An Analysis of Constructed Categories for Textual Classification Using Fuzzy Similarity and Agglomerative Hierarchical Methods.

Marcus Vinicius C. Guelpeli   Ana Cristina Bicharra Garcia
Departamento de Ciência da Computação
Universidade Federal Fluminense – UFF – Brazil
mguelpeli@ic.uff.br   bicharra@ic.uff.br

Abstract

Ambiguity is a challenge faced by systems that handle natural language. To assuage the issue of linguistic ambiguities found in text classification, this work proposes a text categorizer using the methodology of Fuzzy Similarity. The grouping algorithms Stars and Cliques are adopted in the Agglomerative Hierarchical method and they identify the groups of texts by specifying some time of relationship rule to create categories based on the similarity analysis of the textual terms. The proposal is that based on the methodology suggested, categories can be created from the analysis of the degree of similarity of the texts to be classified, without needing to determine the number of initial categories. The combination of techniques proposed in the categorizer's phases brought satisfactory results, proving to be efficient in textual classification.

1. Introduction

The access to means of distributing information is becoming easier and easier. Motivated by the great availability of computational resources and the ease of exchange and storage of information, institutions from the most diverse spheres of knowledge have produced and electronically stored a vast amount of data.

Currently, companies have started to make their products available in these means of distribution, expanding their markets in a global fashion and maximizing their profits. Hence, the amount of information is currently vast and growing with each passing minute. This amount of information, as well as being vast, is organized in a disorganized fashion and is not standardized, which makes location and access more difficult.

As such, it becomes necessary to resort to computational methods that automatically classify available documents, with the purpose of recovering information with greater speed and fidelity when it comes to the content matter of the texts, thereby allowing them to be useful in the decision-making process.

Computational methods contribute to the advent of computational systems that are capable of acquiring new knowledge, new skills and new ways of organizing existing knowledge[5].

Text Mining (TM) is a new, multidisciplinary field, which includes spheres of knowledge like Computing, Statistics, Linguistics and Cognitive Science. This method consists of extracting regularities, patterns or trends in large volumes of texts written in a natural language, usually, for specific purposes. Inspired by Data Mining (DM), which tries to uncover patterns emerging from structured databases, text mining looks to extract useful knowledge from non-structured or semi-structured data.

Text mining can be applied in a variety of contexts: in the creation of summaries; in clusterization (which refers to the grouping of texts according to similarities in their content matter); in identifying languages; to extract terms; in text categorization; to manage electronic mail and manage documents and to in market research and investigation.

Categorization in TM, also known as Knowledge Discovery in Text (KDT)[8], or even as Text Data Mining[7], is the result of the symbiosis between Information Recovery, Machine Learning, Statistics and Databases. This field is designed to analyze and extract knowledge from collections that are comprised of large volumes of non-structured textual documents, with the purpose of identifying

978-0-7695-3122-9/08 $25.00 © 2008 IEEE
DOI 10.1109/SITIS.2007.109

Authorized licensed use limited to: Universidade de Lisboa Reitoria. Downloaded on September 11, 2009 at 10:45 from IEEE Xplore. Restrictions apply.

the mains categories in a text and connecting this same document to one or more predefined categories. Hence, it is an attempt to transform implicit knowledge into explicit knowledge [4].

The aim of this work is to propose a categorizer using the Fuzzy Similarity methodology, to improve the issue of linguistic ambiguities present in the classification of texts, and use the Agglomerative Hierarchical method to create categories with a basis on the similarity analysis of textual terms. The proposal is that with the suggested methodology, we can create categories through the analysis of the degre of similarity of the texts that are to classified.

