Word Sense Disambiguation for Hindi Language

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering
in
Computer Science & Engineering

Thapar University, Patiala

By:
Rohan Sharma
(80832020)

Under the supervision of:
Mr. Parteek Bhatia
Senior Lecturer, CSED

JULY 2008

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147004

Certificate

I hereby certify that the work which is being presented in the thesis report entitled, **'Word Sense Disambiguation for Hindi Language',** submitted by me in partial fulfillment of the requirements for the award of degree of Master of Engineering in Computer Science and Engineering submitted in the Department of Computer Science and Engineering of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of Mr. Parteek Bhatia and refers other researcher's works which are duly listed in the reference section.

The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.

(**Rohan Sharma**)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

(Mr. Parteek Bhatia)
Senior Lecturer
Department of Computer Science & Engineering
Thapar University
Patiala

Countersigned by

(Dr. SEEMA BAWA)
Professor & Head
Computer Science & Engineering, Department
Thapar University
Patiala

(Dr. R.K.SHARMA)
Dean(Academic Affairs)
Thapar University,
Patiala.

Acknowledgment

I express my sincere and deep gratitude to my guide Mr. Parteek Bhatia, Senior Lecturer in Department of Computer Science & Engineering, for the invaluable guidance, support and encouragement. He provided me all resource and guidance throughout thesis work.

I am thankful to Dr. (Mrs.) Seema Bawa, Head of Department of Computer Science & Engineering, Thapar University, Patiala, for providing me adequate environment, and facility for carrying thesis work.

I would like to like to say thanks from deep inside my heart to my Shri Hanumanji who had helped me at every moment of my life and is constantly guiding me and strengthening me. He is with me all the time. He is helping me through my friends, my teachers, my parents, my relatives and every other person because I know Shri Ram is in every person's heart and wherever is Shri Ram, my Hanumanji is also there to love me. After this thesis, I constantly pray to Him, please give me an opportunity to immerse my life in spiritual practices and help me to improve the conditions of common man and my country. I want nothing in this world except it as Adi Shankaracharya said, "Brahman is the only truth, the world is unreal".

Rohan Sharma

(80632020)

.

Abstract

Hindi is a national language of India, spoken by 500 million people and ranking 4th by majority spoken in the world. 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. So, Language Translator is important tool to resolve this problem. Word Sense Disambiguation (WSD) is an important concept that is to be evaluated for performing machine translation and a tool is needed to perform disambiguation so that computers would be able to interpret a word in its proper sense according to its context.

Word Sense Disambiguation (WSD) is the process of identifying which sense of a word is used in a given sentence. A word can have a number of senses, which is termed an ambiguity. Something is ambiguous when it can be understood in two or more possible ways or when it has more than one meaning. This word sense disambiguation is an 'intermediate task', which is not an end in itself, but rather is necessary at one level or another to accomplish most natural language processing tasks. In this way, Word Sense Disambiguation (WSD) is the problem of selecting a sense for a word from a set of predefined possibilities. Here the sense inventory usually comes from a dictionary or thesaurus to determine these different possibilities.

In this thesis work, the different approaches of Word Sense Disambiguation (WSD) like knowledge based approaches, machine learning based approaches and hybrid based approaches are discussed, and later the problem of disambiguation is being tried to solve by using Hindi WordNet developed at IIT, Bombay containing different words and their sets of synonyms called synsets. By the help of the words in these synsets, we make an 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.

Contents

List of Tables

List of Figures

Chapter 1 Introduction

1.1 Introduction of Word Sense Disambiguation

Lexical semantics begins with recognition that a word is a conventional association between a lexicalized concept and an utterance that plays a syntactic role. This lexical semantics is understood clearly by proper disambiguation of words. This disambiguation is achieved by one of the fields of natural language processing known as Word Sense Disambiguation.

The automatic disambiguation of word senses has been an interest and concern since the earliest days of computer treatment of language in the 1950's. Sense disambiguation is an 'intermediate task' which is not an end in itself, but rather is necessary at one level or another to accomplish most natural language processing tasks. It is obviously essential for language understanding applications such as message understanding, man-machine communication, *etc.*; it is at least helpful, and in some instances required, for applications whose aim is not language understanding. Something is ambiguous when it can be understood in two or more possible ways or when it has more than one meaning. Lexical semantic ambiguity occurs when a single word is associated with multiple senses [1].

The problem of word sense disambiguation has been described as AI-complete, that is, a problem which can be solved only by first resolving all the difficult problems in the artificial intelligence (AI), such as the representation of common sense and encyclopedic knowledge. The inherent difficulty of sense disambiguation was a central point in Bar-Hillel's treatise on machine translation [2], where he asserted that he saw no means by which the sense of the word *pen* in the sentence-'The box is in the pen' could be determined automatically. Bar-Hillel's argument laid the groundwork for the ALPAC report [3] which is generally regarded as the direct cause for the abandonment of most research on machine translation in the early 1960's [1].

However, at about the same time considerable progress was being made in the area of knowledge representation, especially the emergence of semantic networks, which were

immediately applied to sense disambiguation. Work on Word Sense Disambiguation (WSD) continued throughout the next two decades in the framework of the AI-based natural language understanding research, as well as in the fields of content analysis, stylistic and literary analysis, and information retrieval. Slowly, attempts to automatically disambiguate word senses have multiplied, due, like much other similar activity that are happening in the field of the computational linguistics, to the availability of large amounts of machine readable text and the corresponding development of statistical methods to identify and apply information about regularities in this data. The other problems amenable to these methods, such as part of speech disambiguation and alignment of parallel translations, have been fairly thoroughly addressed, the problem of word sense disambiguation has taken center stage, and it is frequently cited as one of the most important problems in natural language processing research today.

