Discovering Unexpected Information for Technology Watch

François Jacquenet and Christine Largeron

Université Jean Monnet de Saint-Etienne
EURISE
23 rue du docteur Paul Michelon
42023 Saint-Etienne Cedex 2
{Francois.Jacquenet,Christine.Largeron}@univ-st-etienne.fr

Abstract. The purpose of technology watch is to gather, process and integrate the scientific and technical information that is useful to economic players. In this article, we propose to use text mining techniques to automate processing of data found in scientific text databases. The watch activity introduces an unusual difficulty compared with conventional areas of application for text mining techniques since, instead of searching for frequent knowledge hidden in the texts, the target is unexpected knowledge. As a result, the usual measures used for knowledge discovery have to be revised. For that purpose, we have developed the UnexpectedMiner system using new measures for to estimate the unexpectedness of a document. Our system is evaluated using a base that contains articles relating to the field of machine learning.

1 Introduction

In recent years, business sectors have become more and more aware of the importance of mastering strategic information. Businesses are nevertheless increasingly submerged with information. They find it very difficult to draw out the strategic data needed to anticipate markets, take decisions and interact with their social and economic environment. This has led to the emergence of business intelligence [5, 6, 14] that can be defined as the set of actions involved in retrieving, processing, disseminating and protecting legally obtained information that is useful to economic players. When the data analyzed is scientific and technical, the more specific term used is technology watch, meaning the monitoring of patents and scientific literature (articles, theses, etc.).

A watch process can be broken down into four main phases: a needs audit, data collection, processing of the data collected and integration and dissemination of the results. Our focus in this article is mainly on the third phase. For the purpose of automatically processing the data collected, data mining techniques are attractive and seems to be particularly suitable considering that most of the data is available in digital format.

Data mining has grown rapidly since the mid 90's with the development of powerful new algorithms that enable large volumes of business data to be pro-

J.-F. Boulicaut et al. (Eds.): PKDD 2004, LNAI 3202, pp. 219–230, 2004.

cessed [3]. When the data considered comes in the form of texts, whether structured or not, the term used is text mining [8]. By analogy with data mining, text mining, introduced in 1995 by Ronan Feldman [4], is defined by Sebastiani [19] as the set of tasks designed to extract the potentially useful information, by analysis of large quantities of texts and detection of frequent patterns. In fact text mining is already a wide area of research that provides useful techniques that can be used in the context of technology watch. Losiewicz et al. [12] for example show that clustering techniques, automatic summaries, information extraction can be of great help for business leaders. Zhu and Porter [22, 21] show how bibliometrics can be used to detect technology opportunities from competitors information found in electronic documents. Another use of text mining techniques for technology watch was the works of B. Lent et al. in [10] that tried to find new trends from an IBM patent database using sequential pattern mining algorithms [1]. The idea was to observe along the time, sequences of words that were not frequent in patents at a particular period and that became frequent later. In a similar way, but in the the Topic Detection Tracking[1] framework, Rajaraman et al. [17] proposed to discover trends from a stream of text documents using neural networks. In these cases, text mining techniques are mainly used to help managers dealing with large amount of data in order to find out frequent useful information or discover some related works linked with their main concerns. Nevertheless, one important goal of technology watch and more generally business intelligence is to detect new, unexpected and hence generally infrequent information. Thus, the algorithms for extracting frequent patterns that are commonly used for data mining purposes are inappropriate to this area. Indeed, as the name implies, these tools are tailored for information that occurs frequently in a database. This is no doubt one of the main reasons why the software packages marketed so far fail to fulfill knowledge managers needs adequately.

From that assessment, some researchers have tried to focus on what they called, rare events, unexpected information, or emerging topics, etc, depending on the papers. For example, Bun et al. [2] proposed a system to detect emerging topics. From a set of Web sites, their system is able to detect changes in the sites and then scans the words that appear in the changes in order to find emerging topics. Thus, the system is not able to detect unexpected information from Web sites they visit for the first time. Matsumura et al. [13] designed a system to discover emerging topics between Web communities. Based on the KeyGraph algorithm [15], their system is able to analyze and visualize co-citations between Web pages. Communities, each having members with common interests are obtained as graph-based clusters, and an emerging topic is detected as a Web page relevant to multiple communities. However this system is not relevant for corpora of documents that are not necessarily organized as communities.

