Text Mining: Natural Language techniques and Text Mining applications

M. Rajman, R. Besançon
Artificial Intelligence Laboratory, Computer Science Department,
Swiss Federal Institute of Technology
CH-1015 Lausanne, Switzerland
rajman@lia.di.epfl.ch, romaric@lia.di.epfl.ch

Abstract

In the general framework of knowledge discovery, Data Mining techniques are usually dedicated to information extraction from structured databases. Text Mining techniques, on the other hand, are dedicated to information extraction from unstructured textual data and Natural Language Processing (NLP) can then be seen as an interesting tool for the enhancement of information extraction procedures. In this paper, we present two examples of Text Mining tasks, association extraction and prototypical document extraction, along with several related NLP techniques.

Keywords

Text Mining, Knowledge Discovery, Natural Language Processing

1 INTRODUCTION

The always increasing importance of the problem of analyzing the large amounts of data collected by companies and organizations has led to important developments in the fields of automated Knowledge Discovery in Databases (KDD) and Data Mining (DM). Typically, only a small fraction (5-10%) of the collected data is ever analyzed. Furthermore, as the volume of available data grows, decision-making directly from the content of the databases is not feasible anymore.

Standard KDD and DM techniques are concerned with the processing of structured databases. Text Mining techniques are dedicated to the automated information extraction form unstructured textual data.

In Section 2, we present the differences between the traditional Data Mining and the more specific Text Mining approaches, and in the subsequent sections, we describe two examples of Text Mining applications, along with the related NLP techniques.

2 TEXT MINING VS DATA MINING

According to Fayyad, Piatetsky-Shapiro and Smyth (1996), Knowledge Discovery in Databases is '*the non-trivial process of identifying valid, novel, potentially useful and ultimately understandable patterns in data*', and therefore refers to the overall process of discovering informations from data. However, as the usual techniques (inductive or statistical methods for building decision trees, rule bases, nonlinear regression for classification,...) explicitly rely on the structuring of the data into predefined fields, Data Mining is essentially concerned with information extraction from *structured* databases.

Table 1 shows an example of Inductive Logic Programming based learning from an attribute-value database (Džerovski 1996). The presented tables contain the database and the rules induced by the mining process.

Potential Customer Table

Person	Age	Sex	Income	Customer
Ann Smith	32	F	10 000	yes
Joan Gray	53	F	1 000 000	yes
Mary Blythe	27	F	20 000	no
Jane Brown	55	F	20 000	yes
Bob Smith	50	M	100 000	yes
Jack Brown	50	M	200 000	yes

Married-To Table

Husband	Wife
Bob Smith	Ann Smith
Jack Brown	Jane Brown

induced Rules

if Income(Person) $\geq$ 100 000 **then** Potential-Customer(Person)
if Sex(Person) = F **and** Age(Person) $\geq$ 32 **then** Potential-Customer(Person)
if Married(Person, Spouse) **and** Income(Person) $\geq$ 100 000 **then** Potential-Customer(Spouse)
if Married(Person, Spouse) **and** Potential-Customer(Person) **then** Potential-Customer(Spouse)

Table 1 An example of Data Mining using ILP techniques

This example illustrates how strongly the rule generation process relies on the explicit structure of the relational database (presence of well-defined fields, explicit identification of attribute-value pairs).

In reality however, a large portion of the available information appears in textual and hence unstructured form (or more precisely in an implicitly structured form). Specialized techniques specifically operating on textual data then become necessary to extract information from such kind of collections of texts. These techniques are gathered under the name of Text Mining and, in order to discover and use the implicit structure (*e.g.* grammatical structure) of the texts, they may integrate some specific Natural Language Processing (used for example to preprocess the textual data).

Text Mining applications impose strong constraints on the usual NLP tools. For instance, as they involve large volumes of textual data, they do not allow to integrate complex treatments (which would lead to exponential and hence non tractable algorithms). Furthermore, semantic models for the application domains are rarely available, and this implies strong limitations on the sophistication of the semantic and pragmatic levels of the linguistic models.

