Document Comparison with a Weighted Topic Hierarchy

A. Gelbukh, G. Sidorov, and A. Guzmán-Arenas

Natural Language Laboratory,
Center for Computing Research (CIC),
National Polytechnic Institute (IPN), Mexico City
{gelbukh, sidorov, aguzman}@pollux.cic.ipn.mx

Abstract

A method of document comparison based on a hierarchical dictionary of topics (concepts) is described. The hierarchical links in the dictionary are supplied with the weights that are used for detecting the main topics of a document and for determining the similarity between two documents. The method allows for the comparison of documents that do not share any words literally but do share concepts, including comparison of documents in different languages. Also, the method allows for comparison with respect to a specific "aspect," i.e., a specific topic of interest (with its respective subtopics). A system Classifier using the discussed method for document classification and information retrieval is discussed.

1. Introduction*

In this article, a document comparison method based on document classification is discussed. The task of document classification can be examined from different points of view [1], [6], [8], [10]. We consider it as assignment of one or several topics to the document. For example, some documents are about *health*, and some about *politics*. Accordingly, we consider document comparison with respect to such a classification: two documents are similar if they share their principal topics.

In some existing systems, such as [9], [12], the contents of the document is characterized by the words frequently used in the document, with no external dictionaries being used. In our work, the documents are related to the entries of a pre-determined dictionary of concepts organized in a hierarchical structure. The dictionary, though, is large, so that statistical methods can be applied to its entries.

* The work done under partial support of DEPI-IPN, CONACyT grant 26424-A, REDII-CONACyT, and COFAA-IPN, Mexico.

In our approach to document classification and comparison, a document is associated with many topics rather than only one, principal, topic. More precisely, a document is characterized by a vector of topic weights r^i representing a measure of correspondence of the document to each of the available topics. This still allows for a more traditional view on classification: the topic(s) with the best value of this measure is the principal topic(s) of the document.

On Figure 1, a screen shot of our program, Classifier, with a histogram r^i of the topics for a Spanish document is shown.

Concept hierarchies have been extensively used in information retrieval and recently in text mining [5], [11]. In [3], [4] it was proposed to use a hierarchical dictionary for determining the main themes of a document. In this paper, we discuss the use of the weights r^i for document comparison.

First, the dictionary structure is presented. Then, the algorithm for calculation of the topic weights r^i is described; we also touch upon the issue of calculation of the link weights in the dictionary. Finally, the algorithm of document comparison is discussed.

2. Weighted topic hierarchy

The dictionary consists of two major parts: vocabulary and the hierarchical structure. The vocabulary includes syntagmatic units, i.e., individual words like *Italy* or word combinations like *the United States of America*; we will call any such unit a *keyword*. The hierarchical structure represents semantic units, i.e., concepts, or topics. It is a tree or, more generally, a directed acyclic graph, which represents the concepts by grouping together the words or other concepts. For example, a concept *Europe* includes, among others, the word *Europe* and the concepts *Western Europe*, *Eastern Europe*, *Schengen states*, etc.

