. *Models in environmental regulatory decision making*. National Academies Press, 2007.

[23] National Science Board (US). *Science & engineering indicators*, volume 1. National Science Board, 2012.

[24] N. Padmanabhan, D. J. Schlegel, D. P. Finkbeiner, J. Barentine, M. R. Blanton, H. J. Brewington, J. E. Gunn, M. Harvanek, D. W. Hogg, Ž. Ivezić, et al. An improved photometric calibration of the sloan digital sky survey imaging data. *The Astrophysical Journal*, 674(2):1217, 2008.

[25] N. Padmanabhan, D. J. Schlegel, U. Seljak, A. Makarov, N. A. Bahcall, M. R. Blanton, J. Brinkmann, D. J. Eisenstein, D. P. Finkbeiner, J. E. Gunn, et al. The clustering of luminous red galaxies in the sloan digital sky survey imaging data. *Monthly Notices of the Royal Astronomical Society*, 378(3):852–872, 2007.

[26] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[27] T. Pedersen. Empiricism is not a matter of faith. *Computational Linguistics*, 34(3):465–470, 2008.

[28] S. L. Salzberg. On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data mining and knowledge discovery*, 1(3):317–328, 1997.

[29] I. Strateva, Ž. Ivezić, G. R. Knapp, V. K. Narayanan, M. A. Strauss, J. E. Gunn, R. H. Lupton, D. Schlegel, N. A. Bahcall, J. Brinkmann, et al. Color separation of galaxy types in the sloan digital sky survey imaging data. *The Astronomical Journal*, 122(4):1861, 2001.

[30] A. S. Szalay, J. Gray, A. R. Thakar, P. Z. Kunszt, T. Malik, J. Raddick, C. Stoughton, and J. vandenBerg. The sdss skyserver: public access to the sloan digital sky server data. In *Proceedings of the 2002 ACM SIGMOD international conference on Management of data*, pages 570–581. ACM, 2002.

[31] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 990–998. ACM, 2008.

[32] D. Tkaczyk, P. Szostek, P. J. Dendek, M. Fedoryszak, and L. Bolikowski. Cermine–automatic extraction of metadata and references from scientific literature. In *Document Analysis Systems (DAS), 11th IAPR International Workshop on*, pages 217–221. IEEE, 2014.

[33] J. Vanschoren, J. N. Van Rijn, B. Bischl, and L. Torgo. Openml: networked science in machine learning. *ACM SIGKDD Explorations Newsletter*, 15(2):49–60, 2014.

[34] K. Verstrepen, K. Bhaduriy, B. Cule, and B. Goethals. Collaborative filtering for binary, positiveonly data. *ACM SIGKDD Explorations Newsletter*, 19(1):1–21, 2017.

[35] N. Webster. *Webster's Revised Unabridged Dictionary of the English Language*. G. & C. Merriam Company, 1913.

[36] D. G. York, J. Adelman, J. E. Anderson Jr, S. F. Anderson, J. Annis, N. A. Bahcall, J. Bakken, R. Barkhouser, S. Bastian, E. Berman, et al. The sloan digital sky survey: Technical summary. *The Astronomical Journal*, 120(3):1579, 2000.

[37] X. Zhu and Z. Ghahramani. Learning from labeled and unlabeled data with label propagation. Technical report, Carnegie Mellon University, 2002.