This paper is organized as follows. In Section 2, the thereotical concepts of Fuzzy Similarity are introduced. Section 3 covers Hierarchical Methods and concepts. Section 4 presents the methodology employed, with the relative frequency calculation used in teh characteristic selection, the Fuzzy measure (set theoretic inclusion) used to obtain the similarity matrix and the use of the groupoing algorithms Stars and Cliques, which are used in the Agglomerative Hierarchical method to identify groups of texts by specifying some type of relationship rule. Section 5 discusses the results obtained using the Categorizer. Finally, in Section 6, we present the conclusions drawn from the proposed method and suggest future studies.

2. Fuzzy Similarity

Ambiguity is the greatest challenge faced by systems that handle natural language. Identifying the real meaning of a given word can be so complicated, that it is sometimes only possible by consulting with the user.

In the process of choosing an alternative mathematical treatment that is more appropriate for questions formulated in natural language, there is a big advantage in opting for the use of fuzzy logic. Conventional, binary logic, based on principles of true/false, presents some difficulties when it comes to representing abstract concepts.

[10] describe a number of applications for fuzzy logic, ranging from research in the natural sciences to studies in social sciences, including areas such as engineering, medicine and decision-making systems. In terms of using fuzzy logic to translate natural language into opinion polls, there have been some important works in the field of decision-making systems, such as [15], as well as marketing, especially in consumer behavior, such as in studies developed by [16].

The so-called fuzzy sets attempt to categorize elements not only in terms of pertinence or nonpertinence, as in the case of the classic theory, but also in terms of varied degrees of pertinence. Thus, the fuzzy approach categorizes objects according to a measurement of the similarity between them and the center of a conceptual space, wherein the closer the object is to the center, the more similar it will be and the farther away from the center, the less similarity there is.

Thus, the fuzzy approach categorizes objects according to a measurement of the similarity between them and the center of a conceptual space, wherein the closer the object is to the center, the more similar it will be and the farther away from the center, the less similarity there is.

Having established that, the aim of using fuzzy similarity in text categorization is to define how similar two representative vectors are, where representative vectors can be understood as the set of characteristics that best define the set of the text.

With a basis on the attribution of relevance of the terms in relation to the text, fuzzy systems are anchored on the idea of similarity, allowing the results to offer not only precise/exact classification, but also partial classifications, where each category is attributed a degree of pertinence of relevence in relation to the analyzed text.

A term is considered similar when it is found both in the category index as well as in the text's index. The degrees of equality of the terms are used to determine the degree of similarity between the text index and the category index and, in this way, the text is classified in the category with which it obtains the greatest degree of similarity.

With the use of fuzzy logic in text categorization, one comes closer to finding a solution to the ambiguity problem, since it proposes to treat imprecise situations, offering better results by way of the calculation of pertinence of an element to a set. By way of this technique, it is possible to define just how important and relevant a term is, or isn't, to a given category.

The agglomerative hierarchical method involves partitioning data successively, thereby producing a hierarchical representation of the groupings. This method does not require the number of groupings to be defined. By analyzing the dendogram – a diagram showing the hierarchy in relation to the groupings in the structure – one is able to work out the number of appropriate groupings.

This method demands a distance matrix between the groupings, which is called similarity matrix [6] In order to calculate the distance, a variety of

Authorized licensed use limited to: Universidade de Lisboa Reitoria. Downloaded on September 11, 2009 at 10:45 from IEEE Xplore. Restrictions apply.

methods may be used, the most important of which are [1]: Simple Connection (the distance between two more similar groupings); Complete Connection (the distance between two less similar groupings); Centroid (the distance between two groupings is obtained by their centroids); Average of the Connections (which is the average distance of each grouping); Average of the connection groups (the distance between two groupings is calculated by the union average of both related groups) and Ward (where you find the partitions that minimize the loss associated to each grouping). The Hierarchical method can be subdivided into the Agglomerative and Divisional methods. In this work, the Agglomerative method is used, as illustrated in Figure 1.