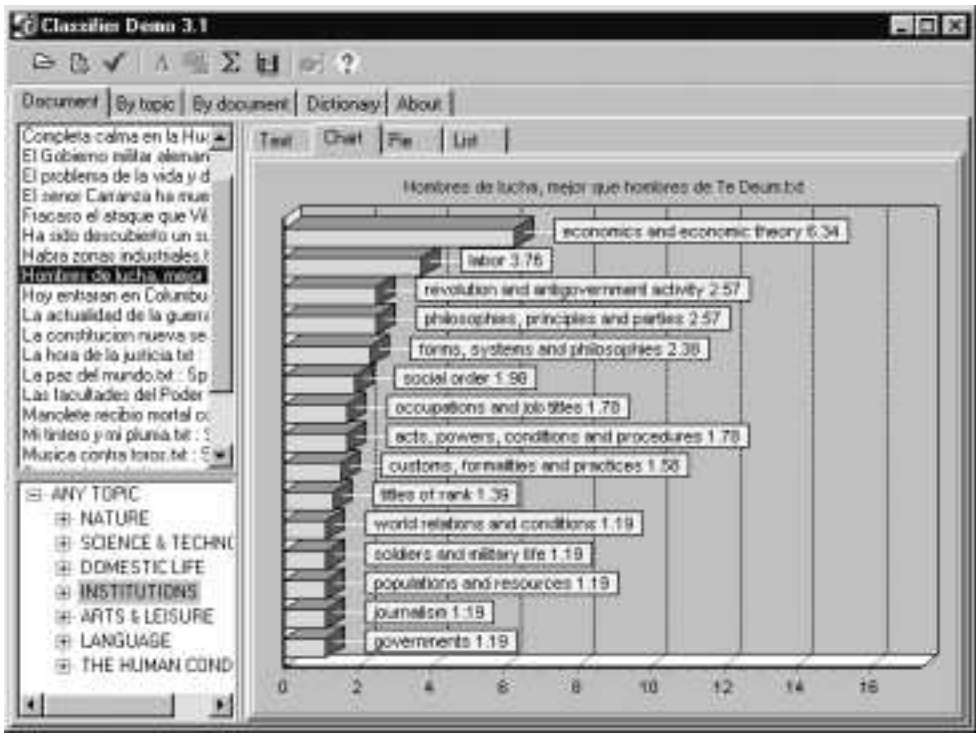

Figure 1. Topic histogram for a document in Spanish.

Figure 2 shows an example of a dictionary entry. The hierarchy of non-terminal concepts is shown in the left side of the picture; the keywords immediately belonging to the selected topic are shown in the right side.

Since our approach is language independent, words of different languages can be mixed in the dictionary. Consequently, depending on the settings chosen by the user, the system can either autodetect the document language and use only the words of the detected language, or use the words of all available languages. As the bottom right part of Figure 2 shows, so far our dictionary is implemented in English, French, and Spanish.

The links in the hierarchy have different strength expressed with the weights of the links. These weights roughly correspond to the probability for the word in a particular context to be really related to the given topic. For example, the word *Italy* or the concept (group) *Schengen states* in practically any context belong to the topic *Europe*; thus, the weight of this link is 1. On the other hand, the word *London* can refer to a city in England or, with much less probability, in Canada; consequently, the weight of the link between *London* and *England* is, say, 0.9. The link between *English* and *England* is very weak because English language is frequently used with no relation to England.

Assigning the weights to the links is not a trivial task, but here we can not deep into details. In short, the weight w_j^i of the link between a node *j* and its parent node *i* characterizes the mean relevance of the documents containing this word for the given topic.

For terminal nodes, a simplified way of automatic assignment of the weights of their links to their parent concepts consists in adopting the inverse proportion to the frequency of the word:

$$w_j^i = \frac{1}{\sum_{k \in \mathbf{D}} n_k^i}$$

independently of the parent topic *i*. Here n_k^i is the number of occurrences of the terminal node *j* in the document *k*, and summation is done by the documents of a training corpus **D**. For example, the articles *a* and *the* have a (nearly) zero weight for any topic, while the word *carburetor* has a high weight in any topic in which it is included.

As to the links between non-terminal concepts, we will not discuss here the issue of assignment of their weights. Since for a shallow hierarchy the number of such links is not very large, the weights can be assigned manually or just considered being all equal to 1.