[1] http://www.nist.gov/speech/tests/tdt

More recently many researches have focused on novelty detection, more specifically in the context of a particular track of the TREC challenges[2]. Many systems have been design in the context of that challenge, nevertheless, they only deal with corpora of sentences and not corpora of texts. Moreover, for most systems, the users need to provide examples of novel sentences. Then the system uses some similarity functions in order to compare new sentences with known novel sentences.

WebCompare, developed by Liu et al. [11] proposed the users to find unexpected information from competitors' Web sites. Hence, a user of the system has to give a set of URLs of Web pages and the URL of his Web site. Then WebCompare is able to find the pages that contain unexpected information with respect to the user's Web site. The unexpectedness of a Web page is evaluated given a measure based on the TF.IDF paradigm. In fact, WebCompare is probably the most related work to our concerns.

Next section presents the global architecture of the system we designed, and called UnexpectedMiner, to automate the Technology Watch process. Section 3 presents the various measures we proposed to mine unexpected information in texts. Section 4 presents some experiments we made to show the efficiency of each measure with respect to each other before we conclude with some future works.

2 The UnexpectedMiner System

In the technology watch area, we have developed the UnexpectedMiner system aimed at extracting documents that are relevant to the knowledge manager from a corpus of documents inasmuch as they deal with topics that were unexpected and previously unknown to the manager. In addition, the system must specifically treat the knowledge manager's request without relying to any large extent on her or his participation. Finally, an important feature we wanted to build into our system is versatility, i.e. a system that is not dedicated to a particular field or topic.

Keeping in mind those objectives, we propose a system made up of several modules as illustrated in Figure 1.

2.1 Pre-processing of Data

In the first phase, the technology watch manager specifies the needs by producing a number of reference documents. In the remainder of this article, this set of documents shall be designated by R while $|R|$ refers to their number. In practice, between ten and twenty documents should be enough to target the scope of interest for the technology watch. The system must then review new documents

[2] The track "novelty detection" at TREC conferences has appear for the first time at the TREC 2002 conference. Papers presented to this conference and next may be found at http://trec.nist.gov

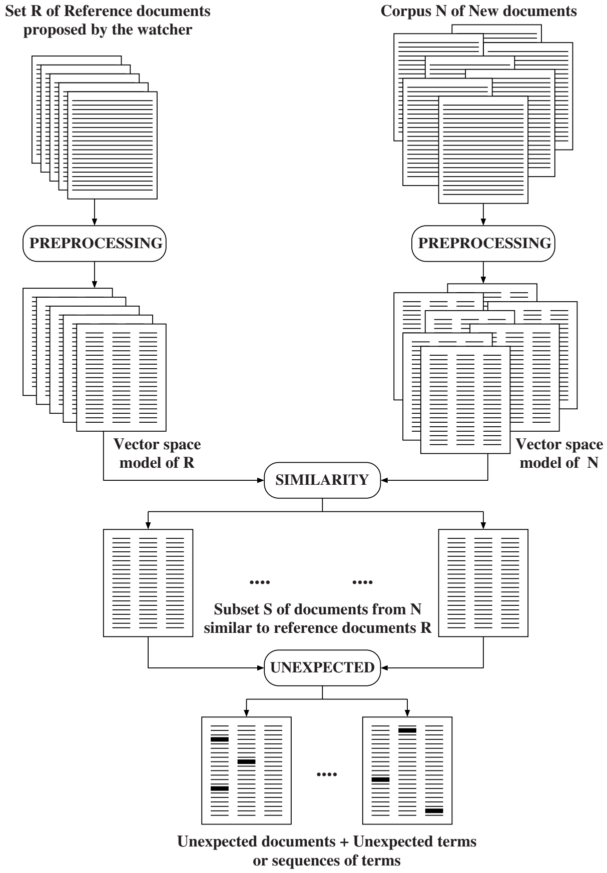
Set R of Reference documents
proposed by the watcher
Corpus N of New documents
PREPROCESSING
PREPROCESSING
Vector space
model of R
Vector space
model of N
SIMILARITY
....
....
Subset S of documents from N
similar to reference documents R
UNEXPECTED
....
Unexpected documents + Unexpected terms
or sequences of terms

