EXTRACTING THE INFORMATION CONTAINED IN ELECTRONIC DOCUMENTS: A PATTERN MATCHING ALGORITHM FOR TEXT MINING

A. Christy
$Research Scholar, Sathyabama University
christy_a1@hotmail.com

P. Thambidurai
Principal, Peruntalaivar Kamarajar Institute of Engineering and Technology
ptdurai58@yahoo.com

Abstract

In Today's world, the huge amount of unstructured data available on the Web, Intranets, Journals, Magazines, E-mails, etc. creates information overloading problem. So, managing the knowledge contained in the textual documents is an important problem of Knowledge Management. Knowledge Extraction from collections of data is possible by Knowledge Discovery in Database (KDD), an interactive and iterative process focused on the exploration of data to discover new and interesting patterns within them. In this paper, we propose a domain-independent algorithm for information extraction, called SOFTRULEMINING for extracting the aim, methodology and conclusion specified by authors in technical abstracts. The algorithm is implemented by combining trigram model combined with soft matching rules and is tested with technical abstracts of www.computer.org and www.ansinet.org and it is found that the system has improved its recall value against search engines, which concentrates on improving the precision values.

Keywords

Parsing, Trigram model, soft matching, information extraction, recall, precision, etc.

Introduction

A recent study indicated that 80% of a company's information was contained in text documents, such as emails, memos, customer correspondence, and reports. Text-based applications involve the processing of written text, such as books, newspapers, reports, manuals, e-mail messages, and so on. Text-based natural language research is ongoing in applications such as :

- Finding appropriate documents on certain topics from a database of texts
- Extracting information from messages or articles on certain topics
- Translating documents from one language to another
- Summarizing texts for certain purposes

Text mining or text data mining, the process of finding useful or interesting patterns, models, directions , trends or rules from unstructured text, is used to describe the application of data mining techniques to automated discovery of knowledge from text (Chakrabarti, 2002 ; Han and Kamber, 2000). Text mining has been viewed as a natural extension of data mining (Hearst, 1999, 2003) or sometimes considered as a task of applying the same data mining techniques to the domain of textual information (Dorre, Gerstl and Seiffert, 1999). This reflects the fact that the advent of text mining relies on the burgeoning field of data mining to a great degree (Un Yong Nahm, 2004). Currently text mining is enjoying a surge of interest fueled by the availability of the Internet, the success of bioinformatics, and a rebirth of computational linguistics.

Text mining is different from web search. In search, the user is typically looking for something that is already known and has been written by someone else. The goal of text mining is to discover unknown information, something that no one yet knows and so could not have yet written down. It is the mapping of natural language texts (such as newswire reports, newspaper and journal articles, electronic mail, World Wide Web pages, any textual database, etc.) into predefined, structured representation, or templates, which, when filled, represent an extract of key information from the original text. The information concerns entities of interest in the application domain (e.g. companies or persons), or relations between such entities, usually in the form of events in which the entities take part (e.g. company takeovers, management successions etc.).

This research work, concentrates on Information Extraction (IE), as it plays a major role in converting unstructured documents to structured form. IE is the process of identifying relevant information where the criteria for relevance are predefined by the user in the form of a template that is to be filled. Typically, the template pertains to events or situations, and contains slots that denote who did, what, to whom, when, and where, and possibly why. The template builder has to predict what will be of interest to the user and define its slots and selection criteria accordingly.

System Architecture

The objective of the system is to extract the aim, methodology and conclusion specified by authors in technical abstracts. The general architecture of a text mining system is depicted in Fig. 1.The system deals with extracting information from multiple documents, stored in database and using data mining techniques to extract knowledge in the form of rules.