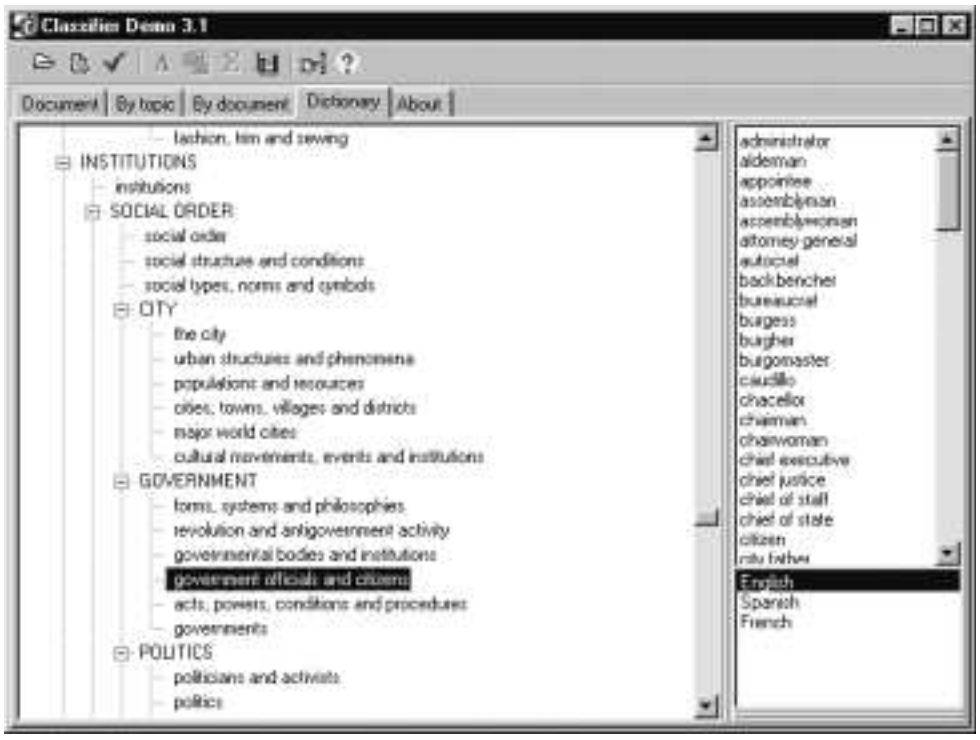

Figure 2. Hierarchical dictionary used by the system.

3. Calculation of topic weights

The algorithm of application of the dictionary for detection of the topic weights is, in the first approximation, very simple. To determine the weights r^i of the topics (nodes) i for a given document, the following two passes are performed:

1. For each terminal node i of the hierarchy, i.e., each keyword (a single word or a word combination), its frequency in the text is determined; the topic weight r^i for this node is equal to its frequency.
2. The frequencies are propagated recursively by the links in the hierarchy:

$$r^i = \sum_j w_j^i r^j \ .$$

Here summation is performed by the dependent nodes of the given node; w_j^i is the weight of the link between the current node i and the dependent node j.

Note that such an algorithm leads to very high weights of the top nodes of the hierarchy: all the documents prove to have *objects* and *actions* as their principal topics. Handling this effect in the application in which it presents a problem goes beyond the scope of this article. This effect, however, does not present any problem in a shallow or one-level hierarchy.

The set of topics can be restricted by the user; such a restriction is a part of the user's query. In the simplest case, the search query consists in selecting a subtree of the topic hierarchy by selecting a desirable top node. Only the topics below this node will participate in the calculations.

One more screen shot of the Classifier program is shown on Figure 3. The words and topics found in the selected document (with Spanish title "Hombres de lucha, mejor…") for the selected topic "Institutions" are presented. The words are shown with their frequencies in the document, and the non-terminal topics with their calculated weights for this document.

4. Document comparison

Thus, we define the document image as a vector of topic weights (r^i). This vector includes all nodes of the hierarchy. As a variant of our approach, this vector can include only non-terminal nodes, i.e., groups of keywords; this greatly decreases memory requirements and increases the efficiency of the algorithm.