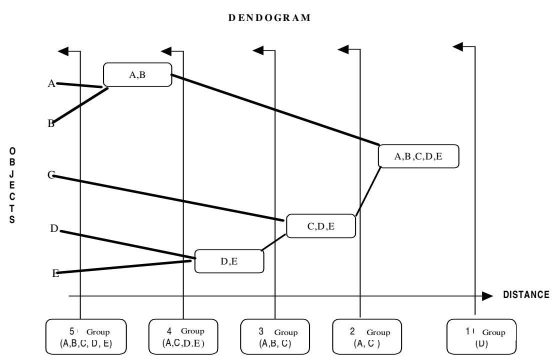

Figure 1. Agglomerative Hierarchical Method.

The Agglomerative method begins with each pattern forming its own grouping and gradually the groups are united into one single grouping. In the beginning, there will be few groupings with a high degree of similarity between their elements but, as the process evolves, these groups will increase and there elements will become dissimilar.

An agglomerative hierarchical algorithm can be described, basically, in the following way:

1. Look for the pair of clusters that are most similar to one another.
2. Create a new cluster that groups the pair selected in step 1.
3. Decrease by 1 the number of remaining clusters.
4. Repeat step 1 until there is only one cluster left.

What differs between various algorithms is the method or strategy used to identify the pairs of clusters that are most similar to one another.

3. Methodology

In the first phase of this work, a technique of text preparation which uses to attempt to remove all that is not meaningful in the text (invalid characters and the removal of stopwords), thereby making the text more streamlined and the categories index more succinct.

The second phase of the methodology adopts the selection of characteristics of the terms in the text with the use of a technique called relative frequency. This defines the importance of given term according to the frequency with which it is found in the text. The more often a term appears in a text, after the removal of the stopwords, the more important the term is in defining it.

The relative frequency proposed by [13] is calculated by way of the formula (1) presented below. This formula normalizes the result, avoiding small documents to be represented by small vectors and large documents by large vectors. With the normalization, all the documents will be represented by vectors of the same size.

$$F_{rel}X = \frac{F_{abs}X}{N} \quad (1)$$

Where:

- F_{real} X= the relative frequency of X;
- F_{abs} X=the absolute frequency of X, the number of times that X appears in the document;
- N = the total number of terms in the text.

Despite being simple, this technique is adopted because [17] shows that the chosen function does not influence the grouping techniques. In fact they are only influenced by the grouping algorithm being used [11].

After selecting the characteristics, a technique to detect important characteristics was employed for which a minimum value of importance, or threshold, was adopted in which the words (characteristics) with an importance (frequency) below this value are simply ignored [14]. The use of this technique is important due to high dimensionality of the space of the characteristics; that is, the large quantity of words that make up a document and need to be treated. It is necessary, therefore, to reduce the space in order to achieve a better classification. The selection of important characteristics aims to aid in this task.

As a way of identifying the similarity between the words in the texts, which is characteristic of the third phase of categorization, fuzzy measures

Authorized licensed use limited to: Universidade de Lisboa Reitoria. Downloaded on September 11, 2009 at 10:45 from IEEE Xplore. Restrictions apply.

were used. The fuzzy similarity measure used here is that iof set theoretic inclusion, as defined by [2],[3], which evaluates the presence of words in both of the compared elements. If the term appears in both elements, the value of (1) is summed to the counter; if not, zero (0) is added. At the end, the degree of similarity is a fuzzy value between 0 and 1, calculated by the average, that is, the total value of the terms counter in common divided by the total number of words in both documents (without counting any repetitions). When the phase of calculating the fuzzy similarity is concluded, a matrix is generated, which indicates the similarity values between the texts. With a basis on this matrix, the algorithms are used to identify the groups of texts, specifying some type of relationship rule. In the fourth phase, the proposed methodology adopts the agglomerative hierarchical method, whose main feature is the non-definition of a number of groupings. Through the dendogram analysis, the number of groupings can be adequately inferred. The most important algorithms that belong to the agglomerative hierarchical method, according to [9] are the following: Cliques, Stars, Connected Components and Strings. For this methodology, we chose to adopt the Star and Cliques algorithms due to the lack of cohesion between the texts [11].