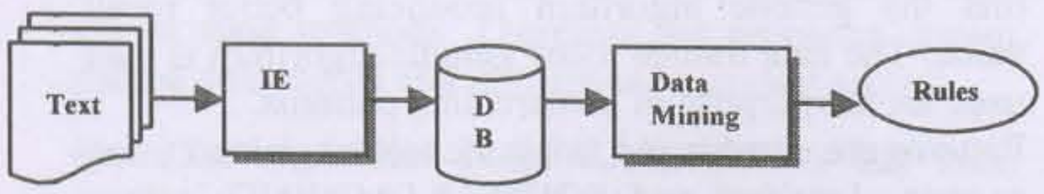


Fig.1 General Architecture of IE systems

Inorder to retrieve rules, we have used the genre of technical abstracts. Abstracts have a well-defined structure that authors use to summarize their ideas and state key facts concisely. This makes abstracts suitable for further shallow analysis and avoids many conceptual-level ambiguities related to the restricted use of concepts in specific contexts. Linguistic evidence shows that an abstract in a given domain follows a prototypical and even modular organization ie, the genre dependent rhetorical structure in which that its author uses to express the background information, methods, achievements, and conclusions. From a scientific viewpoint, there are also claims that important findings could be searched by linking this kind of information across the documents [1]. Therefore, our system retrieves the aim, methodology and conclusion from the abstracts and it is represented as a rule like form in terms of predicates as shown in Fig. 2 .

Goal
Generalize (error, diffusion, produce, FM, halftones .....
Method
Enable(encoding ,subsequent decoding , information .....
Conclusion
Show (block-error, diffusion , applied , embed .....

Fig. 2 Rules to be extracted

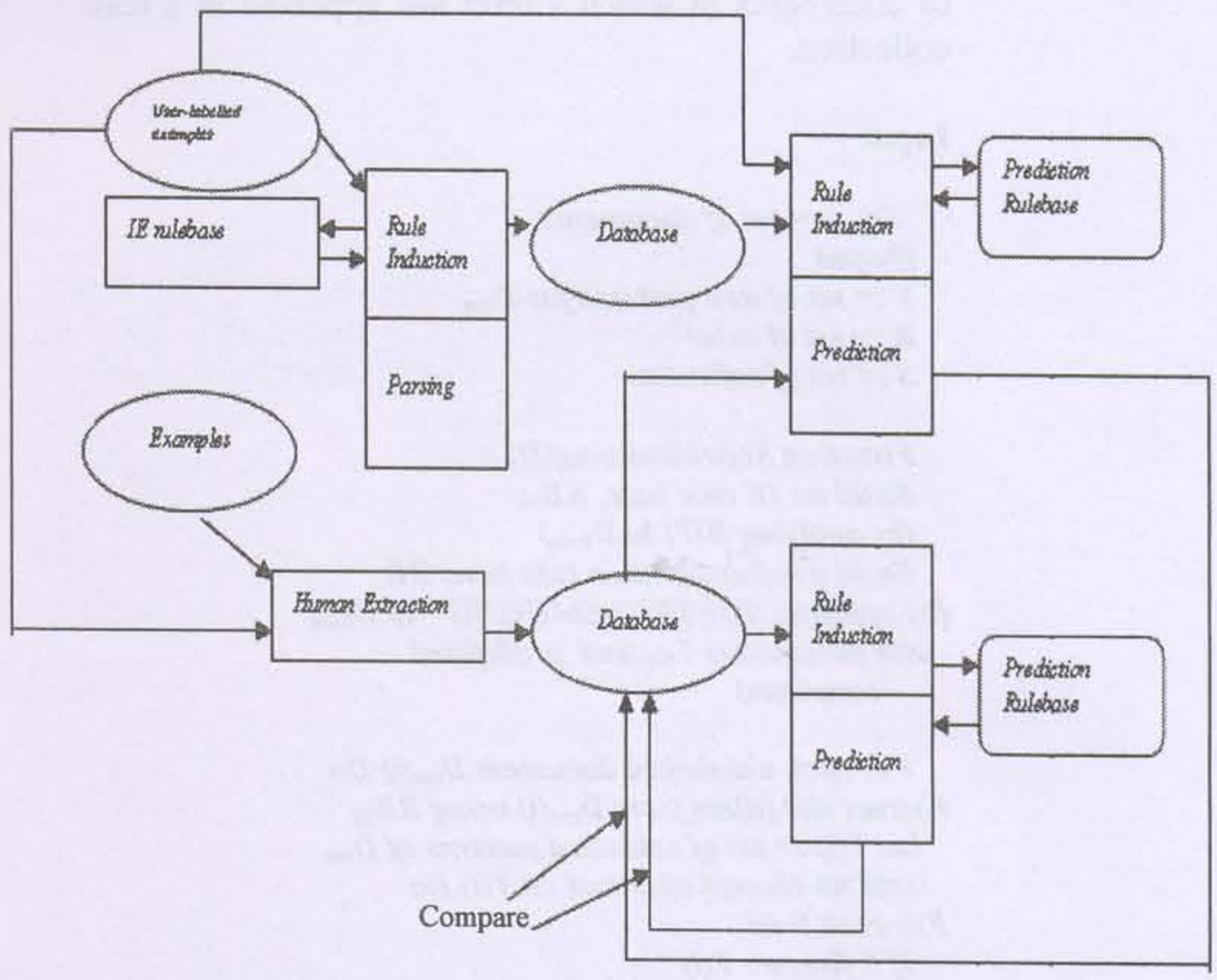
To extract the rules, the IE task takes the set of tagged documents and produces a template representation for every document. This can be easily converted into rule-like form. For this purpose, we wrote a set of domain-independent extraction patterns so that we could match them against the input documents. Each extraction pattern constructs an output representation that involves two levels of linguistic knowledge: the rhetorical information expressed in the abstract and the semantic information contained in it, which we later convert into a predicate-like form. The left-hand expression states the pattern to be identified and the right hand side (following the colon) states the corresponding semantic action to be produced. The overall system architecture is depicted in Fig. 3.