Fig. 1. UnexpectedMiner system architecture

in various corpora made available and retrieve innovative information. This set of new documents shall be referred to below as N and $|N|$ is its cardinal.

Sets R and N then undergo the pre-processing phase. The module designed for that purpose includes a number of conventional processing steps such as removing irrelevant elements and stop words from the documents (logos, url addresses, tags, etc.) and carrying out a morphological analysis of the words in the phrases extracted. Finally, each document is classically represented in vector form. The document d_j is thus considered to be a set of indexed terms t_i where each indexed term is in fact a word in document d_j. An index written out as $T = \{t_1, t_2, ..., t_m\}$ lists all the terms encountered in the documents. Each document is thus represented by a weights vector $\boldsymbol{d}_j = (w_{1,j}, w_{2,j}, ..., w_{m,j})$ where $w_{i,j}$ represents the weight of term t_i in document d_j. If the term t_i does not appear in the document d_j , then $w_{i,j} = 0$. To compute the frequency of a term in a document, we use the TF.IDF formula [18]. TF (Term Frequency) is the relative frequency of term t_i in a document d_j defined by:

$$tf_{i,j} = \frac{f_{i,j}}{max_l f_{l,j}}$$

where $f_{i,j}$ is the frequency of term t_i in document d_j. The more frequent the term t_i in document d_j, the higher the $tf_{i,j}$.

IDF (Inverse Document Frequency) measures the discriminatory power of term t_i defined by:

$$idf_i = \log_2 \frac{Nd}{n_i} + 1$$

where Nd is the number of documents processed and n_i is the number of documents that contain the term t_i. The less frequent the term t_i in the set of documents, the higher is idf_i. In practice, IDF is simply calculated by:

$$idf_i = \log \frac{Nd}{n_i}$$

The weight $w_{i,j}$ of a term t_i in a document d_j is found by combining the two previous criteria:

$$w_{i,j} = tf_{i,j} \times idf_i$$

The more frequent the term t_i is in document d_j and the less frequent it is in the other documents, the higher the weight $w_{i,j}$.

2.2 Similar Document Retrieval

The goal of the second module is to extract from database N new documents which are the most similar to the reference documents R provided by the knowledge manager. The similarity s_{jk} between a new document $d_j \in N$ and a reference document $d_k \in R$ is equal to the cosine measure, commonly used in information retrieval systems. It is equal to the cosine of the angle between the vectors that represent these documents:

$$s_{jk} = \frac{\boldsymbol{d}_j \bullet \boldsymbol{d}_k}{|\boldsymbol{j}| \times |\boldsymbol{k}|}$$

where

$$\boldsymbol{d}_j \bullet \boldsymbol{d}_k = \sum_i w_{i,j} \times w_{i,k}$$

$$|\boldsymbol{j}| = \sqrt{\sum_{i=1,m} w_{i,j}^2}$$

The mean similarity s_j of a new document $d_j \in N$ with the set of reference documents R is equal to:

$$s_j = \frac{1}{|R|} \sum_{k=1}^{|R|} s_{jk}$$

After classifying the mean similarity of new documents in the descending order, a subset S is extracted from N. This is the set of the new documents that are the most similar to those supplied as reference documents by the knowledge manager.

2.3 Unexpected Information Retrieval

The core of the UnexpectedMiner system is the unexpected information retrieval module. The purpose of this module is to find the documents d_j of S that contain unexpected information with respect to $R \cup S - \{d_j\}$. Indeed, a document d_j is highly unexpected if, while similar to documents of $R \cup S - \{d_j\}$, it contains information that is found neither in any other document of S nor in any document of R. This module is described in detail in the section below.