For the purposes of comparison, in most cases, the user is not interested in the absolute amount of information conveyed by a document, i.e., the total number of words in the document that are related to a specific topic. Instead,

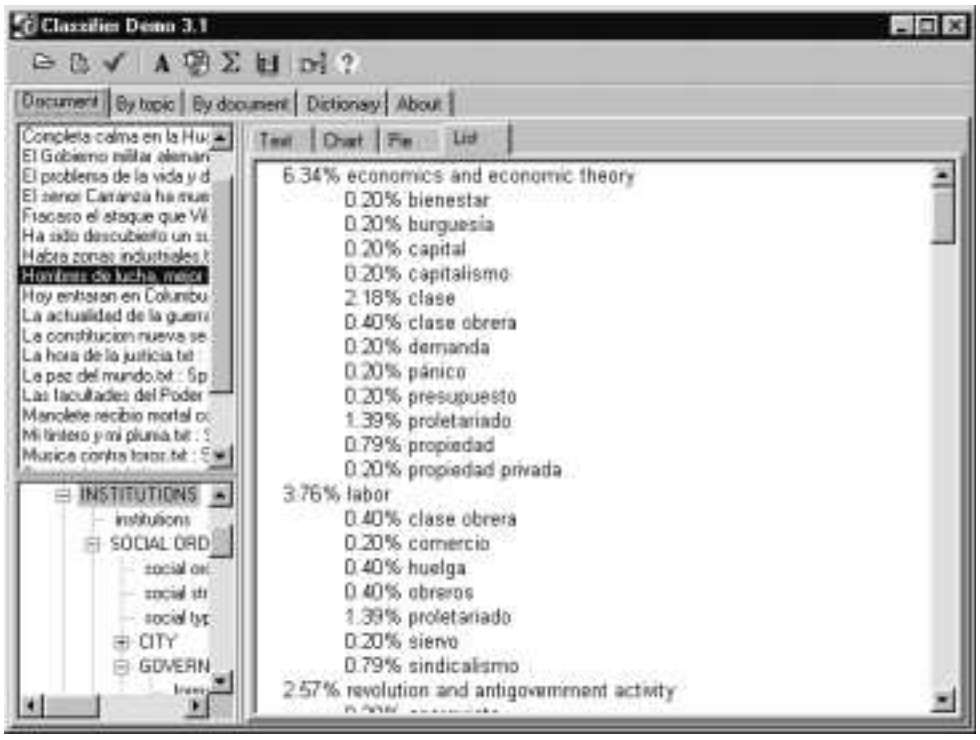

Figure 3. Counting keywords for a Spanish document.

the user is interested in the theme to which the document is devoted, i.e., the density of the specific keywords. In this case we normalize the document image by dividing each coordinate r^i by the total number of words in the document. Thus, the size of the documents does not affect the results of comparison.

The distance between the two documents D_1 and D_2 can be defined now in terms of these vectors. There are several possible ways to determine the distance between two vectors, depending on the needs of the user. The simplest way is a weighted combination of absolute differences of coordinates:

$$\left|D_1 - D_2\right| = \sum_i \alpha_i \left|r_1^j - r_2^i\right|,$$

where summation is performed by all nodes of the hierarchy.

Here α_i are the importance coefficients of the nodes of the hierarchy. In the ideal case they would reflect the user's profile: the user assigns greater coefficients to the most important topics. However, in practice most of them have to be predefined. In our system, they are assigned according to the following rules: the coefficients of individual keywords are much less than those of any group (non-terminal node), or even are zeroes as it was discussed above; the coefficients of the lowest-level non-terminal nodes are maximal; and the coefficients of the top-level nodes are the less the higher the level.

Effectively, the comparison is done by the low-level groups of keywords. On the one hand, this makes it possible for two documents to be very similar even if they do not have any common words literally but do share a common topic. On the other hand, the documents that do share keywords are still slightly closer than those that only share topics are.