In fact, a working hypothesis (Feldman and Hirsh 1997) build upon the experience gained in the domain of Information Retrieval assumes that shallow representations of textual information often provides sufficient support for a range of information access tasks.

3 ASSOCIATION EXTRACTION FROM INDEXED DATA

If the textual data is indexed, either manually or automatically with the help of NLP techniques (such as the ones described in section 3.3), the indexing structures can be used as a basis for the actual knowledge discovery process.

In this section, we present a way of finding information in a collection of indexed documents by automatically retrieving relevant associations between key-words.

3.1 Associations : definition

Let's consider a set of key-words $A = \{w_1, w_2, ..., w_m\}$ and a collection of indexed documents $T = \{t_1, t_2, ..., t_n\}$ (*i.e.* each t_i is associated with a subset of A denoted $t_i(A)$).

Let $W \subseteq A$ be a set of key-words, the set of all documents t in T such that $W \subseteq t(A)$ will be called the **covering set** for W and denoted $[W]$.

Any pair (W, w), where $W \subseteq A$ is a set of key-words and $w \in A \backslash W$, will be called an **association rule**, and denoted $R\ :\ (W \Rightarrow w)$.

Given an association rule $R\ :\ (W \Rightarrow w)$,

- $S(R,T) = |[W \cup \{w\}]|$ is called the **support** of R with respect to the collection T ($|X|$ denotes the size of X)
- $C(R,T) = \frac{|[W \cup \{w\}]|}{|[W]|}$ is called the **confidence** of R with respect to the collection T.
  Notice that $C(R,T)$ is an approximation (maximum likelihood estimate) of the conditional probability for a text of being indexed by the key-word w if it is already indexed by the key-word set W.

An association rule R generated from a collection of texts T is said to satisfy support and confidence constraints σ and γ if

$$S(R,T) \geq \sigma \text{ and } C(R,T) \geq \gamma$$

To simplify notations, $[W \cup \{w\}]$ will be often written $[Ww]$ and a rule $R\ :\ (W \Rightarrow w)$ satisfying given support and confidence constraints will be simply written as:

$$W \Rightarrow w \quad S(R,T)/C(R,T)$$

3.2 Mining for associations

Experiments of association extraction have been carried out by Feldman *et al.* (1996) with the KDT (Knowledge Discovery in Texts) system on the Reuter corpus. The Reuter corpus is a set of 22 173 documents that appeared on the Reuter newswire in 1987. The documents were assembled and manually indexed by Reuters Ltd. and Carnegie Group Inc. in 1987. Further formatting and data file production was done in 1991 and 1992 by David D. Lewis and Peter Shoemaker.

The documents were indexed with 135 categories in the Economics domain. The mining was performed on the indexed documents only (*i.e* exclusively on the key-word sets representing the real documents).

All known algorithms for generating association rules operate in two phases. Given a set of key-words $A = \{w_1, w_2, ..., w_m\}$ and a collection of indexed documents $T = \{t_1, t_2, ..., t_n\}$, the extraction of associations satisfying given support and confidence constraints σ and γ is performed:

- by first generating all the key-word sets with support at least equal to σ (*i.e.* all the key-word sets W such that $|[W]| \geq \sigma$). The generated key-word sets are called the **frequent sets** (or **σ-covers**);
- then by generating all the association rules that can be derived from the produced frequent sets and that satisfy the confidence constraint γ.

(a) Generating the frequent sets

The set of candidate σ-covers (frequent sets) is built incrementally, by starting from singleton σ-covers and progressively adding elements to a σ-cover as long as it satisfies the confidence constraint.

The frequent set generation is the most computationally expensive step (exponential in the worse case). Heuristic and incremental approaches are currently investigated.