3.1. Star Algorithm

The Star algorithm was given this name precisely because it forms a cluster with a shape that resembles a star; that is, with one central element and others connected to it, representing the tips of a star. In this case, the central element is that one that has a relationship with all the other elements in the star, which are interconnected. The elements on the tips don't necessarily have to be directly related to each other, which represents one of the biggest problems with this algorithm, as they may not be similar. To minimize this problem of lack of similarity among the elements that are on the tips of the star, a similarity threshold should be established. Hence, the solution for elements in opposite tips of the star to not be very dissimilar or distant consists of selecting a higher degree of similarity, seeing as the closer they are to the center, the more similar they will be among themselves, giving more coherence to the group as a whole. The outline of STAR (Figure2) algorithm is as follows.

Step 1: Select one element and join every similar element in the same cluster;

Step 2: Ellements not allocated/classified are considered as cluster seeds (repeat Step 1 for elements not yet allocated).

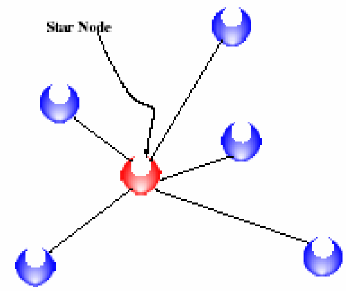

Figure 2. Graph of the STAR Algorithm.

3.2. Clique Algorithm

The Clique algorithm is similar to the Star algorithm, although the elements are only added to a cluster when their degree of similarity is greater than the threshold for all the elements already present in the cluster, and not only in relation to the central element. In this case, the clusters tend to be more cohesive and to have a greater quality, since the elements are closer or more similar to one another. The outline of CLIQUES (Figure3) algorithm is as follows.

Step 1: Select a near object and add him to a new cluster;

Step 2: Find a similar object;

Step 3: If this object is similar to all of the objects in the cluster, add it;

Step 4: Stop criterion: while there is at least one object not allocated, come back to Step 2;

Step 5: come back to Step 1.

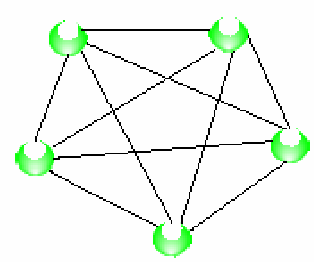

Figure 3. Graph of the CLIQES Algorithm.

3.3. How the Categorizer works

The implemented prototype prepares the text as illustrated in Figure 4, using techniques of term identification, removal of invalid characters and removal of stopwords.

Authorized licensed use limited to: Universidade de Lisboa Reitoria. Downloaded on September 11, 2009 at 10:45 from IEEE Xplore. Restrictions apply.

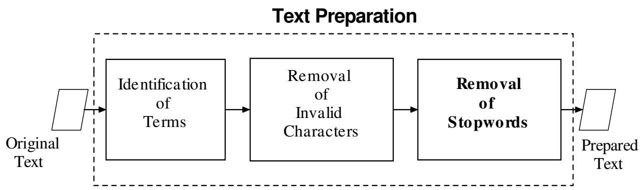

Figure 4. Text Preparation.

In the stage of characteristic selection illustrated in Figure 5, the categorizer calculates the relative frequency between the texts and detects the important characteristics, choosing a threshold in which the words (characteristics) with an importance (frequency) below the chose value are simply ignored.

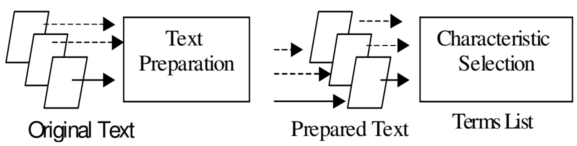

Figure 5. Characteristic Selection Phase of the categories.

