International Journal of Scientific Research Engineering & Technology (IJSRET)
Volume 1 Issue5 pp 313-318 August 2012 www.ijsret.org ISSN 2278 – 0882

A Graph Based Approach to Word Sense Disambiguation for Hindi Language

[1]Sandeep Kumar Vishwakarma, [2]Chanchal Kumar Vishwakarma
[1]Department of Computer Science , Aryabhatt College of Engineering and Technology,
Baghpat, India (Email: s_nitttr@yahoo.com)
[2]Department of Electronics and Communication Engineering, JSIMT, Shikohabad, India
(Email: chanchal_83@sify.com)

ABSTRACT

Hindi is the official language of the Republic of India. Hindi is the third most widely spoken language in the world (after English and Mandarin) an estimated 500-600 million peoples speaks the language. But, the language is making hindrances in the advantages of Information Technology revolution in India. So, there is the need of the adequate measures to perform natural language processing (NLP) through computer processing so that computer based system can be interacted by users through natural language like Hindi and handled by users who have knowledge of regional language. In computational linguistics, word sense disambiguation (WSD) is an open problem of natural language processing, which governs the process of identifying which sense of a word (i.e. meaning) is used in a sentence, when the word has multiple meanings. In this paper, we are concerned with graph-based algorithm for word sense disambiguation for Hindi language and finding the correct sense for given Hindi word. We introduce the graph based WSD algorithm which has few parameters using this algorithm we measures of graph connectivity the aim of identifying those best suited for WSD. We explore the multiple meanings of Hindi word with the help of Hindi Word net prepared by IIT Bombay.

***Keywords**: Introduction to WSD, Ambiguity for Humans and Computers, Hindi WorldNet, related work, approaches to WSD, worked done, WSD algorithm, result, and conclusion.*

I. INTRODUCTION TO WSD

In natural language processing, word sense disambiguation (WSD) is the problem of determining which "sense" (meaning) of a word is activated by the use of the word in a particular context, a process which appears to be largely unconscious in people. WSD is a natural classification problem: Given a word and its possible senses, as defined by a dictionary, classify an occurrence of the word in context into one or more of its sense classes. The features of the context (such as neighbouring words) provide the evidence for classification. Words can have different senses. Some words have multiple meanings. This is called polysemy. For example: bank can be a financial institute or a river shore. Sometimes two completely different words are spelled the same. For example: Can, can be used as model verb: You can do it, or as container: She brought a can of soda. This is called homonymy [1]. In computational linguistics, word sense disambiguation (WSD) is an open problem of natural language processing, which governs the process of identifying which sense of a word (i.e. meaning) is used in a sentence, when the word has multiple meanings. The solution to this problem impacts other computer-related writing, such as discourse, improving relevance of search engines, anaphora resolution, coherence, inference and others. Word Sense Disambiguation (WSD) is defined as the task of finding the correct sense of the word in a context. The task needs large amounts of word and word knowledge let us consider the word **स्वच्छ** in the following Hindi sentence.

आज हर व्यक्ति पर्यावरण की बात करता है, प्रदूषण से बचाव के उपाय सोचता है। व्यक्ति स्वच्छ और प्रदूषण-मुक्त पर्यावरण में रहने के अधिकारों के प्रति सजग होने लगा है और अपने दायित्वों को समझने लगा है। वर्तमान में विश्व ग्लोबल वार्मिंग के सवालों से जूझ रहा है।

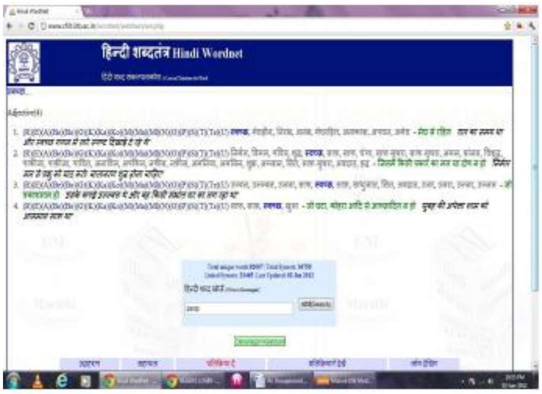

Figure I.1: Senses of **स्वच्छ** obtained from the Hindi Wordnet

In this particular case, sense 1 is the most appropriate one, though sense 2 and 3 too are relevant.

