

Removal of Network Ambiguities through Knowledge based System

Pradeep Kumar

Research Scholar

Dept. Information Technology
Kumaun University, Nainital

Sumit Khulbe

Assistant Professor

Dept. Computer Science
Kumaun University, Nainital

H S Dhami

Professor

Dept. Mathematics
Kumaun University, Nainital

ABSTRACT

Documents on the Internet are composed of several kinds of multimedia information when accessed for personal, entertainment, business, and scientific purposes. There are many specific content domains of interest to different communities of users. Extracting semantic relationships between entities from text documents is challenging task in information extraction. By semantics for natural language in this connection, this paper understand not just the relating of a semantic representation language to natural language but the evaluation of natural language expressions with respect to databases. Evaluating a declarative sentence (on a given reading) with respect to a database involves determining whether the sentence is true with respect to the data base, whether the sentence appropriately describes the database. Evaluating a question with respect to a database might determine what information in the database would lead to appropriate answers to the question. The implementation of a knowledge-based system that deals with the Very Large Scale requires the important consideration of several problems, including the complexity of the domain, the nature of information processing, and the automation requirements to this problem is the aim of this work. It addresses the incorporation of diverse lexical, syntactic and semantic knowledge in feature-based relation extraction using support vector machines. This paper have used the base phrase chunking information for relation extraction and have also demonstrated the use of Word Net in feature-based relation extraction to further improve the performance.

Keywords

Computer Aided Design; Program transformation; Vector machine.

1. INTRODUCTION

With the advent of the World Wide Web there has been immense demand for software tools and appliances for processing unstructured and semi-structured natural language text. Corporate intranets, enterprise portals, and ubiquitous e-mail, have created many challenges and opportunities in application areas such as information retrieval, electronic commerce, and knowledge management. On the supply side, the development of language technology to address such attendant problems as information overload and rapid globalization has been facilitated by two technical breakthroughs. The first is conceptual, and represents a new emphasis upon empirical approaches to language processing that rely more heavily upon corpus statistics than linguistic theory. The second is computational, and consists of more powerful, networked machines that are capable of processing

millions of documents and performing the billions of calculations that the statistical profiling of large corpora requires. This paper cite the research article of GuoDong Zhou et al. [13] in this connection. The work of Barbara Partee [2] presents a very good background reference text and covers many formal aspects of semantics and computation. Barbara Grosz, and Karen Soarck [1] have provided a collection of various classic papers in natural language processing.

A knowledge-based system used for extracting text-lines from mixed and overlapping text/graphics compound document images Original Research Article. An approach combining Natural language and knowledge-based systems by Jeremy Clare, Nick Ostler [15] in a context free grammar. The interaction of domain knowledge and linguistic structure in natural language processing can be seen in the research paper of Thomas C Rindfleisch et al [27]. A logic-based knowledge source system for natural language documents in data and knowledge Engineering has been developed by Skuce et al. [6]. David Warren and Fernando Pereira [7] have described the details of the CHAT-80 question answering system. M. Brady et al. [18] have focused on the Comprehension of Definite Anaphora. Anaphora in natural language processing and Management has been utilized in the work of Elizabeth DuRoss Liddy [8]. Knowledge based system and Information Processing & Management was also discussed by Tomek Strzalkowski [26]. Jose Perez-Carballo et al. [16] have discussed the working of Natural language information in computer systems.

Support vector machines (SVM) are currently used in a wide range of fields, from bioinformatics to astrophysics. Most existing software is written in C or C++, such as the award winning libsvm Chang et al. [4], which provides a robust and fast SVM implementation and produces state of the art results on most classification and regression problems. This paper can also cite the works of Meyer et al. [19], Joachims et al. [17], Collobert. R et al. [5], Gammerman et al. [11], Ruping et al. [22], and Guermeur [14], etc. to mention only a few. Many packages provide interfaces to MATLAB (The MathWorks 2005) (such as libsvm), and there are some native MATLAB toolboxes as well such as the SVM and Kernel Methods Matlab Toolbox whose references can be seen in the work of Canu et al. [3]. The research work of Gunn [12] for the MATLAB Support Vector Machine Toolbox and of Schwaighofer et al [23] for the SVM toolbox for Matlab can be cited in this context. .

In this paper, knowledge-based text extraction and identification method has been utilized to obtain the text-lines with different characteristics in each plane. The proposed system offers high flexibility and expandability by

merely updating new rules to cope with various types of real-life complex document. Experimental and comparative results prove the effectiveness of the proposed knowledge-based system and its advantages in extracting text-lines with a large variety of illumination levels, sizes and font styles from various types of mixed and overlapping text/graphics complex compound document images. This work was initiated by Yen-Lin Chen et al. [28] in text line extraction.

