A Data Cleaning Solution by Perl Scripts for the KDD Cup 2003 Task 2

Martine Cadot
LORIA, B.P. 239
F-54506 Vandoeuvre-les-Nancy
France
cadot@loria.fr

Joseph di Martino
LORIA, B.P. 239
F-54506 Vandoeuvre-les-Nancy
France
jdm@loria.fr

ABSTRACT

In this paper, we present our solution for the KDD CUP 2003 task 2 competition. Our approach is based on a data cleaning methodology using Perl scripts. These scripts contain regular expressions for automatically extracting relevant information from the 35472 LaTeX texts. These expressions were optimized by statistical investigations on the texts. Our solution has permitted us to obtain 144,087 associations.

Keywords

Data Cleaning, Text Mining, Statistical Optimization, Perl Scripts, LaTeX files.

1. INTRODUCTION

The task we have chosen consists in creating the list of couples (citing article-cited article) among 35,472 Latex files. For achieving this goal we decided to create two lists. The first one contains as many records as LaTeX files. As for the second one, the number of records is equal to the total number of references cited in the papers. For the two lists the fields of each record contain: paper number, author list, title of the paper, publication date, journal title, journal number, and first page number of the paper. The articles were associated if all the fields of their records (except obviously article number field) were the same. For example, if the record 00001 of the first list is: "/ 00001 / Jean Dupont, Paul Martin / Physical Review History / 2000 / Physical Review D / 35 / 1028 /" and if in the second list there are three records (whose number are 45122, 50000, 87830) with the same fields except the article number, the following three lines were written to the result file: line 1: 45122 00001, line2: 50000 00001 and line 3: 87830 00001.

The records of these two lists must be filled up with data extracted from the texts. The extraction of these data requires to find the part of the text where these data are situated. If we succeed in localizing the data we need, we must then extract them, clean them , and finally normalize them, in order to facilitate the matching of the two lists.

In a first part, we describe the techniques of localization we used, how we normalized the data, and finally how we made the matching between the two lists.

2. LOCALIZATION OF THE INFORMATION

In order to realize this operation, we made shell scripts accounting for each text the number of occurrences of Latex tags. For example, the parameter `\begin`{thebibliography} was present once, in 27,903 texts, absent from 7,175 texts and present more than once in 394 texts[1]. Progressively, the results of these statistics were saved in a 2-dimensional array: the lines of this array corresponded to the article numbers, and the columns to the Latex tags which were searched for[2]. The mining of this array permitted us to refine our Perl scripts we built for the localization of the data. For example, `\begin`{thebibliography}, or `References`, or `references` were found once in 30,090 texts. So, we localized better the cited references with Perl scripts using these three Latex tags in a regular formula. Each research of one of these Latex tags was followed by the opening by hand, of several files which do not contain it, in order to find a new parameter for the localization of the data. As soon as the shell script corresponding to this new parameter was run, the complementary results were incorporated in the array. If the new parameter was found sufficiently pertinent, the corresponding Perl script was modified in consequence.

We thus wrote Shell scripts permitting, despite of the diversity of the Latex tags, to cut the quasi-totality of the texts in 4 parts rid of line feed characters, and of shape parameters useless for data extraction. In the first part we can find the authors of the article, the title of the paper and the definitions concerning the text. The second one was actually the text, containing the references of the bibliography. The third one contained the bibliography and the fourth one was empty or was filled up with annexes.

Unfortunately, no information concerning the journal where the paper was published was localized, and the years of publication were rarely found[3]. Each of the parameters `\author`{ and `\title`{, appeared exactly once in approximatively half of the texts . The bibliographical references were localized without any problem[4], we could efficiently extract the information concerning the bibliography in order to fill up the list number 2. Perl scripts for the list 2 were optimized according to the two files they produced: one file

[1]For example, the text 13197 which includes 7 times the label `\begin`{thebibliography} is the proceedings of a Workshop which contains several articles.

[2]In order to realize this array, we used the SAS and Excel software.

[3]in 8,755 texts `\date`{ was found once.

[4]the parameter `\bibitem` was not found only in 3624 texts.

which contains the correct fields of list 2, and the other one which contains the bibliographical citations not recognized because of tags not foreseen initially in the regular expressions. Of course these scripts were not only dedicated to the extraction, but also to the data cleaning.

3. THE INFORMATION CREATION

In this part, we describe how we have extracted useful information from the texts.