1.2 Word Sense Disambiguation and Word Sense Discrimination

These are two different terms -

1. **Word sense disambiguation**- It is the problem of selecting a sense for a word from a set of predefined possibilities. Here the sense inventory usually comes from a dictionary or thesaurus. The methods that it uses are Knowledge intensive methods, supervised learning, and (sometimes) Bootstrapping approaches.
2. **Word sense discrimination**- It is the problem of dividing the usages of a word into different meanings, without regard to any particular existing sense inventory. The techniques that it employed are Unsupervised techniques

In our thesis, we are making the emphasis on Word Sense Disambiguation.

1.3 Ambiguity for Humans and Computers

In our day to day life, most words have many possible meanings; this is known as polysemy [6]. This problem is encountered not only by humans but also by computers.

1.3.1 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 computers are as –

कर्मचारियों ने मौत के बाद काम करने से इन्कार किया

बूढ़े दरिया का दिल

बच्चों को बिस्कुट बनाते शामिल करे

डीज़ल ने ओस्कर जीता

1.3.2 Ambiguity for Computers

A computer program has no basis for knowing which one is appropriate, even if it is obvious to a human *e.g.* Ambiguity which won't be resolved by computers is –

वह आम खा रहा है ।

आम आम आदमी की परिधि से परे हो गया है ।

Here in the previous sentence **'आम'** can have two different meaning 'Mango' and 'common person'.

महँगाई से हर वर्ग के लोग परेशान हैं ।

Here **'वर्ग'** is interpreted as 'class'.

वर्ग की परिधि मे काम करे।

Here **'वर्ग'** is interpreted as 'one type of people or people of same locality'.

गाँधी जी एक उच्च वर्ग के नेता थे ।

Here **'वर्ग'** is interpreted as 'classification on the basis of capability or duty'.

हिन्दी व्यंजन कवर्ग, चवर्ग, टवर्ग आदि वर्गों में विभाजित है ।

Here **'वर्ग'** is interpreted as 'those letters which are pronounced from same part of mouth'.

सात का वर्ग उनचास होता है।

Here **'वर्ग'** is interpreted as 'square of the number'.

यह पाँच सेंटीमीटर का वर्ग है।

Here **'वर्ग'** is interpreted as 'square shaped figure'.

अध्यापक ने विद्यार्थियों को एक काँच का वर्ग दिखाया ।

Here **'वर्ग'** is interpreted as 'the thing whose shape is similar to square'.

1.4 Role of Word Sense Disambiguation

The applications where word sense disambiguation is used now-a-days is as follows-

- **Machine Translation** – The sense disambiguation is essential for the proper translation of words such as the Hindi '**सोना'**, which, depending on the context, can be translated as Gold, Sleep, Sona(the name) *etc.* [4]

  For example-

  सोना सोना चाहता है ।

  It can be translated as-

  Sona wants gold.

  Or

  Sona wants to sleep.

  Or

  Gold wants to sleep.

  Or

  Sleep wants gold.

  Or

  Gold wants Sona *etc.*

  So in this way there is ambiguity for '**सोना'** because it is being interpreted of as gold means '**सोना'** or as sleep means '**नींद'** or as Sona (the name) means '**सोना'**.

- **Information retrieval and hypertext navigation -** When searching for specific keywords, it is desirable to eliminate occurrences in documents where the word or words are used in an inappropriate sense; for example, when searching for judicial

references, it is desirable to eliminate documents containing the word '**कोर्ट**' associated with royalty, rather than with law.

- **Content and thematic analysis -** A common approach to content and thematic analysis is to analyze the distribution of pre-defined categories of words- *i.e.*, words indicative of a given concept, idea, theme, *etc.* across a text. The need for sense disambiguation in such analysis has long been recognized in order to include only those instances of a word in its proper sense.

- **Grammatical analysis –** The sense disambiguation is useful for part of speech tagging-for example, in the Hindi sentence '**खाना किताबों के भार से झुका जा रहा है'** [The shelf is bending under (the weight of) the books], it is necessary to disambiguate the sense of '**खाना'** (which can mean food as well as book-shelf) to properly tag it as a noun. Sense disambiguation is also necessary for certain syntactic analyses, such as prepositional phrase attachment and in general restricts the space of competing parses.

- **Speech processing -** The sense disambiguation is required for correct phonetization of words in speech synthesis, *e.g.*, the word '**फल'** in **'उसने फलों को खाया ।'** or in '**मेरा परीक्षा-फल आ गया।'.**

- **Text processing -** The sense disambiguation is necessary for spelling correction *e.g.* **'मान'** can be disambiguated as weight of something or the honour.

1.5 Approaches of Word Sense Disambiguation

There are two main approaches to WSD — deep (but brittle) approaches and shallow (but robust) approaches [5].

1. **Deep Approaches-** Deep approaches presume access to a comprehensive body of world knowledge. These approaches are not very successful in practice, mainly because such a body of knowledge does not exist in computer-readable format outside of very limited domains. But if such knowledge did exist, they would be much more accurate than the shallow approaches.