Figure 6 shows the matrix that results from the calculation of the fuzzy similarity and the use of the Agglomerative Hierarchical method applying the Stars and Cliques algorithm to identify the groups of text specifying some type of relationship rule, thereby generating new categories.

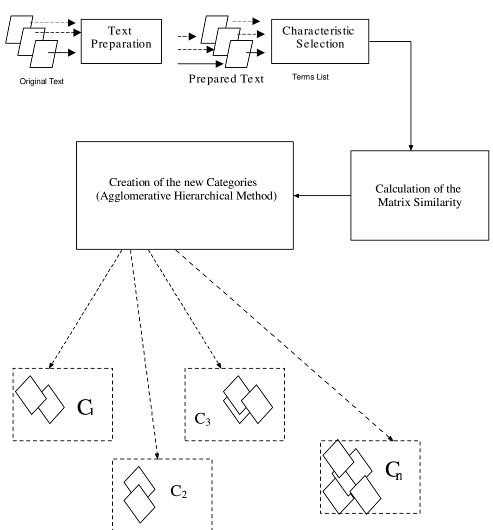

Figure 6. Agglomerative Hierarchical method applying the Stars and Cliques algorithm.

4. Experiments

In this experiment, the TeMario Corpus proposed by [12] was used to conduct the categorizer's texts. This Corpus is made up of 100 texts, which are classified according to Summary and Source Text. In order to conduct the experiments, dotted section are used, as illustrated in Figure 7; in other words, the source text with its origin and title. In the division Source text (with origin and title) there is: a subdivision with texts from two major Brazilian newspapers, Folha de São Paulo and Jornal do Brasil. The texts fall into 5 categories (Special, World, Opinion, Politics and International) and each of these include 20 texts. This choice was made strategically, as it facilitates the analysis of the results after the end of the categorization.

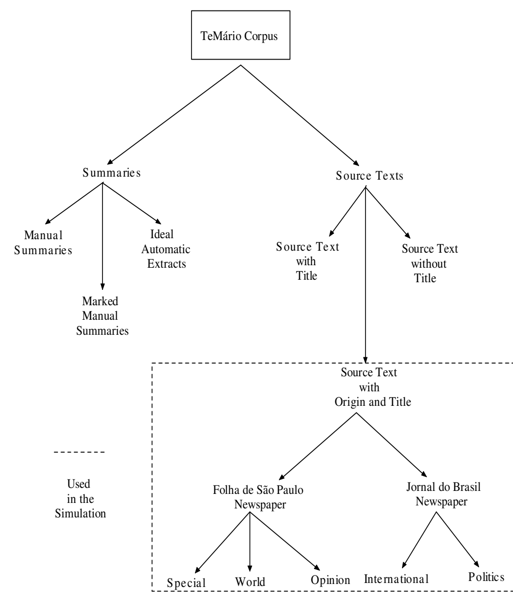

Figure 7. TeMário Corpus used in the simulation of the categorizer.

5. Results

In the simulation, we employed the algorithms Star and Clique in the agglomerative hierarchical method and the TeMario Corpus with 100 texts.

Using the Star algorithm, the prototype created 23 categories, as shown in Table 1. Category C1 was the one that obtained the highest amount of text, with a total of 21. In the TeMario Corpus, these texts were divided as follows: 13 in the Opinion category (on the topic of national politics), 5 in the Special category (4 were on economics and 1

Authorized licensed use limited to: Universidade de Lisboa Reitoria. Downloaded on September 11, 2009 at 10:45 from IEEE Xplore. Restrictions apply.

was about violence), 2 in the World category (1 was on Mexican politics and the other on the global economy) and 1 in the Politics category (national politics).

Categories C_5, C_{10}, C_{11}, C_{17}, C_{18}, C_{19}, C_{20} and C_{23} were categorized with only one single text. In the case of categories C_2, C_{12}, C_{14}, C_{16}, C_{21} and C_{22}, we observed that the texts were grouped into pairs and no coherence was found between the subject matters. In category C_3 we obtained 15 texts, all of which talk about politics (national and international), with the exception of one text on economics.