3 Measures of the Unexpectedness of a Document

Five measures are proposed for assessing the unexpectedness of a document.

3.1 Measure 1

The first measure is derived directly from the criterion proposed by Liu et al. [11] for discovering unexpected pages on a Web site. It is defined by:

$$M1(d_j) = \frac{\sum_{i=1}^{m} U^1_{i,j,c}}{m}$$

with

$$U^1_{i,j,c} = \begin{cases} 1 - \frac{tf_{i,c}}{tf_{i,j}} & if \;\; tf_{i,c}/tf_{i,j} \leq 1 \\ 0 & else \end{cases}$$

where d_j is a document in S and $D_c = R \cup S - \{d_j\}$ is the document obtained by combining all the reference documents in R with the similar documents except d_j.

The main drawback with this measure is that it gives the same value for both the terms t_i and $t_{i'}$ that occur with different frequencies in a new document $d_j \in S$ once these terms do not occur in D_c (in other words, in the other documents in $R \cup S - \{d_j\}$). Now it would be desirable to get an unexpectedness value $U^1_{i,j,c}$ for t_i greater than the value $U^1_{i',j,c}$ found for $t_{i'}$ when t_i is more frequent than $t_{i'}$ in d_j. This is particularly the case when t_i pertains to a word that has never been encountered before whereas $t_{i'}$ is a misspelled word. This consideration led us to propose and experiment other measures for assessing the unexpectedness of a document.

3.2 Measure 2

With the second measure, the unexpectedness of a term t_i in a document $d_j \in S$ in relation to all of the other documents D_c is defined by:

$$U^2_{i,j,c} = \begin{cases} tf_{i,j} - tf_{i,c} \; if \; tf_{i,j} - tf_{i,c} \geq 0 \\ 0 \qquad\qquad\quad else \end{cases}$$

Just as in with $M1$, the unexpectedness of a document d_j is equal to the mean of the unexpectedness values associated with the terms representing d_j:

$$M2(d_j) = \frac{\sum_{i=1}^{m} U^2_{i,j,c}}{m}$$

This second measure gets rid of the drawback in the first. Indeed, if we go back to the previous example, if the term t_i occurs more frequently than $t_{i'}$ in document d_j and that neither appear in D_c, then:

$$U^2_{i,j,c} > U^2_{i',j,c}$$

3.3 Measure 3

With the previous measures, only the terms were considered. In the technology watch area and information retrieval likewise, it is often the association of several terms, e.g. “data mining”, which is operative. On that basis, we decided to represent each document by terms and sequences of consecutive terms.

We then used an algorithm for sequential pattern mining based on the algorithm from Agrawal [1] for extracting sets of frequent sequences of consecutive terms in the textual data.

For terms and sequences of consecutive terms, we defined a third measure that is an adaptation of M2 in which:

$$tf_{i,j} = \frac{f_{i,j}}{max_l f'_{l,j}}$$

where $max_l f'_{l,j}$ is the maximum frequency observed in the terms and sequences of consecutive terms.

However, neither of these three measures takes into account the discriminatory power of a term as expressed by IDF. This inadequacy can partially be overcome by combining all of the documents. Nonetheless, it seemed to us valuable to design unexpectedness measures that make direct use of this information, as with the two methods described below.

3.4 Measure 4

The fourth measure makes direct use of the discriminatory power idf_i, of the term t_i by evaluating the unexpectedness of a document d_j through the sum of the weights $w_{i,j}$ of the terms t_i that represent it (remember $w_{i,j} = tf_{i,j} \times idf_i$):

$$M4(d_j) = \sum_{i=1}^{m} w_{i,j}$$

With this measure, two documents d_j and d'_j may nonetheless have same unexpectedness value in spite of the fact that the weights of the terms representing the first document are equal while those for the second document are very different.