A basic algorithm for generating frequent sets is indicated in Algorithm 1.

```
$i = 1, Cand_i = \{\{w\}, |[\{w\}]| \geq \sigma\}$, where $w$ are key-words;
while ($Cand_i \neq \emptyset$) do
    $Cand_{i+1} = \{S_1 \cup S_2 |\ S_1, S_2 \in Cand_i,$
                        and $|S_1 \cup S_2| = i + 1$
                        and $\forall S \subseteq S_1 \cup S_2, (|S_1 \cup S_2| = i) \Rightarrow (S \in Cand_i)$
                        and $|[S_1 \cup S_2]| \geq \sigma\}$
    $i = i + 1$;
endw
```

Algorithm 1: Generating the frequent sets

(b) Generating the associations

Once the maximal frequent sets have been produced, the generation of the associations is quite easy. A basic algorithm is presented in Algorithm 2.

```
foreach $W$ maximal frequent set do
    generate all the rules $W \backslash \{w\} \Rightarrow \{w\}$, where $w \in W$, such that
    $\frac{|[W \backslash \{w\}]|}{|[W]|} \geq \sigma$;
endfch
```

Algorithm 2: Generating the associations

(c) Examples

Concrete examples of associations rules found by KDT on the Reuter Corpus are provided in Table 2. These associations were extracted with respect to specific queries expressed by potential users.

query	:	‘*find all associations between a set of countries including Iran and any person*’
result	:	[Iran, Nicaragua, Usa] ⇒ Reagan 6/1.000 ...
query	:	‘*find all associations between a set of topics including Gold and any country*’
result	:	[gold, copper] ⇒ Canada 5/0.556 [gold, silver] ⇒ USA 18/0.692 ...

Table 2 Examples of associations found by KDT

3.3 NLP techniques for association extraction: Automated Indexing

In the case of the Reuter Corpus, document indexing has been done manually, but, as manual indexing is a very time–consuming task, it is not realistic to assume that such a processing could systematically be performed in the general case. Automated indexing of the textual document base, performed for example in a preprocessing phase, has to be considered in order to allow the use of association extraction techniques on a large scale.

Techniques for automated production of indexes associated with documents can be borrowed from the Information Retrieval field. In this case, they usually rely on frequency-based weighting schemes (Salton and Buckley 1988). Several examples of such weighting schemes are provided in the SMART Information Retrieval system. Formula (1) presents the SMART atc weighting scheme.

$$w_{i,j} = \begin{cases} 0.5 \times (1 + \frac{p_{i,j}}{max_l(p_{i,l})}) \log(\frac{N}{n_j}) & \text{if } p_{i,j} \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where $w_{i,j}$ is the weight of word w_j in document t_i, $p_{i,j}$ is the relative document frequency of w_j in t_i ($p_{i,j} = f_{i,j}/\sum_k f_{i,k}$, where $f_{i,j}$ is the number of occurrences of w_j in t_i), N is the number of documents in the collection and n_j is the number of documents containing w_j.

Once a weighting scheme has been selected, automated indexing can be performed by simply selecting, for each document, the words satisfying given weight constraints.

The major advantage of automated indexing procedures is that they drastically reduce the cost of the indexing step.One of their main drawbacks is however that, when applied without additional knowledge (such as a thesaurus), they produce indexes with extremely reduced generalization power (key-words have to be explicitly present in the documents, and do not always provide a good thematic description).

3.4 Additional issues

(a) Integration of background knowledge

If background knowledge is available (for example some factual knowledge about the application domain), additional constraints can be integrated in the association generation procedure (either in the frequent set generation, or directly in the association extraction). An example of a system using background knowledge for association generation is the FACT system developed by (Feldman and Hirsh 1996).

(b) Generalization of the notion of association

Several generalizations are possible for the notion of association (rule):

- rules with more than one key-word in their right-hand side, which can express more complex implications;
- more general attributes (*i.e.* not only restricted to key-words presence / absence): discrete and continuous variables;
- non implicative relations, such as pseudo-equivalences;
- different quality measures providing alternative approaches for confidence evaluation.

An example of system integrating such kinds of generalizations is the GUHA system developed at the Institute of Computer and Information Science in Prague.

4 PROTOTYPICAL DOCUMENT EXTRACTION FROM FULL TEXT

The association extraction presented in the previous section exclusively operates on the document indexes, and therefore does not directly take advantage of the textual content of the documents. Approaches based on full text mining for information extraction can then be considered.