   However, there is a long tradition in Computational Linguistics of trying such approaches in terms of coded knowledge, and in some cases it is hard to say clearly whether the knowledge involved is linguistics or world knowledge. The first attempt was that by Margaret Masterman and her colleagues at Cambridge Language Research Unit in England in the 1950s. This used as data a punched-card version of Roget's Thesaurus and its numbered 'heads' as indicators of topics and looked for their repetitions in text, using a set intersection algorithm: it was not very successful [7] but had strong relationships to later work, especially Yarowsky's machine learning optimization of a thesaurus method in the 1990s.

   The different types of Deep approach of Word Sense Disambiguation [5] are-

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

2. **Shallow Approaches-** Shallow approaches don't try to understand the text. They just consider the surrounding words, using information like "if '**आम**' has words '**फल**' or '**खाना**' nearby, it probably is in the sense of mango; if '**आम**' has the words '**आदमी**' or '**जनता**' nearby, it is probably in the person's sense." 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 computers' limited world knowledge. It can, though, be confused by sentences like **'आम एक फल है पर आम**

आदमी की परिधि से परे हो गया है।', which contains the word '**आम'** near both '**फल'** and '**आदमी**'. Our thesis is based on the shallow approach methodology.

The different types of Shallow approaches of WSD [5] are-

- Dictionary-based approaches.
- Machine learning approaches-It can be further shown of using several different methods as –
    - Supervised methods (*i.e.* using annotated corpora)
    - Unsupervised methods (*i.e.* using raw text)
- Various combinations of methods, also known as Hybrid approach.

These approaches normally work by defining a window of ***n*** content words around each word to be disambiguated in the corpus, and statistically analyzing those ***n*** surrounding words.

1.6 Comparison of WSD and Part-of-Speech (POS) tagging

It is instructive to compare the word sense disambiguation problem with the problem of part-of-speech tagging (POS tagging). Both involve disambiguating or tagging with words, be it with senses or parts of speech. However, algorithms used for one do not tend to work well for the other, mainly because the part of speech of a word is primarily determined by the immediately adjacent one to three words, whereas the sense of a word may be determined by words further away. The success rate for part-of-speech tagging algorithms is at present much higher than that for WSD; state-of-the art being around 95% accuracy or better, as compared to less than 80% accuracy in word sense disambiguation with supervised learning [8]. These figures are typical for English. The POS tagging for Hindi by using maximum entropy approach is 82% [9] while the percentage of accuracy of Hindi's WSD vary as its disambiguation depends on different types of proper nouns, still its accuracy is highest as 74% [10] .

Another aspect of word sense disambiguation that differentiates it from part-of-speech tagging is the availability of training data. While it is relatively easy to assign parts of speech to text, training people to tag senses is far more difficult [11]. While users can memorize all of the possible parts of speech a word can take, it impossible for individuals to memorize all of the senses a word can take. Thus, many word sense disambiguation algorithms use semi-supervised learning, which allows both labeled and unlabeled data. The Yarowsky algorithm was an early example of such an algorithm.

1.7 Web and Word Sense Disambiguation

The Web has become a source of data for NLP in general, and word sense disambiguation is no exception. Web can find hundreds/thousands of instances of a particular target word just by searching. Search Engines for English are as: Alta Vista, Google, and Yahoo. Now-a-days Hindi Web search engines are also knocking their presence, for example Google's search engine in Hindi.

The Web can search as a good source of information for selecting or verifying collocations and other kinds of association *e.g.*

- strong tea : 13,000 hits
- powerful tea : 428 hits
- sparkling tea : 376 hits

One can find sets of related words from the Web. Let us consider a web link of Google as http://labs.google.com/sets by giving input of two or three words, it will return a set of words it believes are related *e.g.* if we give Google sets input: bank, credit and the Google outputs we got as: bank, credit, stock, full, investment, invoicing, overheads, cash low, administration, produce service, grants, overdue notices.

Another example to give as input to find out the great source of info about names or current events is as Google Sets Input: Nixon, Carter and the output is as: Carter, Nixon, Reagan, Ford, Bush, Eisenhower, Kennedy, and Johnson.

Chapter 2 Review of Literature

2.1 Steps for handling of WSD

Word Sense Disambiguation (WSD) involves the association of a given word in a text or discourse with a definition or meaning (sense) which is distinguishable from other meanings potentially attributable to that word. The task therefore necessarily involves two steps:

Step 1: The determination of all the different senses for every word relevant (at least) to the text or discourse under consideration;

Step 2: A means to assign each occurrence of a word to the appropriate sense.

2.1.1 Step 1: Determination of different senses

WSD relies on pre-defined senses for step (1), including:

- a list of senses such as those found in everyday dictionaries;
- a group of features, categories, or associated words (*e.g.*, synonyms, as in a thesaurus);
- an entry in a transfer dictionary which includes translations in another language; *etc*.

The precise definition of a sense is, however, a matter of considerable debate within the community. The variety of approaches to defining senses has raised recent concern about the comparability of much WSD work, and given the difficulty of the problem of sense definition, no definitive solution is likely to be found soon. However, since the earliest days of WSD work there has been general agreement that the problems of morpho-syntactic disambiguation and sense disambiguation can be disentangled.

That is, for homographs with different parts of speech (*e.g.*, '**खाना**' as a verb and noun), morpho-syntactic disambiguation accomplishes sense disambiguation, and therefore (especially since the development of reliable part-of-speech taggers), WSD work has since focused largely on distinguishing senses among homographs belonging to the same syntactic category.