In C_7 13 texts were grouped together and they were about presidential successions and the threat of war among countries. The prototype grouped into category C_4 a total of 10 texts, which share the topic of economics.

The remaining categories had the following focal point: C_6 (Historical Reports), C_8 (International Politics), C_9 (Consumer), C_{13} (Violence) and C_{15} (attempted crimes). These texts were grouped in a range of 3 to 9, as can be seen in Table 1, where there was coherence among the subject matter.

The use of the Clique algorithm generated 32 categories, as shown in Table 1. Categories C4 and C10 were the ones that obtained the largest number of texts, 9 in each. In the TeMario Corpus, these were the texts included in category C_4 : 6 in the Opinion category (talking about presidential succession), 1 in the World category (on Mexican politics) and 2 in Politics (national politics), while in category C_{10} there were 4 texts in the International category (on politics and economics), 3 in World (2 were historical reports and 1 was about the economy) and 2 in the Special category (historical reports).

Categories C_{13}, C_{25}, C_{27} and C_{31} were categorized with only a single text. The remaining categories had their texts grouped in a range of 2 to 8, as show in Table 1, and they showed coherence in their subject matter.

In this algorithm, when categories were created with only two texts in it, we did not observer incoherence in terms is the smaller amount of categories that were created with only one text (4 in total) when compared to the Star algorithm (with a total of 8).

In terms of processing time, both algorithm showed similar results, with only a small difference of 0.02 seconds attributed to the Clique algorithm. Another fact that deserves mention is the need of both algorithms to establish a Threshold equal to 0.05. This occurred due to the similarity matrix having been obtained with values between 0.02 and 0.07 for both algorithms.

As illustrated in Figure 8, the Clique algorithm, when compared to Stars, did not have a grouping with over 10 texts in any category, which is explained by the fact that the elements are only added to a category when their degree of similarity is greater than the threshold for all the elements already present in the category and not only in relation to the central element.

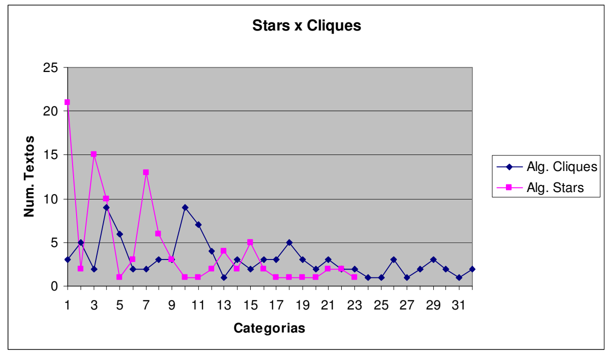

Figure 8. A comparison between the Star and Clique algorithms.

6. Conclusion

This work proposed a text categorization based on the Agglomerative Hierarchical methodology with the use of fuzzy logic. In order to reach this goal, options must be found to solve the problems that are inherent to the treatment of texts. One of the biggest problems in the treatment of textual information is the issue of the concept that exists in the texts and in individual terms.

The ambiguity that occurs naturally in every language, which, many times, is not easily interpreted even by humans, becomes even more difficult to be treated by the computer.

If we take into consideration that there is still a lack of standardized structure in documents and a lack of organization of information, it becomes clear that incorporating techniques of Artificial Intelligence to handle these problems could be a way to find more efficient solutions. When we consider the importance of this field, the search for new technologies and alternatives that lead to better results becomes the greatest motivating force behind current research.

When we analyze the data, we can conclude that the implemented process fulfilled its aim and proved to be efficient. The combination of techniques

Authorized licensed use limited to: Universidade de Lisboa Reitoria. Downloaded on September 11, 2009 at 10:45 from IEEE Xplore. Restrictions apply.

used in each phase of the process was very important for the final aim to be reached.