The algorithm starts with the initial state ((S) 1) and no backup states.

1. Select the current state: Take the first state off the possibilities list and call it C. If the possibilities list is empty, then the algorithm fails (that is, no successful parse is possible)
2. If C consists of an empty symbol list and the word position is at the end of the sentence, then the algorithm succeeds.
3. Otherwise, generate the next possible states.

If the first symbol on the symbol list of C is the one defined in the class for previous token then add it to the possibilities list

If the third token on the symbol list of C is the one defined in the class for the next token then add it to the possibilities list.



Softmatching Rules

The IE system in this work is extracted using trigram model and rules are constructed using patterns which need not strictly adhere to the procedure. The Fig. 4 shows a sample of softmatching rules, those are introduced.

Propose(we propose, I propose, proposes, proposed) ∈Trigram model ∈ verb ➔ aim

Method(principle, algorithm, concept, technique, work,challenge) ∉ trigram ∉ verb ➔ method

Conclude (experiment, result, test, tested, train, analytical) ∉trigram ∉ verb ➔ conclusion

Fig .4 Sample softmatching rules

The above rules are softmatching rules, as these are some frequently occurring terms which best fits the templates. Introduction of these softmatching rules have shown the improvement over the precision value, so as the recall. The algorithm SOTRULEMINING is implemented for Information extraction using softmatching rules and is depicted in Fig. 5.

As Information extraction systems are domain specific, machine learning plays a vital role in classification and prediction. During the learning

process of machine learning, a sample of the database is used to train the system to properly perform the desired task. The quality of the training data determines how well the program learns. The documents are trained with a bag of words and in order to normalize the keywords, the inverse document frequency is used in which each document can be represented as a term vector of the form $\bar{a} = (a_1, a_2, \ldots a_n)$, Each term a_i has a weight w_i associated with it and w_i denotes the normalized frequency of word in the vector space , where $w_i = tf_i \cdot idf_i$ where tf_i is the term frequency of a_i, idf_i is inverse document frequency denoted as log (N/DF) where N is the total number of documents and DF is the number of documents in which a term has appeared in a text collection.

Input :

D_{train} := set of documents
Output :
T := set of new patterns for D_{test}
R := set of rules
S := set of softrules

Function Softrulemining(D)
Build an IE rule base, RB_{IE}
(by applying BWI to D_{train})
Build a softassociation rule base, RB
(by applying SOFTRULEMINING to D_{train} with parameters T_{sim} and predefined templates)

For each unlabelled document $D_{test}(i)$ Do
Extract slot fillers from $D_{test}(i)$ using RB_{IE}
Let T(i) := set of extracted patterns of D_{test}
Until no change obtained on T(i) Do
For each S do
If S fires on T(i)
Extract the pattern in $D_{test}(i)$
Else
For each R $\in$RB do
If R fires on T(i)
Extract the patterns in $D_{test}(i)$
For each matching pattern Y' in $D_{test}(i)$ do
Tf := tf.log (N/df)
(with $Tf < T_{sim}$)
if $tf < T_{sim}$
add Y' to T(i)
Let T := (T(1),T(2),.....T(n))
Return T.