3.5 Measure 5

To overcome the limitation of $M4$, the fifth measure proposed assigns the highest weight in a document's vector of representation as that document's unexpectedness value:

$$M5(d_j) = \max_l w_{l,j}$$

Tests were performed to evaluate this system and compare the various measures. These are described in the next section.

4 Experiments

4.1 Corpus and Evaluation Criteria Used

The reference set R comprises 18 scientific articles in English dealing with Machine Learning none of which deal with some particular topics such as "Support Vector Machines, Affective Computing, Reinforcement Learning,... etc". The N base comprises 57 new documents, 17 of which are considered by the knowledge manager to be similar to the reference documents. Among these 17, 14 deal with topics that the latter considers unexpected.

For the purposes of evaluating UnexpectedMiner, we used the *precision* and *recall* criteria defined by J.A. Swets [20]. In our system, *precision* measures the percentage of documents extracted by the system that are truly unexpected. *Recall* measures the percentage of unexpected documents found in the N document corpus by the system. These are conventional criteria in the area of information retrieval and we shall not consider them in any further detail here.

4.2 Evaluation of the Five Measures

Because the main contribution of this work is to define new means for measuring the unexpectedness of a document, the module that implements those measures was first evaluated independently from the module for the extraction of similar documents, and then in combination with all the other modules.

We therefore first restricted the S base to the 17 new documents considered by the knowledge manager to be similar to the reference documents R. The results obtained in terms of recall and precision using the five measures defined above are provided in figures 2 to 6 where the number of documents extracted by the system is given on the x-axis

Whereas the N base comprises by far a majority of documents that deal with unexpected topics (14 documents out of 17), only the $M1$ measure is unable to return them first since the precision value is 0% considering just the two first documents extracted (figure 2) while this value is 100% for the other measures (figures 3 to 6).

The results achieved with the $M2$ (figure 3) and $M3$ measures (figure 4) are more satisfactory. But it is measures $M4$ and $M5$ that first return the most documents that deal with unexpected topics. Indeed, precision continues to be equal to 100% even considering up to six documents for $M4$ (figure 5) and up to seven for $M5$ (figure 6).