2.1.2 Step 2: Assignment of Senses to words

In it, the assignment of words to senses is accomplished by reliance on two major sources of information:

- Context of the word to be disambiguated, in the broad sense: this includes information contained within the text or discourse in which the word appears, together with extra-linguistic information about the text such as situation, *etc.*;
- External knowledge sources, including lexical, encyclopedic, etc. resources, as well as hand-devised knowledge sources, which provide data useful to associate words with senses.

All disambiguation work involves matching the context of the instance of the word to be disambiguated with either information from an external knowledge source (knowledge driven WSD), or information about the contexts of previously disambiguated instances of the word derived from corpora (data-driven or corpus-based WSD). Any of a variety of association methods is used to determine the best match between the current context and one of these sources of information, in order to assign a sense to each word occurrence.

2.2 Approaches of Disambiguation

In this way there can be the following approaches of disambiguation [12]:

- **Knowledge-Based Disambiguation**
    - **WSD using Selectional Preferences (or restrictions)**

      They have frequently been cited as useful information for WSD. But it has been noted that their use is limited and that additional sources of knowledge are required for full and accurate WSD. Indeed, the exemplar for sense disambiguation in most computational settings is Katz and Fodor's use of Boolean selection restrictions to constrain semantic interpretation. [31] For example, '**खाना'** can be treated as food or to eat, only first sense is available in the context of **'उसको आम खाना है ।'**, only second sense is applicable here as '**आम'** species the selection restriction to eat in the context.

- **Overlap Based Approaches**

  These require a Machine Readable Dictionary (MRD).They find the overlap between the features of different senses of an ambiguous word (sense bag) and the features of the words in its context (context bag).

- **Machine Learning Based Approaches**

  These approaches can be divided into three different approaches-

  - **Supervised Approaches**

    Supervised methods are based on the assumption that the context can provide enough evidence on its own to disambiguate words (hence, world knowledge and reasoning are deemed unnecessary). These supervised methods are subject to a new knowledge acquisition bottleneck since they rely on substantial amounts of manually sense-tagged corpora for training, which are laborious and expensive to create.

  - **Semi-supervised Algorithms**

    Its example is the bootstrapping approach. The bootstrapping approach starts from a small amount of seed data for each word: either manually-tagged training examples or a small number of surefire decision rules (*e.g.,* '**आम**' in the context of '**फल**' almost always indicates the fruit). The seeds are used to train an initial classifier, using any supervised method.

  - **Unsupervised Algorithms**

    They are the greatest challenge for WSD researchers. The underlying assumption is that similar senses occur in similar contexts, and thus senses can be induced from text by clustering word occurrences using some measure of similarity of context. It is hoped that unsupervised learning will overcome the knowledge acquisition bottleneck because they are not dependent on manual effort.

- **Hybrid Approaches**

These approaches are the hybrid between different methods like statistical based and rule based methods of machine learning approaches. By this approach, we can also combine the advantages of corpus-based and knowledge-based methods *e.g.* Sin-Jae Kang [32] uses this approach in which he has taken semi-automatically constructed ontology as an external source and secured dictionary information as context information. It is a knowledge base with information about the concepts existing in the world, their properties and how they relate to each other. In his work, he applied the previously-secured dictionary information to select the correct senses of some ambiguous words with high precision, and then use the ontology to disambiguate the remaining ambiguous words

Chapter 3 Comparative Study of Various Approaches of WSD

As we discussed in Chapter 2, the different approaches of Word Sense Disambiguation, now we discuss different algorithms developed under these approaches.

3.1 Knowledge Based Approaches

It relies on knowledge resources of Machine Readable Dictionaries in form of WordNet, and Thesaurus *etc.* They may use grammar rules for disambiguation or they may use hand coded rules for disambiguation. In recent years, most dictionaries made available in Machine Readable Dictionaries format (MRD) like that of Oxford English Dictionary, Collins, Longman Dictionary of Ordinary Contemporary English (LDOCE);Thesauruses which add synonymy information like Roget Thesaurus ; and Semantic networks which add more semantic relations like WordNet, EuroWordNet. These are for English [6].

For the purpose of Hindi, the purpose the data on which application is to be tested is provided by Central Institute of Indian Languages and the MRD (format that is being used is of Word Net. This thesis uses the Hindi WordNet prepared by IIT, Bombay.

MRD format dictionaries include a list of meanings, definitions (for all word meanings), and typical usage examples (for most word meanings).While a thesaurus adds an explicit synonymy relation between word meanings and the semantic network adds Hypernymy/hyponymy (IS-A), meronymy/holonymy (PART-OF), antonymy, entailment, *etc.* [6]

The different approaches discussed as follows-

3.1.1 Using Selectional Preferences and Arguments

Example using the word form ‘serve’-

Sense 1 -This airline ***serves*** dinner in the evening flight.

- serve (Verb)
- agent
- object – edible

Sense 2 - This airline ***serves*** the sector between Agra & Delhi.

- serve (Verb)
- agent
- object – sector

This approach requires exhaustive enumeration of:

- Argument-structure of verbs.
- Selectional preferences of arguments.
- Description of properties of words such that meeting the selectional preference criteria can be decided.
  e.g. This flight serves the 'region' between Mumbai and Delhi.
  How do we decide if 'region' is compatible with 'sector'?

3.1.2 Using Overlap Based Approaches

These require a Machine Readable Dictionary (MRD) [12].

- Find the overlap between the features of different senses of an ambiguous word (sense bag) and the features of the words in its context (context bag).
- These features could be sense definitions, example sentences, hypernyms etc.
- The features could also be given weights.
- The sense which has the maximum overlap is selected as the contextually appropriate sense.

