Manual and Evolutionary Equalization in Text Mining

GIOVANNI BONANNO, FRANCESCO MOSCHELLA, SALVATORE RINAUDO
IMS Group R&D
STMicroelectronics
St.le Primosole, 50 - 95121 Catania
ITALY

http://www.st.com

PIETRO PANTANO
Evolutionary Systems Group
Department of Linguistics - University of Calabria
ponte P. Bucci - Cubo 17B - Campus di Arcavacata – 87036 Arcavacata di Rende, Cosenza
ITALY
http://galileo.cincom.unical.it/esg/

VALERIO TALARICO
Mathematik und Naturwissenschaften Lehrstuhl fuer Angewandte Mathematik/Numerik
Bergische Universität
Gaussstrasse, 20 – Wuppertal
GERMANY
http://www.comson.org

Abstract: - The phase of Text Retrieval or Information Extraction may represent a weakness for the entire text mining process. During this phase, keywords are extracted that can be used in documents classification or clustering. If the extracted keywords are not meaningful, the entire text mining process will be compromised. This risk is high in the event of heterogeneous sources of documentation, if the keywords extraction method does not count occurrences or, in general, does not compute any statistics. We propose an approach the makes it possible to define, as a first step, which parts of the document will predominate on the others. We call this step equalization. We will show how equalization can be customized in relation to the different sources of information and used both manually, by using a specialized graphical user interface (GUI) and semi automatically, by using a suited genetic algorithm.

Key-Words: - Text Mining, Text Retrieval, Information Extraction, Data Mining, Document Equalization, Digital Libraries

1 Introduction

In this paper we refer to the field of and technologies related to DMS (Document Management Systems). In DMSs several information sources (digital libraries, WEB sites, documentation archives and so on) are integrated, in order to create a single access point to all available documentation. In general, in such systems no assumption is made upon the structure and the nature of the content of documents. We will restrict our problem domain referring to a specific class of DMSs that store, process and distribute documents containing at least text. Such text will be considered to be in an unstructured form. In this paper, a particular attention will be put on the extraction of keywords from non structured documents belonging to different sources. We will show how our approach allows the user to define what parts of the document mostly influence the keywords extraction process. To accomplish this objective, we have created an equalization function controlled by a graphical user interface.

This method can be applied to all those cases in which weight extraction from structured or semi structured documents [1, 2] is not possible because no assumption can be made a priori on such structures and/or the corpus of document is heterogeneous.

2 Text Mining Process

The following figure shows the entire Text Mining process highlighting the role and position of the equalization phase. >From this point onward each reference to the Text Mining process will refer to the process as illustrated in the following figure.

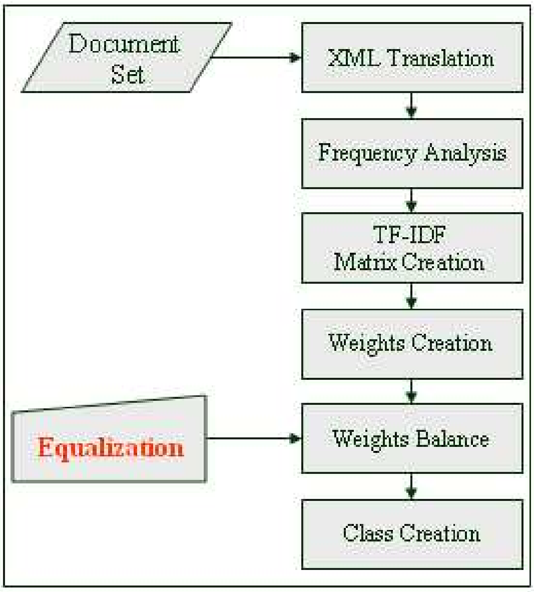


Fig. 1

The Text Mining process , as illustrated in Fig. 1, will be now briefly discussed.

Initially, a phase of preprocessing occurs: documents are suitably reduced and insignificant words are rejected. This process consists of three steps:

1. Translation of the corpus of documents into XML format;

2. Elimination of the "stop-words" (conjunctions, articles, prepositions), which are surely irrelevant words;

3. Search of the common root among the terms (stemming).

Once the execution of this phase is completed, it is possible to attribute a weight to all the words through the generation of the TF-IDF (term frequency – inverse document frequency) matrix [3, 4]. This allows to place words into classes, according to their weight (the relevance inside the document corpus). Such classes can be subsequently clusterized by generating document classes. In the next paragraph we will describe the equalization phase. Its aim is to balance the obtained weights, taking into account the various kinds of analyzed documents.