The technique of relative frequency, employed in the characteristics selection phase, was quite efficient and showed that defining the importance of the terms within the collection and not only within the text is very effective.

The technique of fuzzy similarity (set theoretic inclusion), used in the categorization, presented excellent results. This simple process, based on inference function of fuzzy logic, allows us to define exactly how similar two indexes are.

As for the use of the Star and Clique algorithms used in the agglomerative hierarchical methodology to identify the groups of text by specifying some type of relationship rule, they obtained similar results, but the Clique algorithm showed a slight advantage when compared to the Star algorithm, despite having created a greater number of groupings.

A promising proposal for continued studies is to compare these with other available techniques, such as Genetic Algorithms ore even Support Vector Machines.

7. References

[1] R. Arora, and P..Bangalore, "Textmining: classification & clustering of articles related to sports." Proceedings of the 43rd annual southeast regional conference - Volume 1 ACM-SE 43 Publisher: ACM Press, March 2005. Instituto de Informática, Universidade Federal do Rio Grande do Sul, 2001.

[2] V. Cross, "Fuzzy information retrieval" Journal of Intelligent Information Systems, Boston, v.3, n.1, p.29-56, 1994.

[3] V.Cross, "Similarity or Inference for Assessing Relevance in Information Retrieval", to appear in Proceedings of the Tenth IEEE International Conference on Fuzzy Systems, Melbourne, Australia December, 2001.

[4] U.Fayyad, and R.Uthurusamy, "Data mining and knowledge discovery in databases: Introduction to the special issue." Communications of the Editorial. Data Mining and Knowledge Discovery.

[5] P. Haesun, J.Moongu, J. Ben Rosen "Lower dimensional representation of text data in vector space based information retrieval." Computational information retrieval, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2001.

[6] R.Kannan, S.Vempala and A.Vetta, A. "On clusterings: Good, bad, and spectral." Journal of the ACM, 51(3):497–515, 2004.

[7] King-Lup Liu , Clement Yu , Weiyi Meng , Wensheng Wu , Naphtali Rishe " A Statistical Method for Estimating the Usefulness of Text Databases", IEEE Transactions on Knowledge and Data Engineering, v.14 n.6, p.1422-1437 November 2002.

[8] A.Kontostathis, L. W.Galitsky, M. Pottenger, S. Roy, and D.Phelps, J. "A survey of emerging trend detection in textual data mining. Survey of Text Mining", pages 185-224, 2003.

[9] G.Kowalski, "Information Retrieval Systems: Theory and Implementation." Boston: Kluwer Academic, 1997. 282 p.

[10] G.Klir, J. and T.Folger, A. "Fuzzy sets uncertainty, and information." Englewood Cliffs, NewJersey: Prentice Hall, 1988.

[11] S.Loh, "Abordagem Baseada em Conceitos para Descoberta de Conhecimento em Textos." Porto Alegre: UFRGS, 2001. Requisito Parcial ao Grau de Doutor em Ciência da Computação, Instituto de Informática, Universidade Federal do Rio Grande do Sul, 2001.

[12] T.Pardo, A.S.; L.Rino, H.M. and Martins "TeMário: Um Corpus para Sumarização Automática de Textos." Relatório Técnico número NILC-TR-03-09, Série de Relatórios do Núcleo Interinstitucional Lingüística Computacional, NILC - ICMC-USP, Outubro de 2003.

[13] G.Salton,"Introduction to Modern Information Retrieval."New York: McGraw- Hill, 1983.

[14] H.Schütze, C.Silverstein, "Projections for efficient document clustering" In: SPECIAL INTEREST GROUP ON INFORMATION RETRIEVAL, SIGIR, 1997. Proceeding New York Association for Computing Machinery, 1997. P.74-81.