These machine readable dictionaries may include WordNet, Thesaurus *etc.* Thesauri provide information about relationships among words. Thesaurus based disambiguation makes use of the semantic categorization provided by a thesaurus or a dictionary with subject categories. The most frequently used thesaurus in WSD is Roget's International Thesaurus (Roget, 1946) which was put into machine-tractable form in 1950s. The basic inference in thesaurus-based disambiguation is that semantic categories of the words in a context determine the semantic category of that context as a whole. And this category then determines the correct senses that are used.

There are many algorithms used for Overlap Based Approaches. The major algorithms used for this approaches are as follows –

- Lesk's Algorithm
- Walker's Algorithm
- WSD using Conceptual Density

These are explained in the following sections.

3.1.2.1 Lesk's Algorithm

The Lesk algorithm is used for the Overlap Based Approaches which is explained as follows-

1. For a polysemous word *w* needing disambiguation, a set of context words in its surrounding window is collected. Let this collection be *C*, the context bag.
2. For each sense *s* of *w*, do the following
   (a) Let *B* be the bag of words obtained from the
      (I) Synonyms
      (II) Glosses
      (III) Example Sentences
      (IV) Hypernyms
      (V) Glosses of Hypernyms
      (VI) Example Sentences of Hypernyms
      (VII) Hyponyms
      (VIII) Glosses of Hypernyms
      (IX) Example Sentences of Hypernyms
      (X) Meronyms
      (XI) Glosses of Meronyms
      (XII) Example Sentences of Meronyms
   (b) Measure the 'overlap' between C and B using the intersection similarity measure.
3. Output that the sense s as the most probable sense which has the maximum overlaps [14].

Our thesis is based on this algorithm.

3.1.2.2 Walker's Algorithm

Walker proposed in 1987, an algorithm as follows: each word is assigned to one or more subject categories in the thesaurus. If the word is assigned to several subjects, then it is assumed that they correspond to different senses of the word. Black applied this approach to five different words and achieved accuracies around 50% [15].

3.1.2.3 WSD using Conceptual Density

Select a sense based on the relatedness of that word-sense to the context. Relatedness is measured in terms of conceptual distance (*i.e.* how close the concept represented by the word and the concept represented by its context words are).This approach uses a structured hierarchical semantic net (WordNet) for finding the conceptual distance. Smaller the conceptual distance higher will be the conceptual density *i.e.* if all words in the context are strong indicators of a particular concept then that concept will have a higher density. [12]

3.1.3 Development in the field of Knowledge Based Approaches

Dictionaries have long been recognized as possible sources of information for computational methods concerned with word meanings. For example, in the early to mid 1960's, Sparck–Jones developed techniques that identified synonyms by clustering terms based on the content words that occurred in their glosses. [16]

3.1.3.1 Quillian's Approach

In the mid to late 1960's, Quillian described how to use the content of a machine readable dictionary to make inferences about word meanings. He proposed that the contents of a dictionary be represented in a semantic network. Each meaning associated with a word is represented by a node, and that node is connected to those words that are used to define the concept in the dictionary. The content words in the definitions are in turn connected to the words that are used to define them, and so forth, thus creating a large web of words. Once this structure is created for a variety of concepts, spreading activation is used to find the intersecting words or concepts in the definitions of a pair of words, thus suggesting how they are related. For example, in one of his examples, he finds that *cry*

and *comfort* share the word *sad* in their glosses, which suggests that they are related to this emotion. As such this represents an early use of exploiting gloss overlaps (shared words in dictionary definitions) to make determinations about word meanings. [17]

Due to the limitations of available computing hardware, and the lack of online dictionaries, progress in exploiting dictionary content automatically was slow but steady. However, by the 1980's computing resources were much more powerful, and Machine Readable Dictionaries were becoming more widely available.

3.1.3.2 Lesk's Approach

The Lesk algorithm may be identified as a starting point for resurgence of activity in this area that continues to this day. It selects a meaning for a particular target word by comparing the dictionary definitions of its possible senses with those of the other content words in the surrounding window of context. It is based on the intuition that word senses that are related to each other, are often defined in a dictionary using many of the same words. In particular, the Lesk's algorithm treats glosses as unordered bags of words, and simply counts the number of words that overlap between each sense of the target word and the senses of the other words in the sentence. This algorithm selects the sense of the target word that has the most overlaps with the senses of the surrounding words.

Lesk's description of his algorithm includes various ideas for future research, and in fact several of the issues he raised continue to be topics of research even today. For example, should the Lesk algorithm be used to disambiguate all the words in a sentence at once, or should it proceed sequentially, from one word to the next? If it did proceed sequentially, should the previously assigned senses influence the outcome of the algorithm for following words? Should words that are located further from the target word be given less importance than those that are nearby? Lesk also hypothesized that the length of the glosses is likely to be the most important issue in determining the success or failure of this method. [13]

3.1.3.3 Wilks' Approach

Wilks concerned that dictionary glosses are too short to result in reliable disambiguation. They developed a context vector approach that expands the glosses with related words,

which allows for matching to be based on more words and presumably result in finer grained distinctions in meaning than is possible with short glosses. As become standard for much of the work in the early 1990's, they used Longman's Dictionary of Contemporary English (LDOCE). One of the appeals of LDOCE for gloss matching work is that it has a controlled definition vocabulary of approximately 2,200 words, which increases the likelihood of finding overlaps among word senses. [18]