3 Equalization

One of the limitation of the traditional TF-IDF approach is that weights are functions of word frequencies.

We have observed that, in several cases, it is useful to consider weights that are not only function of frequency but also function of the position of the word within the text. In fact, there are many particular situations, in which traditional TF-IDF will tend to give excessive weight to word coming from parts of the document that are not as relevant as others. For example, very badly estimated weight result form analyzing documents that contain header, bibliography, and, in general, any other "dirty" text deriving from the XML conversion. This is especially true with web pages in which relevant content is often surrounded by news, advertisement, related links etc. Such surrounding text can badly influence the evaluation of weights. Extremely negative consequences on weight estimation can be observed when we analyze many documents extracted from a web site in which a series of constant common surrounding section are translated with each relevant content. In this case words belonging to the surroundings will appear to be very frequent in each and every document thus gaining an high degree of relevance which is indeed totally wrong.

It is thus clear that in all of these situations, it is useful to attribute a greater weight to word that are more interesting because they belong to parts of the converted document that are more relevant than other parts within the same document. On the contrary, lower weight will be attributed to those word that belong to less interesting parts of the document.

Based on this observation we have designed the equalization phase. In order to describe this phase, we will note first how relevant words that summarized the arguments treated in a text (also identified as keywords) can be more likely found in the title and the central part of the text itself.

The equalization phase can be divided in three steps:

1. Definition of the equalization parameters provided by the user;

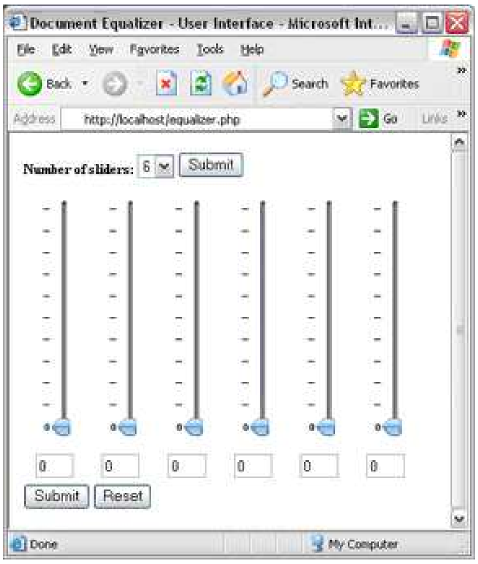
2. Creation of an equalization function based on equalization parameters;

3. Application of the function to the existing TF-IDF weights.

3.1 Definition of equalization parameters

In order to solve the problem of parameter definition, a method is proposed that exploits web distributed systems. Interactivity provided by the GUI is accomplished by the use web scripting language, consisting in a combination of Php and JavaScript.

The advantages of this approach are remarkable: there is no need to install the tool on the single client machines; it is independent from the operating system, simple to update, as the application physically resides on the server and users can easily share data which are stored on the server. Users can input data through sliders, can choose how to set the slider to obtain the desired value, input it directly or load a set of values that were previously saved. Users can also decide the number of sliders to be used, according to the precision they want to obtain in the interpolation of the equalization function.



Address: http://localhost/equalizer.php

Number of sliders: 6

0 0 0 0 0 0

Document Equalizer - User Interface - Microsoft Int...
File Edit View Favorites Tools Help
Back
Search
Favorites
Go
Links
Submit
Submit
Reset
Done
My Computer

Fig. 2

Fig. 2 shows the GUI used to input parameters. Each parameter is associated with a slider. The slider can be used to set the parameter's value. Users can control the number of sliders (6 in Fig. 2) by using the "Number of sliders" control. Possible values are 5, 6 and 11. In this case, the entire document is divided respectively in 4, 5 and 10 intervals which represent 25%, 20% and 10% of the entire document. Theoretically, it could be possible to increase the number of sliders until 100 is divisible for the number of intervals. However, having more than 11 sliders to be set is surely not user-friendly.

3.2 Interpolation of the equalization function

In order to generate the equalization function, we use the cubic spline interpolation method. A spline is a particular function defined in separated intervals of its dominion through polynomials. In case of cubic spline, polynomial expressions are of the third degree. The interpolation method through spline is often preferred to the polynomial interpolation, as it generates similar results in polynomial expressions of lower degree and avoids the Runge phenomenon in polynomial expressions of higher degree.