Our initial experiments on the Reuter Corpus (Rajman and Besançon 1997) were dedicated to the implementation and evaluation of association extraction techniques operating on all the words contained in the documents instead of only considering the associated key-words. The obtained results showed however that association extraction based on full text documents does not provide effectively exploitable results. Indeed, the association extraction process either just detected compounds, *i.e.* domain-dependent *terms* such as [*wall*]

⇒ *street* or [*treasury secretary james*] ⇒ *baker*, which cannot be considered as '*potentially useful*' (referring to the KDD definition given in section 2) or extracted uninterpretable associations such as [*dollars shares exchange total commission stake*] ⇒ *securities*, that could not be considered as '*ultimately understandable*'.

We therefore had to seek for a new TM task that would be more adequate for full text information extraction out of large collections of textual data. We decided to concentrate on the extraction of **prototypical documents**, where 'prototypical' is informally defined as corresponding to an information that occurs in a repetitive fashion in the document collection. The underlying working hypothesis is that repetitive document structures provide significant information about the textual base that is processed.

Basically, the method presented in this section relies on the identification of frequent sequences of terms in the documents, and uses NLP techniques such as automated Part-of-Speech Tagging and Term Extraction to preprocess the textual data.

The NLP techniques can be considered as an automated generalized indexing procedure that extracts from the full textual content of the documents linguistically significant structures that will constitute a new basis for frequent set extraction.

4.1 NLP Preprocessing for prototypical document extraction

(a) Part-Of-Speech tagging

The objective of the Part-Of-Speech tagging (POS-Tagging) is to automatically assign Part-of-Speech tags (*i.e.* morpho-syntactic categories such as *noun, verb, adjective,...*) to words in context. For instance, a sentence as 'a computational process executes programs' should be tagged as 'a/DET computational/ADJ process/N executes/V programs/N'. The main difficulty of such a task is the lexical ambiguities that exist in all natural languages. For instance, in the previous sentence, both words 'process' and 'programs' could be either nouns(N) or verbs(V).

Several techniques have been designed for POS-tagging:

- Hidden Markov Model based approaches (Cutting *et al.* 1992);
- Rule-based approaches (Brill 1992);

If a large lexicon (providing good coverage of the application domain) and some manually hand-tagged text are available, such methods perform automated POS-tagging in a computationally very efficient way (linear complexity) and with a very satisfying performance (on the average, 95-98% accuracy).

One of the important advantage of POS-tagging is to allow automated filtering of non-significant words on the basis of their morpho-syntactic category. For instance, in our experiments (where we used the E. Brill's rule-based tagger (Brill 1992)), we decided to filter out articles, prepositions, conjunctions,... therefore restricting the effective mining process to nouns, adjectives, and verbs.

(b) term extraction

In order to automatically detect domain-dependent compounds, a term extraction procedure has been integrated in the preprocessing step.

Automated term extraction is indeed one of the critical NLP tasks for various applications (such as terminology extraction, enhanced indexing...) in the domain of textual data analysis.

Term extraction methods are often decomposed into two distinct steps (Daille 1994):

- extraction of term candidates on the basis of structural linguistic information; for example, term candidates can be selected on the basis of relevant morpho-syntactic patterns (such as 'N Prep N': board of directors, Secretary of State,...; 'Adj N': White House, annual rate,...; etc);
- filtering of the term candidates on the basis of some statistical relevance scoring schemes, such as frequency, mutual information, Φ^2 coefficient, loglike coefficient,...; in fact, the actual filters often consist of combinations of different scoring schemes associated with experimentally defined thresholds.

In our experiments, we used 4 morpho-syntactic patterns to extract the term candidates: 'Noun Noun' (1), 'Noun of Noun' (2), 'Adj Noun'(3), 'Adj Verbal'(4). In order to extract more complex compounds such as 'Secretary of State George Shultz', the term candidate extraction was applied in an iterative way where terms identified at step n were used as atomic elements for step $n+1$ until no new terms were detected. For example, the sequence 'Secretary/N of/prep State/N George/N Shultz/N' was first transformed into 'Secretary–of–State/N George–Shultz/N' (patterns 2 and 1) and then combined into a unique term 'Secretary–of–State–George–Shultz/N' (pattern 1).A purely frequency-based scoring scheme was then used for filtering.