3.1.3.4 Cowie's Approach

Cowie suggest that while the Lesk algorithm is capable (in theory) of disambiguating all the words in a sentence simultaneously, the computational complexity of such an undertaking is enormous and makes it difficult in practice. They employ simulated annealing to simultaneously search for the senses of all the content words in a sentence. If the assignment of senses was done using an exhaustive search the time involved would be prohibitive (since each possible combination of senses would have to be considered). However, simulated annealing can find a solution that globally optimizes the assignment of senses among the words in the sentence without exhaustive search. [19]

3.1.3.5 Veronis & Ide's Approach

While quite a bit of research has been designed to extend and improve Lesk's algorithm, there has also been a body of work that is more directly linked to Quillian's spreading activation networks. For example, Veronis and Ide represent the senses of words in a dictionary in a semantic network, where word nodes are connected to sense nodes that are then connected to the words that are used to define that sense. Disambiguation is performed via spreading activation, such that a word that appears in the context is assigned the sense associated with a node that is located in the most heavily activated part of the network. [20]

3.1.3.6 Kozima & Furugori's Approach

Kozima and Furugori construct a network from LDOCE glosses that consist of nodes representing the controlled vocabulary, and links to show the co-occurrence of these

words in glosses. They define a measure based on spreading activation that results in a numeric similarity score between two concepts. [21]

3.1.3.7 Nitwa & Nitta's Approach

Niwa and Nitta compare context vectors derived from co–occurrence statistics of large corpora with vectors derived from the path lengths in a network that represent their co–occurrence in dictionary definitions. In the latter case, they construct a Quillian–style network where words that occur together in a definition are linked, and those words are linked to the words that are used in their definitions, and so forth. They evaluate Wilk's context vector method of disambiguation, and find that dictionary content is a more suitable source of co–occurrence information than are other corpora. [22]

The wide availability of WordNet as a concept hierarchy has led to the development of a number of approaches to disambiguation based on exploiting its structure. [13]

3.1.3.8 Sussna's Approach

Sussna proposes a disambiguation algorithm assigns a sense to each noun in a window of context by minimizing a semantic distance function among their possible senses. While this is similar to our approach of disambiguation via maximizing relatedness, his disambiguation algorithm is based on a measure of relatedness among nouns that he introduces. This measure requires that weights be set on edges in the WordNet noun hierarchy, based on the type of relation the edge represents. His measure accounts for *is–a* relations, as well as *has–part*, *is–a–part–of*, and *antonyms*. This measure also takes into account the compressed edge lengths that exist at higher levels of the WordNet hierarchy, where a single link suggests a much greater conceptual distance than links lower in the hierarchy. [23]

3.1.3.9 Agirre & Rigau's Approach

Agirre and Rigau introduce a similarity measure based on conceptual density and apply it to the disambiguation of nouns. It is based on the *is–a* hierarchy in WordNet, and only applies to nouns. This measure is similar to the disambiguation technique proposed by

Wilks, in that it measures the similarity between a target noun sense and the nouns in the surrounding context. [24]

3.1.3.10 Banerjee & Pedersen's Approach

Banerjee and Pedersen suggest an adaptation of the original Lesk algorithm in order to take advantage of the network of relations provided in WordNet. Rather than simply considering the glosses of the surrounding words in the sentence, the concept network of WordNet is exploited to allow for glosses of word senses related to the words in the context to be compared as well. In effect, the glosses of surrounding words in the text are expanded to include glosses of those words to which they are related through relations in WordNet. They also suggest a scoring scheme such that a match of *n* consecutive words in two glosses is weighted more heavily than a set of *n* one word matches. [25] Our thesis is also based on the WordNet's concept hierarchy of Hindi WordNet.

3.1.3.11 Pedersen, Banerjee & Patwardhan's Approach

Pedersen, Banerjee and Patwardhan introduced an algorithm that uses measures of semantic relatedness to perform word sense disambiguation. This algorithm finds its roots in the original Lesk algorithm, which disambiguates a polysemous word by picking that sense of the target word whose definition has the most words in common with the definitions of other words in a given window of context. Lesk's intuition was that related word senses will be defined using similar words, and there will be overlaps in their definitions that will indicate their relatedness. That algorithm performs disambiguation using any measure that returns a relatedness or similarity score for pairs of word senses. [13]

3.1.4 Comparison of Knowledge Based Approaches

The comparisons of the above explained approaches are as follows-

Algorithm	**Accuracy**
WSD using Selectional Restrictions	44% on Brown Corpus

Algorithm	Accuracy
Lesk's algorithm	50-60% on short samples of "Pride and Prejudice" and some "news stories".
WSD using conceptual density	54% on Brown corpus
Walker's algorithm	50% when tested on 10 highly polysemous English words.

Table 3.1: Comparison of Knowledge Based Approaches [12]

3.1.5 Drawbacks of KB Approaches

The drawbacks of knowledge based approaches are as follows-

1. Drawbacks of Selectional Restrictions

The drawbacks of WSD using Selectional Restrictions are as follows-

- They need exhaustive knowledge base.

2. Drawbacks of Overlap Based Approaches

The drawbacks of Overlap based approaches are as follows-

- The dictionary definitions present in MRD are generally very small.
- The dictionary entries rarely take into account the distributional constraints of different word senses *e.g.* selectional preferences, kinds of prepositions, *etc* .For Instance, '**सिगरेट**' and '**राख**' never co-occur in a dictionary.
- They suffer from the problem of sparse match. In the field of Natural Language Processing (NLP) , most of the events occur rarely, even when large quantities of data are available, this condition is known as sparse matching of data.[33]
- The proper nouns are not present in a MRD. Hence these approaches fail to capture the strong clues provided by proper nouns *e.g.* 'Sachin Tendulkar' will be a strong indicator of the category 'sports' as Sachin Tendulkar plays cricket.