3.1 The journal fields

The bibliographical citations are written in LaTeX, with macro-instructions for some of them. For example, in file 00017, line number 405 and 406 contains:
M. Stephanov, K. Rajagopal and E. Shuryak,
`\Journal {\PRL}{81}{4816}{1998}.`
For interpreting the line 406, we need the definitions in line 22 and 31 of this article:
`\def\Journal#1#2#3#4{{#1} {\bf #2}, #3 (#4)}`
`\def\PRL {\em` Phys. Rev. Lett.}
and we obtain, by a Perl script:
M. Stephanov, K. Rajagopal and E. Shuryak, {`\em` Phys. Rev. Lett.} {`\bf` 81}, 4816 (1998).
and another script permitted us to fill up list 2 with this line : / 00017 / M. Stephanov, K. Rajagopal and E. Shuryak / / 1998 / Phys. Rev. Lett. / 81 / 4816 /
We found in the texts that the referenced journal titles were abbreviated in many different ways. These abbreviations were compared to each other with a tool-box of Excel functions we created in order to choose a sole abbreviation by journal title. To achieve this goal, we needed lists of journal titles present in the Web. But as FAQ of 06/19/2003 forbid this possibility, we did not use the journal title for the matching of the two lists.

3.2 Extraction of the authors

For list 2, the author names and the information concerning the journal were extracted simultaneously by the same Perl script. If a list of authors in a bibliographical reference was followed by several journal references, for each journal reference we created a record in list 2 with the same author list. Perl scripts for extracting the authors for list 1 were more complex. It was difficult to write regular formulae for cleaning the great variety of LaTeX tags contained in the text. In the sequel, follows the author list of file 00010 :
`\author` {Vernon Barger^1 `\footnote` {barger@oriole.-physics.wisc.edu}, Tao Han^{1,3} `\footnote` {than@pheno.physics.wisc.edu}, Paul Langacker^{1,2} `\footnote` {plg@electroweak.hep.upenn.edu}, Bob McElrath$^1 $\`\footnote` {mcelrath@pheno.physics.wisc.edu}, Peter Zerwas$^3 $\`\footnote` {zerwas@mail.desy.de}}
Another problem was to separate the surname and the first name of the first author. There again, to finalize the Perl scripts necessary for doing this task, we peered carefully at the sentences where the names were not correctly separated.

3.3 The year of publication

For list 2, the year of publication was extracted by the same Perl script used to get the title journal.
As it was impossible to find the year of publication for list 1, we decided to approximate the year of publication for each article by the max of all the years cited in the paper[5]. All the years were sought by Perl scripts under the form 19xx or 20xx, where the character x can be any digit. Our result file contained as many lines as papers in the data base and for each line, three fields: the article number, the year obtained by our algorithm and if this year was spurious, i.e. was not found between brackets, the line in the text where this year appeared. This method permitted us to eliminate the uncertainty concerning the year of publication by a quick examination of this third field instead of opening and skimming through the associated text files. We were thus able to associate a year of publication for all the files of the data base except for 180 files.

4. THE MATCHING ALGORITHM

The information concerning the journal was missing from list 1, as for list 2 there was few titles, partial lists of authors, which complicated particularly the matching between the two lists.
We sorted list 1 and list 2 by multiple keys. In decreasing order of priority these keys were: the fist name author, the year of publication, the journal title, the volume number, the first page and the title of the paper. In a first pass, we associated a record of list 2 with a record of list 1, if the name of the first author was the same in the two records and if the year of list 2 record is lower or equal than the year of list 1 record. Actually, the years present in list 2 were exact, as for the years of list 1 were underestimated. In a second pass we added the following condition: the number articles of the two records had to be different.
In a third pass all the list 2 records with the same information had to be associated to the same list 1 record. This pass was not the latest because list 1 was not complete. Actually, whenever we failed to extract the author list of a paper, we put the beginning of this paper in a third list. Finally, in a fourth pass, we should have to match the list 2 records with the list 1 and 3 records.

5. CONCLUSION

When the deadline arrived for this challenge, the result file was obtained only after the second pass. As it was not possible to finish the third and fourth pass, the file we sent was simply sorted and cleaned from its identical lines. Despite the fact that it was not possible for us to finish all the programs, we greatly enjoyed this competition, and we thank the organizers of this fascinating challenge.

6. ACKNOWLEDGEMENTS

This work was supported by the group “Orpailleur” of laboratory LORIA and particularly we thank A. Napoli, the leader of this team who provided us all the technical means, which allowed us to reach the third place in KDD Cup 2003, task 2. We also acknowledge Tarek Ziade who shared with us his knowledge in ”regular expressions”.

[5]we assume that an author cannot refer to papers published in the future.