II. AMBIGUITY FOR HUMANS AND COMPUTERS

In our life most words have many possible meanings; this is known as polysemy. This problem is encountered not only by humans but also by computers.

Ambiguity for Humans

Ambiguity is rarely a problem for humans in their day to day communication, except in extreme cases e.g. Ambiguity as seen in newspapers which won't be resolved by Humans are as

1. महिला एक छाता के साथ आदमी को मारा. (महिला एक छाता उपयोग करके आदमी को मारा या वह एक आदमी जो एक छाता ले जा रहा है उसे मारा)
2. आदमी दूरबीन के साथ लड़के को देखा.
3. वे फ्रेंच, जर्मन और जापानी के शिक्षकों के लिए देख रहे हैं (हर एक भाषा या सभी तीन भाषाओं को पढ़ाने के लिए देख रहे हैं

Ambiguity for Computer

Ambiguity is rarely a problem for computer in their day to day communication, except in extreme cases e.g.

1. आम आम आदमी की परिधि से बाहर है (यहाँ आम के दो मतलब हैः फल और सामान्य आदमी)
2. *महँगाई से हर वर्ग के लोग परेशान हैं*"(Here *वर्ग* is interpreted as class)
3. *यह पाँच सेंटीमीटर का वर्ग है* (Here *वर्ग* is interpreted as square shaped figure')

III. APPROACHES OF WSD

As in all natural language processing, there are two main approaches to WSD – deep approaches and shallow approaches.

Deep Approaches

Deep approaches presume access to a comprehensive body of world knowledge. Knowledge, such as *"दया एक सात्विक भावना है"* or *"दया भुवनेश्वर के पास से बहती है"*, here *दया* is ambiguated by two meaning 'compassion' and 'name of river'.

Then Deep approaches used to determine in which sense the word is used. These approaches are notvery successful in practice, mainly because such a body of knowledge does not exist in a computer readable format, outside of very limited domains. However, if such knowledge did exist, then deep approaches would be much more accurate than the shallow approaches [2].

There are two types of Deep approach of Word Sense Disambiguation are:

- Selectional restriction'- based approaches
- Approaches based on general reasoning with 'world knowledge'

Shallow Approaches

Shallow approaches don't try to understand the text. They just consider the surrounding words, using information such as: if *दया* has a word *भावना* or *दुख* nearby, it probably in the sense of 'compassion'; if *दया* has a world *बहती* or *भुवनेश्वर* nearby, it probably in the sense of 'river'.

These rules can be automatically derived by the computer, using a training corpus of words tagged with their word senses. This approach, while theoretically not as powerful as deep approaches, gives superior results in practice, due to the computer's limited world knowledge. Our paper is base on the Shallow approach methodology.

The different types of Shallow approaches of WSD are:

- Dictionary-based approaches.
- Machine learning approaches
- Supervised methods
- Semi-supervised
- Unsupervised methods
- Hybrid approach

IV. HINDI WORDNET

Pushpak Bhattacharyya [3], the Hindi WordNet is a system for bringing together different lexical and semantic relations between the Hindi words. It organizes the lexical information in terms of word meanings and can be termed as a lexicon based on psycholinguistic principles. The design of the Hindi WordNet is inspired by the famous English WordNet.

In the Hindi WordNet the words are grouped together according to their similarity of meanings. Two words that can be interchanged in a context are synonymous in that context. For each word there is a synonym set, or synset, in the Hindi WordNet, representing one lexical concept. This is done to remove ambiguity in cases where a single word has multiple meanings. Synsets are the basic building blocks of WordNet. The Hindi

WordNet deals with the content words, or open class category of words. Thus, the Hindi WordNet contains the following category of words- Noun, Verb, Adjective and Adverb.