3.2 Machine Learning Based Approaches

It uses three different types of approaches-

- **Supervised approaches-** It is based on a labeled training set. The learning system has a training set of 'feature-encoded inputs' and their appropriate sense label (category).
- **Semi-supervised algorithms-** It is based on unlabeled corpora. The learning system has a training set of 'feature-encoded inputs' but not their appropriate sense label (category).
- **Unsupervised Algorithms-** The earlier approaches disambiguate each word in isolation. But in this approach, connections between words in a sentence can help in disambiguation. The graph [34] is a natural way to capture connections between entities, which utilize relations between senses of various words.

3.2.1 Supervised Learning

This learning collects a set of examples that illustrate the various possible classifications or outcomes of an event. These identify patterns in the examples associated with each particular class of the event. These generalize those patterns into rules and further these rules are applied to classify a new event. [6] In this way, they are class of methods that induces a classifier from manually sense-tagged text using machine learning techniques. The resources used by these can be Sense Tagged Text, Dictionary (implicit source of sense inventory), Syntactic Analysis (POS tagger, Chunker, Parser).

Its scope is typically one target word per context; part of speech of target word resolved or lends itself to 'targeted word' formulation. This approach reduces WSD to a classification problem where a target word is assigned the most appropriate sense from a given set of possibilities based on the context in which it occurs.

The methodology of supervised learning [6] is as follows-

- Create a sample of training data where a given target word is manually annotated with a sense from a predetermined set of possibilities.
    - One tagged word per instance/lexical sample disambiguation
- Then, select a set of features with which to represent context.
    - Co-occurrences, collocations, POS tags, verb-object relations, etc.

- Convert sense-tagged training instances to feature vectors.
- Apply a machine learning algorithm to induce a classifier.
    - Form – structure or relation among features
    - Parameters – strength of feature interactions
- Convert a held out sample of test *data* into feature vectors.
    - 'correct' sense tags are known but not used
- Apply classifier to test instances to assign a sense tag.
- Once data is converted to feature vector form, any supervised learning algorithm can be used. Many have been applied to WSD with good results:

    - Support Vector Machines
    - Nearest Neighbor Classifiers
    - Decision Trees
    - Decision Lists
    - Naïve Bayesian Classifiers
    - Perceptrons
    - Neural Networks
    - Graphical Models
    - Log Linear Models

The major algorithms are as follows-

3.2.1.1 Naïve Bayesian Classifiers -

- Naïve Bayesian Classifier is well known in Machine Learning community for good performance across a range of tasks and so WSD is no exception.
- Assumes conditional independence among features, given the sense of a word. The form of the model is assumed, but parameters are estimated from training instances.
- When applied to WSD, features are often "a bag of words" that comes from the training data. It has usually thousands of binary features that indicate if a word is present in the context of the target word (or not).

- This algorithm suffers from the problem of data sparseness.
- Since the scores are a product of probabilities, some weak features might pull down the overall score for a sense.
- In it, a large number of parameters to be trained [6].

$$\hat{s} = \underset{s \in senses}{\operatorname{argmax}} \Pr(s|V_w)$$

'V_w' is a feature vector consisting of-

- POS of w
- Semantic & Syntactic features of w
- Collocation vector (set of words around it) typically consists of next word(+1), next-to-next word(+2), -2, -1 & their POS's
- Co-occurrence vector (number of times w occurs in bag of words around it)

Applying Bayes rule and naive independence assumption

$$\hat{s} = \underset{s \in senses}{\operatorname{argmax}} \Pr(s).\Pi_{i=1} P(n,r)(V_{w\,i}|s)$$

3.2.1.2 Decision Lists and Trees

- These are very widely used in Machine Learning. Decision trees used very early for WSD research. It is a word-specific classifier and a separate classifier needs to be trained for each word. It uses the single most predictive feature which eliminates the drawback of Naïve Bayes.
- It is based on 'One sense per collocation' property.
  - The nearby words provide strong and consistent clues as to the sense of a target word.
- It represent disambiguation problem as a series of questions (presence of feature) that reveal the sense of a word.
- The list decides between two senses after one positive answer
  - Tree allows for decision among multiple senses after a series of answers
- Uses a smaller, more refined set of features than "bag of words" and Naïve Bayes.
- More descriptive and easier to interpret.

The Decision List for WSD is given by Yarowsky, 1994. The algorithm for decision lists [6] is as -

- Identify collocational features from sense tagged data.
- Collect a large set of collocations for the ambiguous word.
- Calculate word-sense probability distributions for all such collocations.
- Calculate the log-likelihood ratio

  Abs (Log (P(Sense$_A$ | Collocation$_i$) / P(Sense$_B$ | Collocation$_i$)))
- Higher log-likelihood is equal to more predictive evidence.
- Collocations are ordered in a decision list, with most predictive collocations ranked highest.

Well known decision tree learning algorithms include ID3 and C4.5.In Senseval-1, a modified decision list (which supported some conditional branching) was most accurate for English Lexical Sample task.

3.2.1.3 Exemplar Based WSD (K-NN)

It is a word-specific classifier. This algorithm will not work for unknown words which do not appear in the corpus. It uses a diverse set of features (including morphological and noun-subject-verb pairs). The algorithm for it, can be stated as -

An exemplar based classifier is constructed for each word to be disambiguated.