The automatic processing of written texts is being tackled by a variety of scientific disciplines. Within Computer Science the area of Natural Language Processing is deeply concerned with the problem of developing software systems that include language analysis functionalities to solve real problems and through this approach. This work can cite the paper of P Martínez Fernández et al [20] in this arena. Natural language processing (NLP) techniques substantially enhance most phases of the information system lifecycle, starting with requirements analysis, specification and validation, and going up to conflict resolution, result processing and presentation. The work of Elisabeth Métais [9] can be cited for application of NLP in Enhancing information systems management. Program transformations can be classified according to various criteria such as amount of automation; improvement achieved, and subject language as is evident from the works of Feather [10], Partsch [21], Smaragdakis et al. [24]. The transformation systems in dealing with above mentioned three criterions have been utilized as given under:

- Formulation of rule-based solutions to a wide range of transformation problems.
- Concise and reusable generalization of rules and strategies.
- Generation of efficient implementations for rule-based generalization.

2. INCLUSION OF THE TEXT INTO THE FRAMEWORK

First, a text query shall be sent directly to the search database (Software) (augmented by query markup, if it is available). In the next phase, the extractor shall pull text as well as markup out of retrieved data. With the use of semantic markup, extracted text may be filtered or translated in various ways before being used. Potentially useful filters include translation, summarization, trust verification, *etc.* Incorporation of extracted text into the query of a subsequent round of processing corresponds to blind relevance feedback. The framework then will provide a way to include both text and semantic markup as relevance feedback terms, even when the original query is homogeneous (bi-directional).

In the first step, an input sentence, hypothesized to contain errors, is first reduced to a "canonical form" devoid of articles, prepositions, and auxiliaries ("can," "would," "be," etc.). Furthermore, all nouns are reduced to their singular forms, and all verbs are reduced to their root forms. All of their alternative inflections are then inserted into the lattice in parallel. Insertions of articles, prepositions and auxiliaries are allowed at every position. This algorithm thus expands the sentence into a lattice of alternatives, as illustrated in Fig 1 also known as Vector machine.

3. SELECTIONS AND TESTING OF THE PROCESS

This paper shall demonstrate how a transformation based approach can be taken for tagging unknown words, by automatically learning cues to predict the most likely tag for words not seen in the training corpus. If the most likely tag for unknown words can be assigned with high accuracy, then the contextual rules can be used to improve accuracy, as described:-

1. Deleting the prefix (suffix) x , $|x| < 4$, results in a word (x is any string of length 1 to 4).
2. The first (last) (1, 2, 3, 4) character of the word are x .
3. Adding the character string x as prefix (suffix) results in a word ($|x| \sim 4$).
4. Word w ever appears immediately to the left (right) of the word.

4. STRUCTURING OF SENTENCES

Any sentence for particular objects and verbs defined in any voice shall be structured as elucidated in the table as fig 2 & fig 4

5. ERROR CORRECTION METHOD

The error correction system has three analysis functions as structured in figure 3:

Two notations have been used here. When one letter cannot be recognized uniquely, the users for the letter are enclosed in parentheses. A letter, which cannot be recognized at all, is expressed by a question mark. The Word analysis function searches the dictionary to find a grammatically and semantically valid user. When the Word analysis is unsatisfactory to resolve the ambiguities, the Syntax analysis is applied to them. The syntax analysis program refers the contextual information. In the sentences of patent gazettes, important words or key words are repeatedly used with anaphoric pronouns. This fact is a very important clue to find an anaphora or to guess the ambiguous letter in this system.

6. FOUNDATION OF THE PROGRAM

This work is an attempt to develop software based on computer tools, for the wrong verb or helping verb of a sentence from the text form without altering its sense. In the earlier work of Sumit Khulbe et al. [25], this paper utilized an organized process in which consecutive versions are obtained by employing changes to the syntax definition of a language and have taken conversion as a concept in traditional logic referring to a "type of immediate inference. In addition of following three software engineering tools, employed in earlier works (2009) and (2010):

1. Computer languages and specification formalisms.
2. Tool support. Software-Advanced Integrated Development Environments (IDEs)
3. Process of software development in order to be reusable and for the generation of many artifacts.

The hardware requirements of this project are- 20GB Hard Disk Drive (HDD), 256MB Random Access Memory (RAM), all the buses that are generally used and simple Intel mother board via chip. vb has been used as File Handling to help over

the several database files. The developed software requires only **50 MB** space in the unzipped category and can run efficiently even in 17'' Color Monitor. This project has used

Window XP Operating System to develop but can be run in any operating system.