The prototype integrating POS-tagging and term extraction that we used for our experiments was designed in collaboration with R. Feldman's team at Bar Ilan University.

4.2 Mining for prototypical documents

(a) The extraction process

The extraction process can be decomposed into four steps:

- NLP preprocessing: POS-tagging and term extraction, as described in the previous section;
- frequent term sets generation using an algorithm globally similar to the one described in Algorithm 1 (with some minor changes, particularly concerning the data representation);
- clustering of the term sets based on a similarity measure derived from the number of common terms in the sets;
- actual production of the prototypical documents associated with the obtained clusters.

The whole process is described in more detail in subsection (b), on the basis of a concrete example.

As we already mentioned earlier, association extraction from full text documents provided uninterpretable results, indicating that associations constitute an inadequate representation for the frequent sets in the case of full text mining. In this sense, the prototypical documents are meant to correspond to more operational structures, giving a better representation of the repetitive documents in the text collection and therefore providing a potentially useful basis for a partial synthesis of the information content hidden in the textual base.

(b) example

Figure 1 presents an example of a (SGML tagged) document from the Reuter Corpus.

Figure 2 presents, for the same document, the result of the NLP preprocessing step (POS-tagging and term extraction: the extracted terms are printed in boldface).

During the production of term sets associated with the documents, filtering of non-significant terms is performed, on the basis of:

- morpho-syntactic information: we only keep nouns, verbs and adjectives;
- frequency criteria: we only keep terms with frequency greater than a given minimal support;
- empiric knowledge: we remove some frequent but non-significant verbs (is, has, been,...).

After this treatment, the following indexing structure (term set) is obtained for the document and will serve as a basis for the frequent set generation:

```
<REUTERS NEWID=”2088”>
(...)
<BODY>Nissan Motor Co Ltd <NSAN.T> is issuing a 35 billion yen eurobond
due March 25 1992 paying 5-1/8 pct and priced at 103-3/8, Nikko Securities Co
(Europe) Ltd said.
The non-callable issue is available in denominations of one mln Yen and will be
listed in Luxembourg.
The payment date is March 25.
The selling concession is 1-1/4 pct while management and underwriting combined
pays 5/8 pct.
Nikko said it was still completing the syndicate. </BODY></TEXT>
</REUTERS>
```

Figure 1 An example of Reuter Document

<DOC2088>
Nissan_Motor_Co_Ltd/N “/“ NSAN/N ./. T/N ”/” is/V issuing/V a/DET **35_billion_yen/CD** eurobond/V due/ADJ **March_25/CD** 1992/CD paying/V **5-1/8_percent/CD** and/CC priced/V at/PR 103-3/8/ADJ ,/, **Nikko_Securities_Co/N** (/(Europe/N)/SYM Ltd/N said/V ./.
The/DET **non-callable_issue/N** is/V available/ADJ in/PR denominations/N of/PR **one_million/CD** Yen/CD and/CC will/MD be/V listed/V in/PR Luxembourg/N ./.
The/DET **payment_date/N** is/V **March_25/CD** ./.
The/DET **selling_concession/N** is/V **1-1/4_percent/CD** while/PR management/N and/CC underwriting/N combined/V pays/V **5/8_percent/CD** ./.
Nikko/N said/V it/PRP was/V still/RB completing/V the/DET syndicate/N ./.

Figure 2 A tagged Reuter Document

{available/adj combined/v denominations/n due/adj europe/n issuing/v listed/v luxembourg/n management/n paying/v payment_date/n pays/v priced/v selling_concession/n syndicate/n underwriting/n}

The frequent sets generation step (of course operating on the whole document collection) then produces, among others, the following frequent term sets (POS-tags have been removed to increase readability):

{due available management priced issuing paying denominations underwriting} 86
{due available management priced issuing denominations payment_date} 87
{due available management priced issuing denominations underwriting luxembourg} 81
{due management selling priced issuing listed} 81
{due priced issuing combined denominations payment_date} 80
{management issuing combined underwriting payment_date} 80
(...)

where the numeric values correspond to the frequency of the sets in the collection.

