A strategy for data storage and the search for semi-structured data in the Web

C. A. S. A. do Nascimento[1], N. F. F. Ebecken[1] & J. L. dos A. Rosa[2]
[1]*PEC/Computational Systems, COPPE/UFRJ, Brazil*
[2]*Computing and Mathematics Department, UEZO, Brazil*

Abstract

This paper has the objective of developing a methodology to store a great mass of semi-structured data and its later recovery, focusing on possible improvements and applications by text mining, uniting related articles from many specialties. The present work intends to contribute by way of supplying a strong methodology based on text mining algorithms, searching for a better automated analysis of documents by text classification and making it easy to store and acquire the relevant information in documents on the Web.

Keywords: data mining, text mining, KNN classification.

1 Introduction

There are many texts available in electronic format nowadays. These texts are stored both on the Web and on companies' intranets or agencies in different institutions. With these publications becoming available, the capacity to obtain an enormous quantity of information is growing quickly, but this does not mean that the capacity to read and analyze the information is increasing in the same way. These electronic texts are very usual today, and every second a new one is available to everyone. The continuous growth of available data necessitates new features to extract knowledge and information stored in memos, notices, articles, e-mail messages and Web pages [1,2]. With all of this material ready for use, and without the possibility of reading it all, the great problem is finding the data with which one is concerned.

The search for information is difficult and hard, such that the traditional methods of investigation return many replies. For example: a Google [3] search about "text mining" performed on 2007/02/20 returns over forty millions results.

WIT Transactions on Information and Communication Technologies, Vol 42, © 2009 WIT Press
www.witpress.com, ISSN 1743-3517 (on-line)
doi:10.2495/DATA090061

It is necessary to find a way to obtain fewer responses for queries during the search of documents, with fewer documents with personal concerns. This means that a list of documents that can automatically determine the best results for the investigation must be created [4,5].

One should also consider that the computational effort for the accomplishment of various related tasks is very large. This obstacle is solved by dividing the tasks into modules, so that the workload is distributed among several computers.

The context of a document is composed of the existing relationships (when, where and by whom it was written, in addition to the data for publication). Thus, a document will require a context, since a subject may not be relevant depending on the time of publication. Data written in some places may be more relevant than that written in other documents. Thus, for some applications, it can be very important to analyze the context of the document.

According to Tan [6], there is greater potential in the use of Text Mining than the use of Data Mining. "... 80% of information of a company is stored in text documents."

Currently, the automation of processes has been growing in all areas, not only in relation to Information Technology, and is taken as natural, inevitable and desirable, or a symbol of modernity. There is no denying that applications generate direct impacts at the social level, which arouses the attention of many researchers. This attention has significance and contributes to the understanding of automation [6].

The Internet is a great source of data, easily available to any user, providing the most varied subjects. However, the data are analyzed through the comparison of keywords, indexed in databases, by powerful servers within search engines. These data do not include the intangible value of a search, but only the pure and simple occurrence of a specific word in the content of the files to search.

The automation of a process is the use of technology to simplify, reduce or eliminate human intervention in this process, or to make it automatic. The observed trend can be justified by the need to reduce the quantity of documents to read in order to obtain a smaller number of occurrences that are of greater interest.

This research project will be able to automate the search for relevant documents, using the techniques summarized below. The main application is to continuously classify the new scientific papers that were published in the context of international publications according to the interest of the researchers from the Coppe/Federal University of Rio de Janeiro. In this way, for example, those 300 researchers will receive weekly the top 10 newer published papers closest to the scientific pattern of their work.

2 Text Mining

Text Mining (also known as Text Data Mining and knowledge discovery in text databases), refers to the process of extracting interesting and non-trivial process of patterns from the text documents with unstructured data. It is also seen as an extension of Data Mining; the extraction of knowledge in structured databases [6]. Currently with the tasks of Data Mining consolidated, many efforts are being applied to the area of Text Mining.

Unlike the techniques of Data Mining, Text Mining techniques are designed to make searches on non-structured or semi-structured data, and are not yet consolidated or are not as efficient as those of Data Mining. Text Mining, however, involves much more complex tasks than those of Data Mining [6]. Many solutions have been proposed for the clustering, summarization and categorization of documents in order to facilitate the information.

2.1 Data preparation

2.1.1 Case folding

This is a process used to speed up the comparisons to be made in the preparation of data in Text Mining. It aims to convert all the characters in the same way, all uppercase or all lowercase.