An interesting application of the method is classification of the documents by similarity with respect to a given topic. Clearly, a document mentioning "the use of animals for military purposes" and the document mentioning "feeding of animals" are similar (both mention *animals*) from the point of view of a biologist, while from the point of view of a military man they are very different. This is handled by selecting the "aspect" of comparison – a subtree of the topic hierarchy, so that the document images contain only the selected topics.

5. Conclusions and future work

We have discussed a method of document comparison based on the use of a weighted hierarchy of topics (concepts). The method has the following advantages:

1. The documents that do not share any words literally still can be identified as similar ones if they do share common topics.

2. The comparison can be done taking into account the user profile, or the "aspect" – a subset (subtree) of topics that are of interest for the user.

The need in a large dictionary is a disadvantage of the method. However, the method has proved to be insensitive to a rather low quality of the dictionary. For example, in our experiments we used a French dictionary that was an automatic translation of the English one. We applied our algorithm to a set of English documents and the corresponding set of their manual French translations; the difference in the results was insignificant. Also, the documents representing the same text in different languages were reported by the algorithm as very similar.

Though generally the results obtained in our experiments showed good accordance with the opinion of human experts, we have encountered some problems with using our method. Most of such problems are related with lexical ambiguity of different types, such as *well* (noun versus adverb) or *bill* (five different meanings as a noun) [7]. In the future, we plan to apply a part of speech tagger to resolve the ambiguity of the first type, and implement an algorithm making use of different senses like $bill_1$, $bill_2$ manual marked up in the dictionary; such an algorithm can be thesaurus-based [2] or statistical.

Another direction of improvement of the algorithm is taking into account the anaphoric relationships in the text. For example, the pronouns and zero subjects (in Spanish) could be replaced with the corresponding nouns.

References

[1] Cohen, W., Singer, Y.: Context-sensitive Learning Methods for Text Categorization. In: SIGIR'96 (1996)

[2] Gelbukh, A.: Using a Semantic Network for Lexical and Syntactic Disambiguation. In: Proceedings of Simposium Internacional de Computación: Nuevas Aplicaciones e Innovaciones Tecnológicas en Computación. Mexico (1997)

[3] Guzmán-Arenas, A.: Hallando los Temas Principales en un Artículo en Español. Soluciones Avanzadas **5 (45)** (1997) 58, **5 (49)** (1997) 66

[4] Guzmán-Arenas, A.: Finding the Main Themes in a Spanish Document. Journal Expert Systems with Applications **14 (1, 2)** (1998) 139-148

[5] Feldman R., I. Dagan: Knowledge Discovery in Textual Databases (KDT), In Proc. of Intern. Symposium "KDD-95", pages 112-117, Montreal, Canada (1995)

[6] Jacob, E. K.: Cognition and Classification: A Crossdisciplinary Approach to a Philosophy of Classification. (Abstract.) In: Maxian, B. (ed.): ASIS '94: Proceedings of the 57th ASIS Annual Meeting. Medford, NJ: Learned Information (1994) 82

[7] Krowetz, B.: Homonymy and Polysemy in Information Retrieval. In: Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (1997) 72-79

[8] Lewis, D. D., Ringuette, M.: A Comparison of Two Learning Algorithms for Text Categorization. In: Third Annual Symposium on Document Analysis and Information Retrieval (1994) 81-93

[9] Meunier, Jean Guy, Lakhdar Remaki, and Dominic Forest: Use of classifiers in computer-assisted reading and analysis of texts (CARAT). To be published. (1999)

[10] Riloff, E., Shepherd, J.: A Corpus Based Approach for Building Semantic Lexicons. In: Proceedings of the Second Conference on Empirical Methods in Natural Language Processing (EMNLP-2) (1997)

[11] Srikant R., R. Agrawal: Mining Sequential Patterns: Generalizations and Performance Improvements. In Proc. of the 5th International Conference on Extending Database Technology (EDBT), Avignon, France, March (1996)

[12] TextAnalyst system. http://www.analyst.ru (in Russian), http://www.megaputer.com (1998)