Fig. 5 SOFTRULEMINING Algorithm

RESULTS AND DISCUSSION

Discovered knowledge is only useful and informative if it is accurate. It is important to measure the discovered knowledge on independent test data. For the dataset, 200 abstracts were collected from www.computer.org containing 2 data sets related to information retrieval and image processing and manually annotated with correct extraction patterns. Inorder to construct the patterns classification algorithms C4.8, Random tree, Random forest, Decision tree, Decision stump were used with 10-folds cross validation. Genetic algorithms with crossover probability 0.99 and mutation level 0.01, it is found that the genetic algorithm producing better recall value. The data trained using genetic algorithm is then used for the purpose of constructing patterns.

Patterns are constructed using the tokens trained using genetic algorithm and SOFTRULEMINING is then used for information extraction. The results obtained using SOFTRULEMINING is compared with results of HMM model and Hardmatchingrules. The results are depicted in Table 1.The patterns, which are constructed are verified using training data and tested using different domains on www.computer.org and www.ansinet.org. Three fourth of the technical magazines from www.computer.org are checked using the proposed algorithm and it is found that the system has improved its recall value after the implementation of softmatching rules.

Technique	Category	Domain	Precision	Rec
Trigram model with SOFT matching rules	Aim	Domain 1	1	
		Domain 2	.98	
	Methodology	Domain 1	1	
		Domain 2	1	
	Conclusion	Domain 1	.94	
		Domain 2	.96	
Trigram model with Hardmatching rules	Aim	Domain 1	.90	
		Domain 2	.76	
	Methodology	Domain 1	.88	
		Domain 2	.84	
	Conclusion	Domain 1	.84	
		Domain 2	.81	
HMM models	Aim	Domain 1	0.9	
		Domain 2	0.9	
	Methodology	Domain 1	0.8	
		Domain 2	0.62	
	Conclusion	Domain 1	0.82	
		Domain 2	0.78	

Table 1. Experimental Results of IE using softmatching rules

Conclusion

Since the success of any machine learning algorithm depends on the type of features selected, patterns are constructed using softmatching rules, which improved the recall value of the information extraction system. The following are some of the findings of the system.

(i) In specifying the aim and conclusion authors have used only a frequent set of tokens in different domains than for specifying the methodology. More training is needed for identifying tokens for methodology.

(2) The system is tested with different websites having technical abstracts and the introduction of softmatching rules have shown good performance over the existing methods. Therefore the proposed system can be considered as a domain-independent system.

(3) The algorithm SOFTRULEMINING has been proposed and it has shown 84% recall value as against the other methods which have shown recall value of 70% and less.

References

[1] John Abutridy , Chris Mellish and Stuart Aitken . "Combining Information extraction with genetic algorithm for text mining",IEEE Intelligent Systems, May/June 2004., pg 22-30

[2] Eugene Agichtein and Luis Gravano. "Extracting Relations from Large Plain-Text collections", Columbia University.

[3] Raymond J. Mooney, Razvan Bunescu, 2005 "Mining Knowledge from Text using Information Extraction",ACM SIGKDD Explorations, 7: pp 3-10

[4] Yiyu Yao, Fei-Yue Wang and Daniel Zeng, Jue Wang. "Rule + Exception Strategies for Security Information Analysis", IEEE Intelligent Systems, September/October 2005 pp 52 – 57

[5] Jose´ Ramo´n Cano, Francisco Herrera, Manuel Lozano." Evolutionary stratified training set selection for extracting classification rules with trade off precision-interpretability",Data & Knowledge Engineering 60 (2007) PP 90–108

[6] Jeonghee Yi, Tetsuya Nasukawa Razvan Bunescu_, Wayne Niblack ." Sentiment Analyzer: Extracting Sentiments about a Given Topic using Natural Language Processing Techniques", IBM Almaden Research Center.

[7] Hsin-Hsi Chen, Lun-Wei Ku. "Description of a Topic Detection Algorithm on TDT3 Mandarin Text", National Taiwan University.