2.1.2 Stopwords

One of the important points of data preparation is to eliminate words that have no representation in the text. These words, commonly used with high incidence, should not be used to represent differences or similarities between documents, since they are very common. The semantic content of words is insignificant, and is not relevant to the analysis of documents. When these words are removed from the index, storage space is reduced and performance is improved during the task of Text Mining. A basic list of words can be used to form a list of stopwords, called a stoplist, to improve the task of Text Mining. A set of 138 words considered universal for the English form was used as a stoplist in the data preparation [8].

Most of these words are conjunctions, prepositions and pronouns that contribute little to the formation of the index [8]. According to the specialty of the text, other lists can be created that are more appropriate, but in this case, the documents to be worked are distinct genera.

Another set of words (such as: http, web, etc.) may be included to in the defined stoplist to also take into account Web documents. These words also slightly contribute to the formation of the index.

2.1.3 Stemming

It is the task of the merger or combination to equal morphological variants of texts, which is described below [9–11].

The stemming process aims to reduce the radical (root), so that different words with the same meaning are not counted as separate words. The radical is a word used in a query and the answers may bring documents not relevant to the context required. It turns out that a variant of that word can be found in other documents with higher relevance. Thus, the query can be more efficient when working with the radicals of the words instead of the original word. The same word can have, among others, variations such as plurals, gerund forms of suffixes and prefixes [9,10,12,13]. The technique is then to remove the suffixes and prefixes, so that the same words with similar meanings have radicals [14,15].

The stemming process is dependent on the language, and rules should be adapted for different that languages exist, i.e., the algorithms must have modified their rules in order to meet a certain language. For English we can identify the words cats, catlike and catty as the radical cat. The first publication on the subject appeared in 1968 in the work of Julie Beth Lovins. In 1979 the stemming algorithm [17] was published, which was the basis for the creation of the widespread stemming algorithm, the Porter algorithm [18]. There are currently several stemming algorithms available.

3 Classification data

The KNN or K Nearest Neighbors classification algorithm is considered a very simple method and is based on the identification of groups of data with similar characteristics to its further assembly. This method uses the concept of distance between samples. For each sample a vector is created (line) containing several variables (columns). The similarity is measured by calculating the distances between samples.

The calculation of the distance between the unknown to known samples can be done by any of the following methods [19–21]:

- Manhattan Distance

$$d(x,y)=\sum_{i=1}^{n-1}\left|x_i-y_i\right| \qquad (1)$$

- Minkowski Distance

$$d(x,y)=\sqrt[q]{\sum_{i=1}^{n-1}w_i\left|x_i-y_i\right|^q} \qquad (2)$$

where q = 2 and wi = 1.

- Euclidean Distance

$$d(x,y)=\sqrt{\sum_{i=1}^{n-1}(x_i-y_i)^2} \qquad (3)$$

The value of K indicates the number of nearest neighbors that will be used to identify a new sample for all samples. Thus, the more samples of a given class are close to the unknown, the greater the probability that this sample belongs to that particular class. The most used is Euclidean distance. Figure 1 gives the algorithm of the method of KNN classification [19,21].

```
 Begin
   Read the n samples of known classes ($ac$)
   Read a sample of unknown class ($ad$)
   i ← 1
   Repeat
      Calculate the distance $ad$ for $ac_i$
      Distance vector ← distance and class of $ac_i$
      i ← 1 + 1
   Until (calculating all distances, ie, $i > n$);
   Sort the vector of distances for the values of
distances
   Select the first K elements of the vector of distances
   i ← 1
   Repeat
      Compute $\mu_i (x)$
      i ← 1 + 1
   Until (calculate all $K$ samples, ie, $i > K$);
End.
```

Figure 1: KNN classification method.

4 System architecture

The system is started by access to a Web application. Accreditation is requested from the system and a login is obtained. With this login, a profile is created by including the summary of one's interest. This profile will be used for publication classifications available considering only the summaries. In this way the ranked list of relevant publications is created. The system is responsible for making the classification, generating the list, and refreshing the page and the user's query, where this is made necessary, automatically.