In order to reduce the important information redundancy due to partial overlapping between the sets, clustering was performed to gather some of the term sets into classes (clusters), represented by the union of the sets:

{due available management priced issuing combined denominations listed underwriting luxembourg payment_date paying} 45

To reduce the possible meaning shifts linked to non corresponding word sequences, the term sets representing identified clusters were split into sets of distinct terms *sequences* associated with paragraph boundaries in the original documents. The most frequent sequential decompositions of the clusters are then computed and some of the corresponding document excerpts extracted. These document excerpts are by definition the prototypical documents corresponding to the output of the mining process.

Figure 3 presents both the most frequent sequential decomposition for the previous set and the associated prototypical document.

(issuing due paying priced) (available denominations listed luxembourg) (payment_date) (management underwriting combined) 41

<**DOC2088**>
Nissan_Motor_Co_Ltd "NSAN.T" is **issuing** a 35_billion_yen eurobond **due** March_25 1992 **paying** 5-1/8_percent and **priced** at 103-3/8, Nikko_Securities_Co (Europe) Ltd said.
The non-callable_issue is **available** in **denominations** of one_million Yen and will be **listed** in **Luxembourg**.
The **payment_date** is March_25.
The selling_concession is 1-1/4_percent while **management** and **underwriting combined** pays 5/8_percent.
Nikko said it was still completing the syndicate.

Figure 3 An example of prototypical document

4.3 Future Work

(a) Name entity tagging

We performed syntactic Part-Of-Speech tagging on the document base. Similar techniques can also be used for semantic tagging.

For instance, the Alembic environment, developed by the MITRE Natural Language Processing Group (MITRE NLP Group 1997), correspond to a set of techniques allowing rule based name entity tagging. The rules used by the system have been automatically learned from examples.

Figure 4 presents the prototypical document given in the previous section, as tagged by Alembic. This tagging has been provided by Christopher Clifton, from MITRE, and used two rule bases, trained to respectively recognize person/location/organization, and date/time/money/numbers.

This kind of semantic tagging will be undoubtfully useful for the generalization of the variable parts in prototypical documents, and could be considered

```
<s> <ENAMEX TYPE=ORGANIZATION>Nissan Motor Co Ltd</ENAMEX>
"<ENAMEX TYPE=ORGANIZATION>NSAN</ENAMEX>.</s><s>T" is issuing a <NUMBER>35</NUMBER> <NUMBER>billion</NUMBER> yen
eurobond due <TIMEX TYPE=DATE>March 251992</TIMEX> paying
<NUMBER>5</NUMBER>-<NUMBER>1/8</NUMBER> percent and priced
at <NUMBER>103</NUMBER>-<NUMBER>3/8</NUMBER> , <ENAMEX
TYPE=ORGANIZATION>Nikko Securities Co</ENAMEX> ( <ENAMEX
TYPE=LOCATION>Europe</ENAMEX> ) Ltd said.</s>
<s>The non-callable issue is available in denominations of
<NUMBER>one</NUMBER> <NUMBER>million</NUMBER> <ENAMEX
TYPE=ORGANIZATION>Yen</ENAMEX> and will be listed in <ENAMEX
TYPE=LOCATION>Luxembourg</ENAMEX>.</s>
<s>The payment date is <TIMEX TYPE=DATE>March 25</TIMEX>.</s>
<s>The selling concession is <NUMBER>1</NUMBER>-
<NUMBER>1/4</NUMBER> percent while management and underwriting
combined pays <NUMBER>5/8</NUMBER> percent.</s>
<s><ENAMEX TYPE=ORGANIZATION>Nikko</ENAMEX> said it was still
completing the syndicate.</s>
```

Figure 4 Name entity tagging of a Prototypical Document

as an abstraction process that will provide a better representation of the synthetic information extracted from the base.

(b) Implicit user modeling

In any information extraction process, it is of great interest to try to take into account an interaction with the user. Experiments in Information Retrieval (IR) have shown for instance that better relevance results can be obtained by using *relevance feedback* techniques (techniques that allow to integrate relevance evaluation by the user of the retrieved documents).