[15] Sutton Jr S.M. and I.Rouvellou, "Applicability of Categorization Theory to Multidimensional Separation of Concerns." In: Proceedings of the Workshop on Advanced Separation of Concerns, OOPSLA 2001.

[16] S.Tsaur, T. Chang, and C. Yen, "The evaluation of airline service quality bu fuzzy MCDM." Tourism Management, n. 23, p. 107-115 2002.

[17] P.Willet,"Recent Trends In Hierarchic Document Clustering: A Critical Review." Information Processing & Management, [S.l.], v.24, n.5, p.577- 597, 1988.

Authorized licensed use limited to: Universidade de Lisboa Reitoria. Downloaded on September 11, 2009 at 10:45 from IEEE Xplore. Restrictions apply.

Table 1.Categories and texts grouped by the prototype using the algorithms Star and Clique.

Source Text with Origin and Title(*Corpus TeMário*)													
Categorias Criadas		Folha de São Paulo						Jornal do Brasil					
		Opinion		World		Special		International		Politics		Totais Categorias	
Stars	*Cliques*	*Stars*	*Cliques*	*Stars*	*Cliques*	*Stars*	*Cliques*	*Stars*	*Cliques*	*Stars*	*Cliques*	*Stars*	*Cliques*
C_1	C_1	13	3	2	-	5	-	-	-	1	-	21	3
C_2	C_2	2	4	-	-	-	1	-	-	-	-	2	5
C_3	C_3	2	2	2	-	3	-	2	-	6	-	15	2
C_4	C_4	2	6	6	1	1	-	1	-	-	2	10	9
C_5	C_5	1	3	-	2	-	1	-	-	-	-	1	6
C_6	C_6	-	1	-	-	3	-	-	1	-	-	3	2
C_7	C_7	-	1	6	1	1	-	5	-	1	-	13	2
C_8	C_8	-	-	-	-	2	3	2	-	2	-	6	3
C_9	C_9	-	-	-	-	3	3	-	-	-	-	3	3
C_{10}	C_{10}	-	-	-	3	1	2	-	4	-	-	1	9
C_{11}	C_{11}	-	-	-	-	1	3	-	-	-	4	1	7
C_{12}	C_{12}	-	-	1	-	-	4	1	-	-	-	2	4
C_{13}	C_{13}	-	-	1		-	1	3	-	-	-	4	1
C_{14}	C_{14}	-	-	1	1	-	1	-	1	1	-	2	3
C_{15}	C_{15}	-	-	1	-	-	1	1	-	3	1	5	2
C_{16}	C_{16}	-	-	-	2	-	-	1	1	1	-	2	3
C_{17}	C_{17}	-	-	-	3	-	-	1	-	-	-	1	3
C_{18}	C_{18}	-	-	-	1	-	-	1	4	-	-	1	5
C_{19}	C_{19}	-	-	-	3	-	-	1	-	-	-	1	3
C_{20}	C_{20}	-	-	-	1	-	-	1	1	-	-	1	2
C_{21}	C_{21}	-	-	-	1	-	-	-	2	2	-	2	3
C_{22}	C_{22}	-	-	-	1	-	-	-	1	2	-	2	2
C_{23}	C_{23}	-	-	-	-	-	-	-	1	1	1	1	2
-	C_{24}		-		-		-		1		-		1
-	C_{25}		-		-		-		1		-		1
-	C_{26}		-		-		-		1		2		3
-	C_{27}		-		-		-		1		-		1
-	C_{28}		-		-		-		-		2		2
-	C_{29}		-		-		-		-		3		3
-	C_{30}		-		-		-		-		2		2
-	C_{31}		-		-		-		-		1		1
-	C_{32}		-		-		-		-		2		2
Totais		20	20	20	20	20	20	20	20	20	20	100	100

Authorized licensed use limited to: Universidade de Lisboa Reitoria. Downloaded on September 11, 2009 at 10:45 from IEEE Xplore. Restrictions apply.