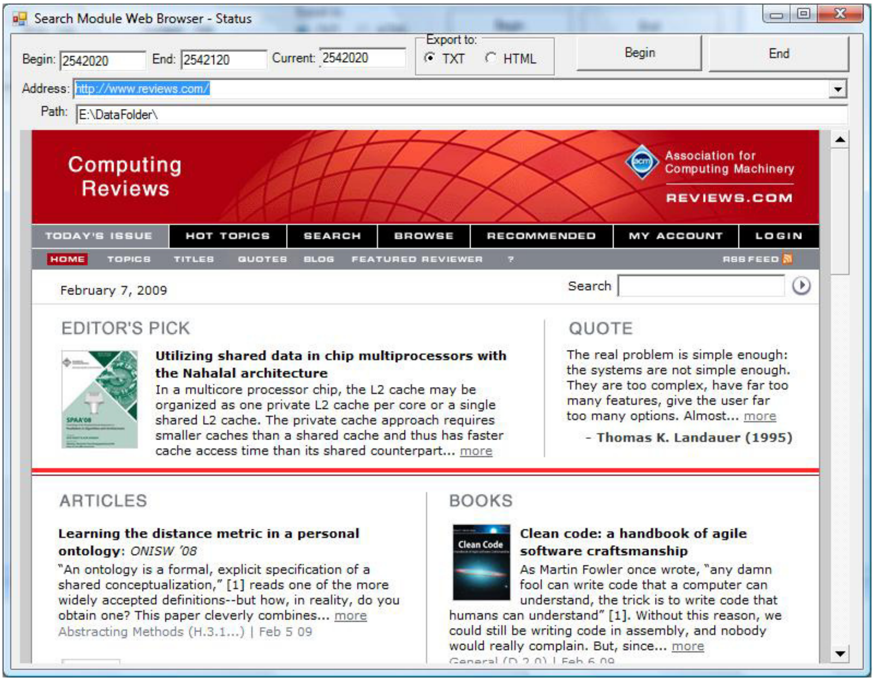
The application will be responsible for seeking the documents through the portal of CAPES (Coordination of Graduated Programs in Brazil that includes a very complete list of periodicals and journals) in two different ways: automatic and manual. The automatic form that is the main objective of this project is used for new publications available.

The application was divided into three different modules: the search for information on the Internet, the preparation and storage of semi-structured data, and building the list of documents.

4.1 The search module

A major problem is the search for documents via the Internet. These documents are usually stored in different locations and take much time to search. The search will access a database containing the main sites for the search of the publications, which will be automatically found and the files will be running constantly, always looking for new publications that are included in the CAPES portal. This tool consists of an Internet browser equipped to access a database containing the local search, to make access to these places and obtain the contents of the document; it does not need to be read and its contents will be stored on the server machine.

The application will allow separate documents to be inserted; they are simply stored in the storage area and are registered in the database. Information such as date of incorporation and place of origin is part of the context of the document. There will also be the ability to include new local searches by adding new locations based on periodical publications. Figure 2 shows an example of the Web Browser Status Screen.



Search Module Web Browser - Status

Begin: 2542020 End: 2542120 Current: 2542020

Export to: ● TXT ○ HTML

Address: http://www.reviews.com/

Path: E:\DataFolder\

February 7, 2009

EDITOR'S PICK

Utilizing shared data in chip multiprocessors with the Nahalal architecture
In a multicore processor chip, the L2 cache may be organized as one private L2 cache per core or a single shared L2 cache. The private cache approach requires smaller caches than a shared cache and thus has faster cache access time than its shared counterpart... more

QUOTE

The real problem is simple enough: the systems are not simple enough. They are too complex, have far too many features, give the user far too many options. Almost... more

- **Thomas K. Landauer (1995)**

ARTICLES

Learning the distance metric in a personal ontology: *ONISW '08*
"An ontology is a formal, explicit specification of a shared conceptualization," [1] reads one of the more widely accepted definitions--but how, in reality, do you obtain one? This paper cleverly combines... more
Abstracting Methods (H.3.1...) | Feb 5 09

BOOKS

Clean code: a handbook of agile software craftsmanship
As Martin Fowler once wrote, "any damn fool can write code that a computer can understand, the trick is to write code that humans can understand" [1]. Without this reason, we could still be writing code in assembly, and nobody would really complain. But, since... more
General (D.2.0) | Feb 6 09

Figure 2: Search module Web browser – status screen.

4.2 The preparation module

The module is a tool for Text Mining, which is able to monitor the storage space of documents supplied by the search module. Its main functions are the identification of new files available and the preparation of data for further processing by the Research Module. This phase aims to increase the processing speed, since each user will have a different search and does not need to prepare the data as they will already be ready. A document that reaches the main server will be stored in a directory. All documents received will be submitted to the stage of preparing the data, which will be divided into three different tasks. Before the first task, the text that was stored in the directory will be transferred for arrival to a working directory. In the working directory, the text is then submitted to an algorithm capable of extracting relevant words and eliminating any type of data that do not form words. The result is an XML document containing the list of words, with no duplication, but it is possible to know the number of occurrences of each word in the text. The document generated is stored in the database, with information on the date of acquisition, the source of publication, the location of the original document and a status of reading. The result of the preparation is stored in another directory, which will be available to perform the search for the next module.