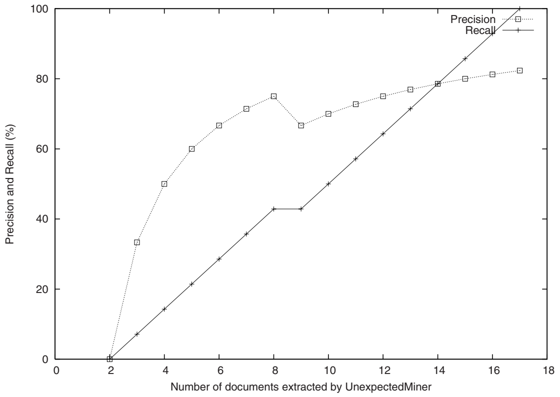

Fig. 2. Precision and recall: measure 1

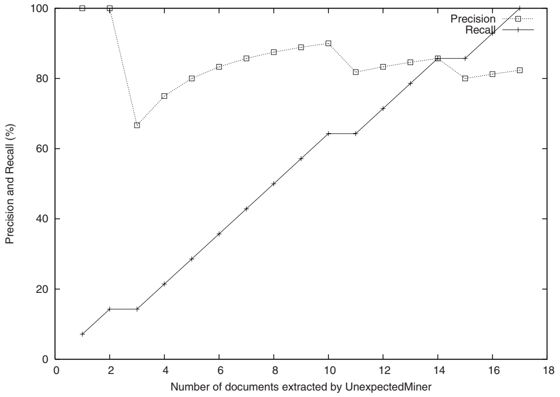

Fig. 3. Precision and recall: measure 2

4.3 Evaluation of the Whole System

We finally evaluated the complete system in respect of the N base that comprised 57 new documents. Among the 15 first documents considered similar to the reference documents by the system, only 9 actually were, i.e., a precision rate of 60 % and a recall rate of 52.9 %. Among those 9 documents, 7 dealt with unexpected topics. Under this second experiment, only measure $M1$ is unable to first extract a document that deals with an unexpected topic: precision is 0% whereas it is 100% for $M2$, $M3$, $M4$ and $M5$ that also correctly identify the same unexpected document. It is noteworthy that unexpected documents are not detected as well by the $M1$ measure since the recall is 100% only when the number of documents extracted is equal to the number of documents supplied to the system. The performance of $M2$ and $M4$ are more or less comparable but once again it is measure $M5$ which first extracts the documents relating to unexpected topics. However, this measure rather often assigns the same value

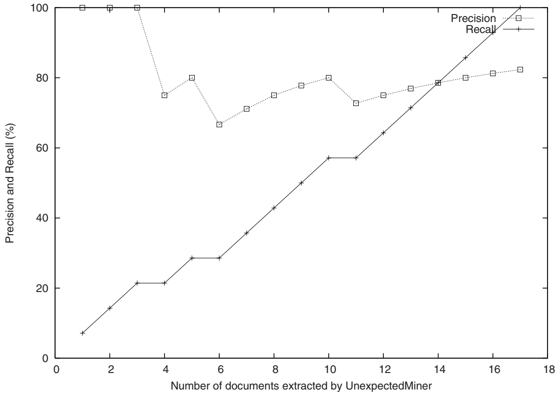

Fig. 4. Precision and recall: measure 3

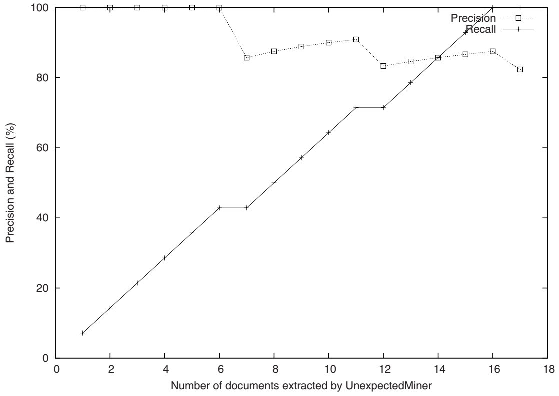

Fig. 5. Precision and recall: measure 4

to several documents. Finally, although the results achieved with $M3$ are somewhat less satisfactory, there is a match between the sequences of unexpected words that it returns and those being sought, i.e. “support vector machine” or “reinforcement learning”. It is worth noting that a useful feature of the UnexpectedMiner system in this respect is that it indicates the words or sequences of words that most contributed to making a documents submitted to the system an unexpected document.

5 Conclusion

We have developed a watch system that is designed to extract relevant documents from a text corpus. Documents are relevant when they deal with topics that were unexpected and previously unknown to the knowledge manager. Several measures of the unexpectedness of a document were proposed and compared.

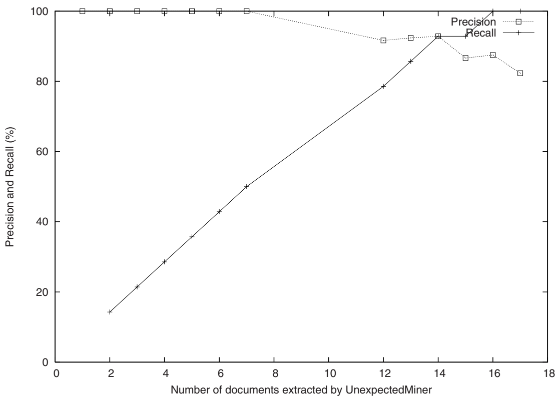

Fig. 6. Precision et recall: measure 5

Although the results obtained are encouraging, we think they can still be improved. To find the information that is interesting to the knowledge manager, it therefore appears essential to properly target the set of documents in which this information is to be retrieved. Thus, it would be worthwhile studying other measures of similarity [9] in the module that extract similar documents to the set of reference documents. Furthermore, another improvement to the system might be achieved by considering the structure of the documents [16]. In the area of competitive intelligence, this could be easily done as most of the bases used for technology watch contain highly structured documents (such as XML files for examples). Unexpectedness measures could then be parameterized by weights depending on the part of the documents. Boosting techniques such as in [7] could then be used to automatically learn those weights and discover unexpected information.