Given $m+1$ knots t_i with $t_0 \le t_1 \le \ldots \le t_m$ a B-Spline of degree n is a parametric curve $\mathbf{S}:[t_{0,}t_m] \to \Re^2$, made of **basis B-splines** of degree n:

$$S(t) = \sum_{i=0}^{m} P_i b_{i,m}(t), t \in [t_0, t_m]$$

The $b_{i,m}(t)$ are polynomial expressions of degree n. For a correct interpolation of the curve, it is necessary to supply n points P_i. Such points are called control points or De-Boor points. A polygon can be constructed by connecting De Boor points to the lines starting with P_0 and finishing with P_m; this polygon is called De Boor polygon. Then, applying the Cox-de Boor recursion formula, it is possible to generate a B-spline of degree n:

$$b_{j,0}(t) = \begin{cases} 1 & \mathit{if} \quad t_j \le t \le t_{j+1} \\ 0 & \textbf{otherwise} \end{cases}$$

$$b_{j,n}(t) = \frac{(t - t_j)}{(t_{(j+n)} - t)} b_{j,(n-1)}(t) + \frac{(t_{(j+n+1)} - t)}{(t_{(j+n+1)} - t_{(j+1)})} b_{(j+1),(n-1)}(t)$$

In our case, the B-Spline is generated through a polynomial expression of the third degree and is uniform, as the nodes are equidistant from each other. The formula to be applied is:

$$\mathbf{S}(t) = \sum_{i=0}^{n} P_i b_{i,3}(t)$$

The advantage of the interpolation through B-spline is that we can have whichever number of check points independently from the degree of the polynomial interpolation, differently from Bézier curves, where the degree of the polynomial expression must be equal to the number of check points.

Once the polynomial expression of third degree has been calculated, it is possible to obtain all points necessary to the interpolation.

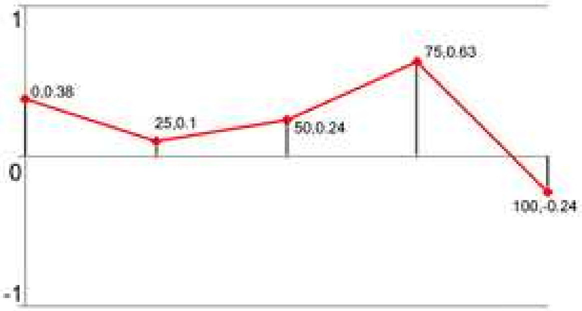


Fig. 3

In our case, the polynomial expressions by which we interpolate P_i support points are of the following type:

$$p_i(x) = a_i(x - x_i)^3 + b_i(x - x_i)^2 + c_i(x - x_i) + d_i$$

It is necessary to set the following continuity conditions for the polynomial expressions and their first and second derivatives:

$$p_i(x_i) = y_i$$
$$p_i(x_{i+1}) = y_{i+1}$$
$$p_i'(x_{i+1}) = p_{i+1}'(x_{i+1})$$
$$p_i''(x_{i+1}) = p_{i+1}''(x_{i+1})$$

The parameters p_i of the polynomial expression are function of the second derivatives:

$$a_i = \frac{\left(p''_{(i+1)}\left(x_{(i+1)}\right) - p_i''(x_i)\right)}{\left(6\left(x_{(i+1)} - x_i\right)\right)}$$

$$b_i = \frac{\left(p_i''(x_i)\right)}{2}$$

$$c_i = \frac{\left(y_{(i+1)} - y_i\right)}{\left(x_{(i+1)} - x_i\right)} - \left(x_{(i+1)} - x_i\right)\frac{\left\lfloor p_i'' + 1\left(x_{(i+1)}\right) - 2p_i''(x_i)\right\rfloor}{6}$$

$$d_i = y_i$$

3.3 Apply the results to the weights

The last operation to do is mining the useful number from the interpolation function and applying it to the existing weight.

In order to find the correspondent value of the function to the word position, it is necessary to have an array containing the positions of the words inside of the document.

The array will be:

Numeric key → array [document id, word id, value]

And can be easy calculated from the xml file, taking into account that the position in percentage of the word inside the document is given by the formula

$$wordposition = \frac{(word.number)}{(tot.number.of.words)} * 100$$

For example, the position of word number 5 in a document made of 10 words will be (5/10) *100, that is 50. And, effectively, the word is in the first half of the document (50% position).