In our model, such an approach could lead to integrate both *a posteriori* and *a priori* information about the user, and therefore correspond to the integration of an implicit model of the user.

- A posteriori information could be obtained, with a similar procedure as in classical IR processes, through the analysis of the reactions of the user concerning the results provided by the TM system (relevance or usefulness of extracted prototypical documents).
- A priori information could be derived, for example, from any pre-classification of the data (often present in the real data: for example, users often classify their files in directories or folders). This user pre-partitioning of the document base contain interesting information about the user and could serve as a basis for deriving more adequate parameters for the similarity measures (for instance, the parameters could be tuned in order to minimize inter-class similarity, and maximize intra-class similarity).

5 CONCLUSION

The general goal of Data Mining is to automatically extract information from databases. Text Mining corresponds to the same global task but specifically applied on unstructured textual data. In this paper, we have presented two different TM tasks: association extraction from a collection of indexed documents, designed to answer specific queries expressed by the users, and prototypical document extraction from a collection of full-text documents, designed to automatically find information about classes of repetitive document structures that could be used for automated synthesis of the information content of the textual base.

REFERENCES

Brill E. (1992) A Simple Rule-Based Part-of-Speech Tagger. In *Proc. of the 3rd Conf. on Applied Natural Language Processing.*

Cutting D. *et al.* (1992) A Practical Part-of-Speech Tagger. In *Proc. of the 3rd Conf. on Applied Natural Language Processing.*

Daille B. (1994) Study and Implementation of Combined Techniques for Automatic Extraction of Terminology. In *Proc. of the 32nd Annual Meeting of the Association for Computational Linguistics.*

Džerovski S. (1996) Inductive logic programming and Knowledge Discovery in Databases. In *Advances in Knowledge Discovery and Data Mining.* AAAI Press / The MIT Press.

Fayyad U.M., Piatetsky-Shapiro G. and Smyth P. (1996) From Data Mining to Knowledge Discovery: An Overview. In *Advances in Knowledge Discovery and Data Mining.* AAAI Press / The MIT Press.

Feldman R., Dagan I. and Kloegsen W. (1996) Efficient Algorithm for Mining and Manipulating Associations in Texts. 13th *European Meeting on Cybernetics and Research.*

Feldman R. and Hirsh H. (1996) Mining Associations in Text in the Presence of Background Knowledge. In *Proc. of the 2nd Int. Conf. on Knowledge Discovery.*

Feldman R. and Hirsh H. (1997) Finding Associations in Collections of Text. In Michalski R.S., Bratko I. and Kubat M. (edts) *Machine Learning, Data Mining and Knowledge Discovery: Methods and Application* (John Wiley and sons Ltd).

MITRE NLP Group (1997) Alembic Language Processing for Intelligence Applications. At URL :
http://www.mitre.org/resources/centers/advanced_info/g04h/nl-index.html

Rajman M. and Besançon R. (1997) A Lattice Based Algorithm for Text Mining. Technical Report TR-LIA-LN1/97, Swiss Federal Institute of Technology.

Salton G. and Buckley C. (1988) Term Weighting Approaches in Automatic

Text Retrieval. *Information Processing and Management*, 24:5, 513-523.

6 BIOGRAPHY

Born 1962, **Martin RAJMAN** graduated from the Ecole Nationale Supérieure des Télécommunications (ENST, Paris), where he also obtained a PhD in Computer Science. In March 1992, M. RAJMAN joined the permanent teaching and research staff of the ENST as member of the Artificial Intelligence Group, responsible for the Natural Language activity. Since September 1996, he is member of the Artificial Intelligence Laboratory of the Ecole Polytechnique Fédérale de Lausanne (EPFL), where he is in charge of the Natural Language Processing (NLP) group.

Born 1972, **Romaric BESANÇON** graduated from the Institut d'Informatique d'Entreprise (IIE, Evry) and then obtained a DEA from the University Paris-XI (Orsay). He is currently research assistant at the EPFL NLP group, where he is working in the domain of Text-Mining.