Each entry in the Hindi WordNet consists of following elements:

1. Synset: It is a set of synonymous words. For example, “विदयालय. पाठशाला, , स्कूल” (vidyaalay, paathshaalaa, skuul) represents the concept of school as *an educational institution*. The words in the synset are arranged according to the frequency of usage.
2. Gloss: It describes the concept. It consists of two parts:

Text definition: It explains the concept denoted by the synset. For example, “वह स्थान जहाँ प्राथमिक या माध्यमिक स्तर की औपचारिक शिक्षा दी जाती है” (vah sthaan jahaaM praathamik yaa maadhyamik star kii aupacaarik sikshaa dii jaatii hai) explains the concept of school as an educational institution.

Example sentence: It gives the usage of the words in the sentence. Generally, the words in a synset are replaceable in the sentence. For example," इस विद्यालय में पहली से पाँचवीं तक की शिक्षा दी जाती है" (is vidyaalay me pahalii se pancvii tak kii shikshaa dii jaatii hai) gives the usage for the words in the synset representing school as an educational institution. Each synset is mapped into some place in the ontology. A synset may have multiple parents. The ontology for the synset representing the concept school is shown in figure.

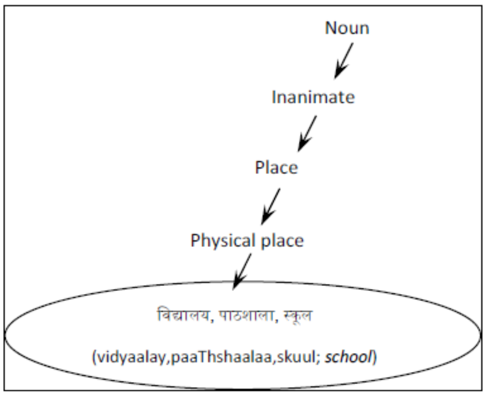

Figure 2. Ontology for the synset of school

Current Status of Hindi WordNet is still under construction. In the version 1.0 we have attempted to cover all the common concepts in Hindi. The present status is as follows:

Total unique words: 93584
Total Synsets: 37391
Linked Synsets: 24319
Last Updated: 14 Jul 2012

V. RELATED WORK

Manish Sinha, Mahesh Kumar Reddy .R, Pushpak Bhattacharyya , Prabhakar Pandey and Laxmi Kashyap[4], “Hindi Word Sense Disambiguation” that was the first attempt for an Indian language at automatic WSD. The use of the Wordnet for Hindi developed at IIT Bombay, which is a highly important lexical knowledge base for Hindi. The main idea is to compare the context of the word in a sentence with the contexts constructed from the Wordnet and chooses the winner. The output of the system is a particular synset number designating the sense of the word. The mentioned Wordnet contexts are built from the semantic relations and glosses, using the Application Programming Interface created around the lexical data. The evaluation has been done on the Hindi corpora provided by the Central Institute of Indian Languages and the results are encouraging. Currently the system disambiguates nouns. Work is on for other parts of speech too.

Neetu Mishra, Shashi Yadav and Tanveer J. Siddiqui[5], “An Unsupervised Approach to Hindi Word Sense Disambiguation” The algorithm learns a decision list using untagged instances. Some seed instances are provided manually. Stemming has been applied and stop words have been removed from the context. The list is then used for annotating an ambiguous word with its correct sense in a given context. The evaluation has been made on 20 ambiguous words with multiple senses as defined in Hindi WordNet.

Rohan[6], “Word Sense Disambiguation for Hindi Language” attempt to resolve the ambiguity by making the comparisons between the different senses of the word in the sentence with the words present in the synset form of the WordNet and the information related to these words in the form of parts-of-speech. This WordNet is considered to be the most important resource available to researchers in computational linguistics, text analysis and many related areas.

Avneet Kaur[7], “Development of an Approach for Disambiguating Ambiguous Hindi postposition” They have chosen to develop an efficient algorithm for disambiguating ambiguous postpositions present in the Hindi language. They are taking this problem with the case study of existing HindiPunjabi Machine Translation System. Thus the disambiguation will be done from the machine translation point of view. This is mainly used for removing the ambiguity from the corpus. N-gram algorithm is used for developing the Hindi postpositions. N-gram algorithm is used for extracting the words from the corpus.