In order to avoid having several decimal figures, it is sufficient to approximate the number to the closer integer. Once obtained the array of the positions, we can proceed to the equalization, remembering that a word can appear many times in a document.

In order to extract the opportune values from the generated array, through the interpolation of the function, we consider the word position as the value of x axis, whereas axis y values represent the weight to add or subtract to the word.

Once this operation is complete, we obtain an array of balanced value such as:

ID Document	ID Keyword	Value

4 Evolutionary equalization

The initial choice of the equalization parameters can be sub optimal, especially when dealing with new sets of documents entering the corpus that need to be properly associated with key words and clusterized. In this case, an approach based on genetic algorithm can be implemented.

Other approaches have been proposed in the past that make use of bio inspired techniques. In [10] a technique has been proposed that makes use of genetic programming in order to evolve weighting schemes for information retrieval. Such an approach was generic because it did not stat from any preoptimization scheme such as the one proposed by our equalization method and showed to perform poorly for a large corpus of documents.

Other approaches not based on preoptimization appear to be either ineffective or too slow.

Our approach promises to overcome speed and effectiveness issues by providing a combination of preoptimization provided by equalization and user driven fitness evaluation.

This approach is characterized by having a set of equalization parameters coded as a genotype to and individual. We let evolve and let compete a population of individual with different genotypes. Their phenotypes are represented by the effect on weighting, and thus clustering of documents, of their genotype. The fittest individuals are selected among those providing the best clustering performances and allowed to mutate and reproduce. The iteration of such algorithm converges towards an arrangement of parameters that perform very well in clusterization. The fittest individuals are selected by the web distribution service that is used to retrieve document. The user drives the selection by using the web interface to express satisfaction or delusion with respect to the results returned by the information retrieval system using a given set of keywords. The reiteration of the selection process for a few generations is believed to converge rapidly towards a wel performing weighting and clustering.

The initial values for the genotype of the first generation of individuals can be chosen randomly. However, the algorithm could perform significantly better if the initial population is generated starting by slight o no variation with respect to the active set of equalization parameters already present in the system.

5 Conclusion

The weight balance, obtained through the equalization process described in this work, has allowed us to reach excellent results. Our DMS flow ends with the clustering phase, which groups million documents into classes. Our clustering starts from keywords that are common to all documents; thus, it is fundamental that are keywords extracted in the first phase describe best their contents.

By applying the equalization function to our flow, we have noticed an improvement in cluster creation. For example, lowering the weight of keywords extracted from articles references, we have drastically reduced cases in which documents were considered similar to each other because of personal or place names. Similarly, huge improvements have been accomplished in clustering documents coming from the web by lowering the relevance of certain parts of the web page containing them.

Finally, further developments based on semi automatic hybrid techniques based on equalization and preoptimization appear to be able to provide very effective solutions for dynamic sets of documents.

References:

[1] Yang Jianwu , Chen Xiaoou, A semi-structured document model for text mining, Journal of Computer Science and Technology, v.17 n.5, p.603-610, May 2002

[2] Jeonghee Yi , Neel Sundaresan, A classifier for semi-structured documents, Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, p.340-344, August 20-23, 2000, Boston, Massachusetts, United States

[3] Salton, G. and McGill, M. J. 1983 Introduction to modern information retrieval. McGraw-Hill, ISBN 0070544840

[4] Salton, G. and Buckley, C. 1988 Term-weighting approaches in automatic text retrieval. Information Processing & Management

[5] Weisstein, Eric W. "B-Spline." From MathWorld--A Wolfram Web Resource. http://mathworld.wolfram.com/B-Spline.html

[6] C. de Boor, List of m-Files for Doing Least Interpolation. [Online]. Available: http://www.cs.wisc.edu/~deboor/multiint/m_files.html.

[7] T. A. Grandine, MVP, a Package Designed to Create, Evaluate, and Manipulate Multivariate Polynomials. [Online]. Available: http://www.netlib.org/a/mvp.tgz

[8] L. Schumaker. Spline Functions: Basic Theory. John Wiley & Sons, New York, 1981.

[9] M. G. Cox. The numerical evaluation of B-splines. J. Inst. Math. & Applic., 10:134-149, 1972.

[10] Cummins, R. and O'Riordan, C. 2006. Evolving local and global weighting schemes in information retrieval. Inf. Retr. 9, 3 (Jun. 2006), 311-330.