The following tasks are performed:

- folding Case - all the letters of all words are transformed to capital letters;
- stopwords - the resulting text is submitted to the algorithm, which is able to extract the words, removing any kind of data that do not form words, and remove all words that are in the list of stopwords and words that are irrelevant to the context.
- Stemming – the final task is the preparation of the data, which is the task of stemming the document using the Porter stemming algorithm [22].

4.3 The research module

After preparation, the XML file created contains the events that have significance for the classification and quantity of their occurrences. This facilitates the construction of vectors for classification (composed of the quantities of occurrences found) that will be compared with the vector created from the profile the user created during the registration system through the technique of classification by KNN. The Browser Module is responsible for processing and creating a final list of publications of greater relevance to the user. The list can be parameterized in order to better serve the needs of the user. The main items to be configured are:

1. number of documents in the list of publications of major significance;
2. if the application will check in all the publications stored or whether the user will select those considered to be most important;
3. period of search for documents by date of purchase, specifying free period for the beginning and end, or from when and to.

The list is also adjustable on the removal of documents that have already been read; it is easy to find the same as the user can directly access the publication of interest via a link.

5 Conclusion

The methodology will provide the means to organize and automate the tasks of research in journals to the user, considerably increasing production and enabling the maximization of time spent for this research. Its division into modules also supports distributed processing, the incorporation of new algorithms for preparation or classification of their features, increased options-conditioning tasks and actions to be executed. The system is now undergoing evaluation, where small researcher groups are testing and sending comments about the performance and interest of the obtained results.

References

[1] Zanasi A., Ebecken N., Brebbia C.A.: Web Mining through the Online Analyst. Conference on Data Mining 2000, pp. 3-14, Cambridge, July WIT Press 2000.

[2] Lopes, M. C. S.: Text Mining in portuguese., COPPE/UFRJ publication Rio de Janeiro, 2004.

[3] Google. WWW http://www.google.com.

[4] Agichtein E., Zheng Z..: Identifying "Best Bet" Web Search Results by Mining Past User Behavior. Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM Press New York, NY, USA, 2006.

[5] Joachims T., Granka, L., PANG, B., Hembrooke H. and Gay G.: Accurately Interpreting Click through Data as Implicit Feedback. In Proceedings of SIGIR, 2005.

[6] Tan A.H.: Text Mining the State of the Art and the Challenges. In Proceedings of the PAKDD 1999. Workshop on Knowledge Discovery from Advanced Databases, pages 65–70, Beijing, China, April 1999.

[7] Vieira, P. A.: Work automatization in: http://orbita.starmedia.com/~novosdebates/textos/t10.htm.

[8] Konchady M.: Text Mining Application Programming. 1a Edition, Charles River Media, Boston, Massachusetts, 2006.

[9] Lopes, M. C. S.: Text Mining for the Portuguese Language: clustering documents. Doctoral Thesis COPPE/ UFRJ, Rio de Janeiro, 2004.

[10] Chaves, M. S.: Stemming algorithms for the Portuguese. IX Iberoamerican Informatics Workshop. Cartagena de Indias - Colômbia, 2003.

[11] Frakes, W.B., Baeaza-Yates, R.: Readings in Information Retrieval: Data Structured Algorithms. Ed. Upper Saddle River, NJ: Prentice Hall, 1992.

[12] Spark-Jones, K., Willet, P.: Readings in Information Retrieval. San Francisco: Morgan Kaufmann, 1997.

[13] Baeza-Yates, R.: Modern Information Retrieval. New York, N.Y.: Addison-Wesley, 1999.

[14] Martha, A. S. Barra, P.S.C., Campos, C. J. R.: in WWW por http://www.sbis.org.br/sbis/arquivos/636.pdf.

[15] Wives, L.K.: Document indexing. http://www.inf.ufrgs.br/~wives/publicacoes/IDT.pdf.

[16] Lovins J. B.: Development of a Stemming Algorithm. Mechanical Translation and Computational Linguistics, 11, 22-31, 1968.

[17] Porter, M. F.: Stemming Algorithm Paper. Computer Laboratory Cambridge England, 1979.

[18] Porter, M. F.: The Porter Stemming Algorithm. in WWW http://www.tartarus.org/~martim/PorterStemmer/index.html.

[19] Rosa, J. L. A.: Data Classification by an optimized Fuzzy-KNN algorithm implemented in a Parallel Environment. Doctoral Thesis COPPE/ UFRJ, Rio de Janeiro, 2003.

[20] Miller, G.: Wordnet: An online lexical database, International Journal of Lexicography, 3(4):235-312, 1996.

[21] Han J.: Kamber, M., Data Mining: Concepts and Techniques, Morgan Kaufmann, 2001.

[22] Porter, M. F.: An Algorithm for Suffix Stripping Program, 14(3), 130-137, 1980.