Ripul Gupta [8], “Speech Recognition for Hindi” Speech interface to computer is the next big step that computer science needs to take for general users. Speech recognition will play a important role in taking technology to them. The need is not only for speech interface, but speech interface in local languages. His goal is to create speech recognition software that can recognise Hindi words. That report takes a brief look at the basic building block of a speech recognition engine. That talks about implementation of different modules. Sound Recorder, Feature Extractor and HMM training and Recogniser modules have been described in details. The results of the experiments that were conducted are also provided. The report ends with a conclusion and Future plan.

Ravi Sinha and Rada Mihalcea[9] “Unsupervised Graph-based Word Sense Disambiguation Using Measures of Word Semantic Similarity” that paper describes an unsupervised graph-based method for word sense disambiguation, and presents comparative evaluations using several measures of word semantic similarity and several algorithms for graph centrality. The results indicate that the right combination of similarity metrics and graph centrality algorithms can lead to a performance competing with the state-of-the-art in unsupervised word sense disambiguation, as measured on standard data sets.

Siva Reddy, Abhilash Inumella, Rajeev Sangal, Soma Paul[10], “All Words Unsupervised Semantic Category Labeling for Hindi” they use the ontological categories defined in Hindi Wordnet as semantic category inventories. In this paper they present two unsupervised approaches namely Flat Semantic Category Labeler (FSCL) and Hierarchical Semantic Category Labeler (HSCL). The former method treats semantic categories as a flat list, whereas the latter one exploits the hierarchy among the semantic categories in a top down manner. Further their methods use simple probabilistic models, using which the category labelling becomes a simple table look up with little extra computation and thus opening the possibility of its use in real-time interactive systems.

R. Mahesh K. Sinha,[11] “ Learning Disambiguation of Hindi Morpheme 'vaalaa' with a Sparse Corpus” The Hindi morpheme ‘vaalaa’ is very widely used as a suffix and also as a separate word. The common usage of this suffix is to denote an activity or profession of a person. This form of the usage has been borrowed in English with the spelling of ‘wallah’. However, it has a large number of other interpretations depending upon the context in which it is used. That paper presents an account of different senses in which this morpheme is used in Hindi and presents a strategy for learning their disambiguation based on contextual features with sparse data using a semi-supervised method. They present a new technique of unifying learned instances using supervised training with limited data and computing matching index and bootstrapping the training set to deal with corpus sparseness. This study finds application in machine translation, information retrieval, text understanding and text summarization.

Parul Rastogi andDr. S.K. Dwivedi[12], “Performance comparison of Word Sense Disambiguation (WSD) Algorithm on Hindi Language Supporting Search Engines”, The major population of India use Hindi as a first language. The Hindi language web information retrieval is not in a satisfactory condition. Besides the other technical setbacks, the Hindi language search engines face the problem of sense ambiguity. Their WSD method is based on Highest Sense Count (HSC). That works well with Google. The objective of that paper is comparative analysis of the WSD algorithm results on the three Hindi language search engines-Google, Raftaar and Guruji. They have taken a test sample of 100 queries to check the performance level of the WSD algorithm on various search engines.

Mitesh M. Khapra, Pushpak Bhattacharyya, Shashank Chauhan and Soumya Nair,[13] “Domain Specific Iterative Word Sense Disambiguation in a Multilingual Setting”, they work on Domain Specific Iterative Word Sense Disambiguation (WSD) for nouns, adjectives and adverbs in the trilingual setting of English, Hindi and Marathi The methodology proposed relies on dominant senses of words in specified domains. They combine corpus biases for senses along with information in wordnet graph structure to arrive at the sense decisions. To the best of our knowledge, that is the first attempt at a large scale multilingual WSD involving Indian languages and English.