References

1. R. Agrawal and R. Srikant. Mining sequential patterns. In *Eleventh International Conference on Data Engineering*, pages 3–14, Taipei, Taiwan, 1995. IEEE.
2. K. K. Bun and M. Ishizuka. Emerging topic tracking system. In *Proceedings of the International Conference on Web Intelligence*, LNAI 2198, pages 125–130, 2001.
3. U.M Fayyad, G. Piatetsky, P. Smyth, and R. Uthurusamy. *Advances in Knowledge Discovery and Data Mining.* AAAI/MIT Press, 1996.
4. R. Feldman and I. Dagan. Knowledge discovery from textual databases. In *Proceedings of KDD95*, pages 112–117, 1995.
5. B. Gilad and J. Herring. *The Art and Science of Business Intelligence Analysis.* JAI Press, 1996.
6. C. Halliman. *Business Intelligence Using Smart Techniques : Environmental Scanning Using Text Mining and Competitor Analysis Using Scenarios and Manual Simulation.* Information Uncover, 2001.

7. M.V. Joshi, R. Agarwal, and V. Kumar. Predicting rare classes: Can boosting make any weak learner strong? In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 297–306. ACM, 2002.
8. Y. Kodratoff. Knowledge discovey in texts: A definition and applications. In *Proceedings of the International Symposium on Methodologies for Intelligent Systems*, LNAI 1609, pages 16–29, 1999.
9. L. Lebart and M. Rajman. Computing similarity. In *Handbook of Natural Language Processing*, pages 477–505. Dekker, 2000.
10. B. Lent, R. Agrawal, and R. Srikant. Discovering trends in text databases. In *Proceedings of KDD'97*, pages 227–230. AAAI Press, 14–17 1997.
11. B. Liu, Y. Ma, and P. S. Yu. Discovering unexpected information from your competitors' web sites. In *Proceedings of KDD'2001*, pages 144–153, 2001.
12. P. Losiewicz, D.W. Oard, and R. Kostoff. Textual data mining to support science and technology management. *Journal of Int. Inf. Systems*, 15:99–119, 2000.
13. N. Matsumura, Y. Ohsawa, and M. Ishizuka. Discovery of emerging topics between communities on www. In *Proceedings Web Intelligence'2001*, pages 473–482, Maebashi, Japan, 2001. LNCS 2198.
14. L. T. Moss and S. Atre. *Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications.* Addison-Wesley, 2003.
15. Y. Ohsawa, N. E. Benson, and M. Yachida. Keygraph: Automatic indexing by co-occurrence graph based on building construction metaphor. In *Proceedings of the Advances in Digital Libraries Conference*, pages 12–18, 1998.
16. B. Piwowarski and P. Gallinari. A machine learning model for information retrieval with structured documents. In *Proceedings of the International Conference on Machine Learning and Data Mining in Pattern Recognition*, LNCS 2734, pages 425–438, July 2003.
17. K. Rajaraman and A.H. Tan. Topic detection, tracking and trend analysis using self-organizing neural networks. In *Proceedings of PAKDD'2001*, pages 102–107, Hong-Kong, 2001.
18. G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill, 1983.
19. F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, March 2002.
20. J.A. Swets. Information retrieval systems. *Science*, 141:245–250, 1963.
21. D. Zhu and A.L. Porter. Automated extraction and visualization of information for technological intelligence and forecasting. *Technological Forecasting and Social Change*, 69:495–506, 2002.
22. D. Zhu, A.L. Porter, S. Cunningham, J. Carlisie, and A. Nayak. A process for mining science and technology documents databases, illustrated for the case of "knowledge discovery and data mining". *Ciencia da Informação*, 28(1):7–14, 1999.