7. FIGURES/CAPTIONS

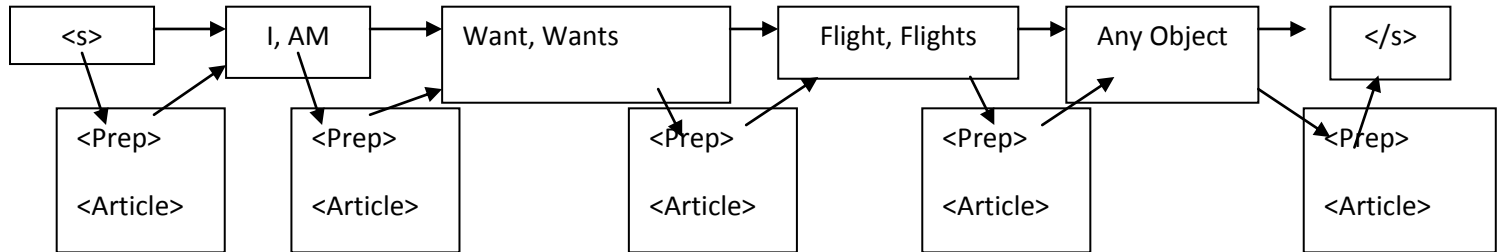


Fig 1: Vector machine

	→	1	2	3	4	5	6	7
		I	We	You	He	She	They	Noun
11		Am	Are	Are	Is	Is	Are	Is, are
22		Am	Are	Are	Is	Is	Are	Is, are
33		Have	Have	Have	Has	Has	Have	Has, have
44		Was	Were	Were	Was	Was	Were	Was, were
55		Was	Were	Were	Was	Was	Were	Was, were
66		Had	Had	Had	Had	Had	Had	Had
77		Shall	Will	Will	Will	Will	Will	Will
88		Shall have	Will have	Will have	Will have	Will have	Will have	Will have

Fig 2:

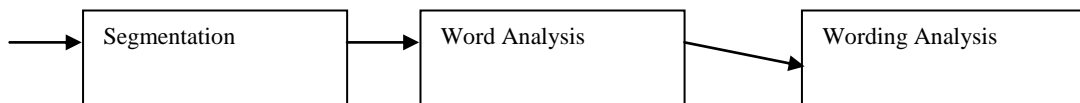


Fig 3:

8. COMPARISION TABLE

The superiority of the developed method can be depicted as under:

Dimension	Continuity	Discontinuity	Order
1.	Time event	Gaps	Out
2.	Large event	Large gaps	In
3.	System event	Change	In- Out

Fig 4:

9. DESCRIPTION OF THE PROGRAM

If hv = "empty" Or hv = "past" Then subject = subj Else s = Len(subj) - (Len(hv) + 1) subject = Left(subj, s) End If If subject = " i " Or subject = " I " Then su = Left(subject, Len(subject) - 2) Else su = subject End If Step 3: To check words that matches the clues and patterns. E.g. Jones was born in 1975; the matching words are Jones, born, 1975. The clues and patterns are all in MS-Access database. Step 4: To check proper noun - capital word for each starting word. Here are my halfway done codes with module by module. To count the verb from the database Public Function GetWordCount (ByVal InputString As String) As Integer Return

Split(System.Text.RegularExpressions.Regex.Replace(InputString, "\s+", Space(1))).Length End Function To eliminate the wrong verb and correct it

End If w = InStr(1, s, " we ") Or InStr(1, s, " We") If w > 1 Then psubject = "We" End If Y = InStr(1, s, " you ") Or InStr(1, s, " You ") If Y > 1 Then psubject = "You" End If t = InStr(1, s, " they ") Or InStr(1, s, " They ") If t > 1 Then psubject = "They" End If

This paper included all possible helping verbs in the database whose position can be searched out with the help of (Instr) function.

9. CONCLUSION

This paper have observed a fundamental conflict between linguistic processing strategies and knowledge-based reasoning Processing in adapting CAD to a specific application task and have confronted a tradeoff between the long term goals of designing a general purpose application Independent NL system and the specific requirements of the application task by a vector machine. This paper is able to satisfy the immediate demands of the application without changing the fundamental architecture of the system and have been able to remove the network ambiguities and extract the knowledgeable information in a more precise manner.