VI. WSD ALGORITHM

Word Ambiguity is one of the problems which have been a great challenge for computational linguistics. In general, people are unaware of the ambiguities in the language because they have very good Memory, thinking, acting, reasonable, and six sense. On this level they use context and their knowledge of the world. But computer systems don’t have this knowledge, and consequently don’t do a good job of making use of the context.

In this paper, we will focus on developing a method used to resolve semantic ambiguity for Hindi languages. In

fact, some Hindi word has more than one meaning. For example, consider the word.
Consider the word: शाखा

It can refer 5 meanings which is obtained from the Hindi WordNet

1. बच्चे आम की शाखाओं पर झूल रहे हैं"

Here the meaning of शाखा is branches of a tree.

2. वह शैव सम्प्रदाय का अनुयायी है"

Here the meaning of शाखा is exclusive system of religious beliefs and practices

3. जैन धर्म के अंतर्गत दो शाखाएँ हैं- दिगंबर और श्वेतांबर"

Here the meaning of शाखा is a group of nations having common interests.

4. इस संस्था के पाँच अंग हैं"

Here the meaning of शाखा is an administrative division of some larger

5. "हर बड़ी नदी की कई शाखाएँ होती हैं"

Here the meaning of शाखा is a stream.

In this paper, we describe a graph-based algorithm for Hindi WSD. The algorithm proceeds incrementally on the sentence-by- sentence basis. The algorithm annotates all the words in a text by exploiting similarities identified among word senses, and using centrality algorithms applied on the graphs encoding these sense dependencies. This paper provides a comparative evaluation of several measures of word semantic similarity using a graphical framework. Specifically, we experiment with Depth-first- search. The Following steps in our approach to graph based WSD.

Graph Construction Process

1. Our disambiguation algorithm precede sentence-by-sentence basis.
2. Initially we construct a graph G= (V, E) for each target sentence σ which we include from the graph of reference lexicon.
3. We assume that the sentences in hindi language are part of speech tagged. So our algorithm considers context word only.
4. In the graph node represent word sense and edge represent semantic relation.
5. With the help of DFS, and Hindi WordNet we construct the final graph.

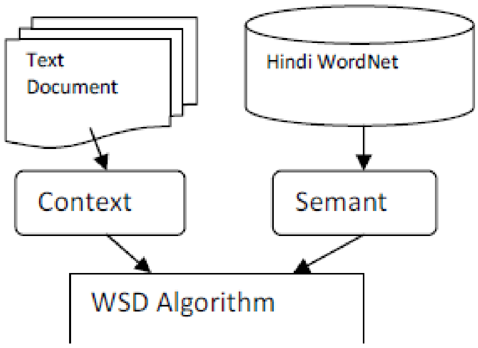

Figure 3. System architecture

Graph Connectivity Measures

In this section, we described the measure of graph connectivity as fallowing way.

1. First we calculate the distance function d(u,v), which used by some of measures as:

$$d(d,u)=\begin{cases}\text{Length of shortest path} & \text{if } u \sim v\\ \infty, & \text{otherwise,}\end{cases}$$

Where u ~ v indicates the existence path from u to v.

2. In second step we calculate two measure namely
   a) Local measure- determine the degree of single vertex in a graph
   b) Global measure – global connectivity measures are concern with the structure of graph as a whole rather than individual.
3. Finally we measure the time complexity of WSD algorithm.

VII. RESULTS

The algorithm was tested on a sample hindi corpus. For extracting Nouns, all words that have a valid Noun Synset in WordNet were extracted. Out of this set, those that had been incorrectly chosen as Nouns were then manually removed.

For the purpose of Clustering, a Hierarchical Agglomerative Clustering algorithm was used with the distance between 2 clusters being

- Shortest Distance between any 2 synsets from each cluster
- Average Distance between all synsets from each cluster

To save on computation time during the Polysem disambiguation phase in our algorithm, instead of assigning the synset(and a cluster) to the closest Polysem and then recomputing the distances for the remaining Polysems, half of the unclustered Polysems that were closest to the existing clusters were assigned a synset(and a cluster) in one iteration of the algorithm. Then, the distances for the remaining Polysems were recomputed and the algorithm is iterated.

Running the algorithm assigned a synset to every Noun in the sample text. For each of these Nouns, the assigned synset was manually labelled as Right/Wrong after seeing all possible synsets for the Noun. The Results are

No. Of Monosems in the Text = 247
No. Of Polysems in the Text = 666
Total No. Of Nouns in the Text = 913
No. Of Correctly Assigned Synsets = 595
Accuracy Obtained = 65.17%

VIII. CONCLUSION AND FUTURE WORK

In this paper, we described a graph-based word sense disambiguation algorithm for Hindi language, which combines Lesk semantic similarity measures and Indegree algorithms for graph centrality. To our knowledge, no attempt has been made in the past to address the problem of word sense disambiguation by comparatively evaluating measures of word similarity in a graph theoretical framework for Hindi language.

There are many possible extensions of this work that can be undertaken in further research. Some of them are listed below.

1. In this paper, we have used the database of text files saved from Hindi WordNet prepared by IIT, Bombay but in future, the database for Hindi language's WSD can use the database prepared for Hindi WordNet directly.
2. The accuracy of the graph base algorithm could be checked on other languages.
3. For the semantic similarity other similarity method can be use

For graph centrality other algorithm can use such as: BFS.

REFERENCES

1. Esha Palta, "Word Sense Disambiguation," M.Tech thesis, Indian Institute of Technology, Mumbai, CSE dept., India,2006.
2. "Word Sense Disambiguation",2009. http://en.wikipedia.org/wiki/Wordsense_ disambiguation#Approaches_and_methods
3. Dr. Pushpak Bhattacharyya, "Hindi WordNet Data and Associated Software License Agreement", Indian Institute of Technology, Mumbai, CSE dept., Tchnical Report 2006.
4. Manish Sinha, Mahesh Kumar Reddy, R Pushpak Bhattacharyya, Prabhakar Pandey and Laxmi Kashyap, "Hindi Word Sense Disambiguation", *Indian Institute of Technology Bombay, Department of Computer Science and Engineering Mumbai*, 2008.
5. Neetu Mishra, Shashi Yadav and Tanveer J. Siddiqui, "An Unsupervised Approach to Hindi Word Sense Disambiguation," *IndianInstitute of Information Technology, Allahabad. UP, India*, 2009.
6. Rohan, "Word Sense Disambiguation For Hindi language" *Thapar University Patiyala*, CSE Dept., India, 2007.
7. Avneet Kaur, "Development of an Approach for Disambiguating Ambiguous Hindi postposition," *International Journal of Computer Applications (0975 – 8887)*, vol.5, no.9, August 2010.
8. Ripul Gupta, "Speech Recognition for Hindi," M.Tech. thesis Indian Institute of Technology, Mumbai, CSE dept., India,2007.
9. Ravi Sinha and Rada Mihalcea, "Unsupervised Graph-based Word Sense Disambiguation Using Measures of Word Semantic Similarity," *IEEE International Conference on Semantic Computing*, pp. 363 – 369, Sept. 2007.
10. Siva Reddy, Abhilash Inumella, Rajeev Sangal, Soma Paul, "All Words Unsupervised Semantic Category Labeling for Hindi" *Proceedings of the International Conference RANLP, Borovets, Bulgaria*, pages 365–369, September 2009.
11. R. Mahesh K. Sinha, "Learning Disambiguation of Hindi Morpheme 'vaalaa' with a Sparse Corpus," *International Conference on Machine Learning and Applications*, pp. 653 – 657, December 2009.
12. Parul Rastogi and Dr. S.K. Dwivedi, "Performance comparison of Word Sense Disambiguation (WSD) Algorithm on Hindi Language Supporting Search Engines", *International Journal of Computer Science Issues*, vol. 8, issue.2, March 2011.
13. Mitesh M. Khapra, Pushpak Bhattacharyya, Shashank Chauhan and Soumya Nair, "Domain Specific Iterative Word Sense Disambiguation in a Multilingual Setting", *Proc. of ICON-2008: 6th International Conference on Natural Language Processing Macmillan Publishers, India*, 2008.