10. REFERENCES

- [1] Barbara Grosz, Karen Soarck Jones and Bonnie Lynn Webber [1986] Readings in Natural Language Processing available in AI SB library.
- [2] Barbara Partee, Alice ter Muelen and Robert Wall [1990], "Mathematical Methods in Linguistics".
- [3] Canu S, Grandvalet Y, Rakotomamonjy A (2003). "SVM and Kernel Methods MATLAB Toolbox." Perception Syst`emes et Information, INSA de Rouen, Rouen, France. URL <http://asi.insa-rouen.fr/~arakotom/toolbox/index>.
- [4] Chang CC, Lin CJ (2001). "libsvm: A Library for Support Vector Machines." URL <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [5] Collobert R, Bengio S, Mari`ethoz J (2002). "Torch: A Modular Machine Learning Software Library." URL <http://www.torch.ch/>.
- [6] D. Skuce, S. Matwin, B. Tauzovich, F. Oppacher, S. Szpakowicz [1990] A logic-based knowledge source system for natural language documents Original Research Article Data & Knowledge Engineering, Volume 1, Issue 3, pp. 201-231.
- [7] David Warren and Fernando Pereira [1982], Computational Linguistics, Vol. 8, Nos 3-4, pp 110-122, An Efficient Easily Adaptable System for Interpreting Natural Language Queries".
- [8] Elizabeth DuRoss Liddy [2000], Anaphora in natural language processing Volume 26, Issue 1, pp. 39-52.
- [9] Elisabeth Métais [2002] Enhancing information systems management with natural language processing techniques Data Knowledge Engineering, Volume 41, Issues 2-3, pp. 247-272.
- [10] Feather, M.S., 1987. A survey and classification of some program transformation approaches and techniques. In: Meertens, L.G.L.T. (Ed.), Program Specification and Transformation. In: IFIP, Elsevier Science Publishers, pp. 165-195.
- [11] Gammerman A, Bozanic N, Scholkopf B, Vovk V, Vapnik V, Bottou L, Smola A, Watkins C, LeCun Y, Saunders C, Stitson M, Weston J (2001). "Royal Holloway Support Vector Machines." URL <http://svm.dcs.rhnc.ac.uk/dist/index.shtml>.
- [12] Gunn SR (1998). "MATLAB Support Vector Machines." University of Southampton, Electronics and Computer Science, URL <http://www.isis.ecs.soton.ac.uk/resources/svminfo/>.
- [13] GuoDong Zhou, Min Zhang [2007], Extracting relation information from text documents by exploring various types of knowledge Original Information Processing & Management, Volume 43, Issue 4, pp. 969-982.
- [14] Guermeur Y (2004). "M-SVM." Lorraine Laboratory of IT Research and its Applications, URL <http://www.loria.fr/~guermeur/>.
- [15] Jeremy Clare, Nick Ostler [1990] An approach combining Natural language and knowledge-based systems Volume 26, Issue 2, pp 40-42.
- [16] Jose Perez-Carballo, Tomek Strzalkowski [2000] Natural language information retrieval: progress report Information Processing & Management, Volume 36, Issue 1, January 2000, Pages 155-178.
- [17] Joachims T (1999). "Making Large-scale SVM Learning Practical." In "Advances in Kernel Methods – Support Vector Learning," chapter 11. MIT Press. URL http://www-ai.cs.uni-dortmund.de/DOKUMENTE/joachims_99a.ps.gz.
- [18] M. Brady and R. Berwick, pp 331-371.[1983] Focusing in the Comprehension of Definite Anaphora.
- [19] Meyer D, Leisch F, Hornik K (2003). "The Support Vector Machine Under Test." Neurocomputing, Volume 55, pp. 169-186.
- [20] P Martínez Fernández, A.M García-Serrano [2000] The role of knowledge-based technology in language applications development Original Research Article Expert Systems with Applications, Volume 19, Issue 1, pp. 31-44.
- [21] Partsch, H., 1990. Specification and Transformation of Programs. A Formal Approach to Software Development. Springer-Verlag.

- [22] Ruping S (2004). “ ” mySVM – A Support Vector Machine.” University of Dortmund, Computer Science, URL <http://www-ai.cs.uni-dortmund.de/SOFTWARE/MYSVM/index.html>.
- [23] Schwaighofer A (2005). “SVM Toolbox for MATLAB.” Intelligent Data Analysis group (IDA), Fraunhofer FIRST, URL <http://ida.first.fraunhofer.de/~anton/software.html>.
- [24] Smaragdakis, Y., Batory, D., 2000. Application generators. In: Webster, J. (Ed.), Encyclopedia of Electrical and Electronics Engineering. John Wiley and Sons.
- [25] Sumit Khulbe, Richanshu Sharma and H.S.Dhami (2009), Mathematical formalism and computer program for tense conversion in English Grammar, International Transactions in Mathematical Sciences and Computer; July-December 2009, Volume 2, No. 2, pp. 07-317.
- [26] Tomek Strzalkowski [1996 Knowledge based system and Information Processing] Volume 31, Issue 3, pp. 397-417.
- [27] Thomas C Rindflesch, Marcelo Fiszman [2003] The interaction of domain knowledge and linguistic structure in natural language processing Volume 36, Issue 6, pp. 462-477.
- [28] Yen-Lin Chen, Zeng-Wei Hong, Cheng-Hung Chuang [2012], Expert Systems with Applications, Volume 39, Issue 1, pp. 494-507.