Step 1: From each sense marked sentence containing the ambiguous word, a training example is constructed using:

- POS of *w* as well as POS of neighboring words.
- Local collocations
- Co-occurrence vector
- Morphological features
- Subject-verb syntactic dependencies

Step 2: Given a test sentence containing the ambiguous word, a test example is similarly constructed.

Step 3: The test example is then compared to all training examples and the k-closest training examples are selected.

Step 4: The sense which is most prevalent amongst these 'k' examples is then selected as the correct sense.

3.2.1.4 WSD Using Support Vector Machines

It is a word-sense specific classifier. It gives the highest improvement over the baseline accuracy. It uses a diverse set of features. SVM is a binary classifier which finds a hyper plane with the largest margin that separates training examples into 2 classes. As SVMs are binary classifiers, a separate classifier is built for each sense of the word

Training Phase: Using a tagged corpus, f or every sense of the word a SVM is trained using the following features:

- POS of ***w*** as well as POS of neighboring words.
- Local collocations
- Co-occurrence vector
- Features based on syntactic relations (e.g. headword, POS of headword, voice of head word etc.)

Testing Phase: Given a test sentence, a test example is constructed using the above features and fed as input to each binary classifier.

The correct sense is selected based on the label returned by each classifier.

3.2.1.5 WSD Using Perceptron trained HMM

It is significant in lieu of the fact that a fine distinction between the various senses of a word is not needed in tasks like Machine Translation. A broad coverage classifier as the same knowledge sources can be used for all words belonging to super sense. Even though the polysemy was reduced significantly there was not a comparable significant improvement in the performance. In this methodology, WSD is treated as a sequence labeling task. The class space is reduced by using WordNet's super senses instead of actual senses. A discriminative Hidden Markov Model is trained using the following features:

- POS of *w* as well as POS of neighboring words.
- Local collocations
- Shape of the word and neighboring words

e.g. for s = "Merrill Lynch & Co shape(s) =Xx*Xx*&Xx

This method lends itself well to Named Entity Recognition (NER) as labels like 'person', 'location', 'time' *etc*. are included in the super sense tag set.

3.2.2 Comparison of Supervised Approaches

The comparison of all the supervised explained above are as follows-

Approach	**Average Precision**	**Average Recall**	**Corpus**	**Average Baseline Accuracy**
Naïve Bayes	64.13%	Not Reported	Senseval-3 All Words Task	60.9%
Exemplar Based (k-NN)	68.6%	Not Reported	WSJ6 containing 191 content words	63.7%
Decision Lists	96%	Not applicable	Tested on a set of 12 highly polysemous English words	63.9%
SVM	72.4%	72.4%	Senseval 3 – Lexical sample Task (Used for disambiguation of 57 words)	55.2%
Perceptron trained HMM	67.6%	73.74%	Senseval 3 – All Words Task	60.9%

Table 3.2: Comparison of Supervised approaches [12]

3.2.3 Semi-Supervised Algorithms

These work at par with its supervised version even though it needs significantly fewer amounts of tagged data. It has all the advantages and disadvantaged of its supervised version.This learning algorithm has the characteristics of learning sense classifiers from annotated data, with minimal human supervision. For example [6]

1. Automatically bootstrap a corpus starting with a few human annotated examples

2. Use monosemous relatives / dictionary definitions to automatically construct sense tagged data
3. Rely on Web-users + active learning for corpus annotation

In this way, it expands applicability of supervised WSD. The algorithms that this approaches use come under the category of bootstrapping approaches. The basic ingredients of this bootstrapping approach is –

- Some labelled data
- Large amounts of unlabelled data
- One or more basic classifiers

The output by this approach is a new classifier that improves over the basic classifiers.

This is based on Yarowsky's supervised Bootstrapping algorithm that uses Decision Lists. The Yarowsky's approach relies on two heuristics and a decision list

- One sense per collocation :
  - Nearby words provide strong and consistent clues as to the sense of a target word
- One sense per discourse :
  - The sense of a target word is highly consistent within a single document

The learning algorithm in it uses a decision list to classify instances of target word. The classification is based on the highest ranking rule that matches the target context.

The algorithm for it is as follows-

Step1: Train the Decision List algorithm using a small amount of seed data.

Step2: Classify the entire sample set using the trained classifier.

Step3: Create new seed data by adding those members which are tagged as Sense-A or Sense-B with high probability.

Step4: Retrain the classifier using the increased seed data.

- Exploits "One sense per discourse" property
- Identify words that are tagged with low confidence and label them with the sense which is dominant for that document

e.g. 'the loss of animal and plant species through extinction'

LogL	Collocation	Sense
9.31	flower (within +/- k words)	A (living)
9.24	job (within +/- k words)	B (factory)
9.03	fruit (within +/- k words)	A (living)
9.02	***plant* species**	**A (living)**

Table 3.3: An Example of Semi-supervised Algorithm: Here last row is appropriated in word sense

Its process can be summarized as in the way of - initialization, progress and convergence. In this initialization phase, all occurrences of the target word are identified and a small training set of seed data is tagged with word sense. In the progress phase, the seed set grows and the residual set shrinks, as in the upper half it shows different circled data of life, cell, species and microscopic type and slowly it is expanding. In the convergence phase, the convergence stops when residual step stabilizes.

3.2.4 Comparison of Semi- Supervised Approaches

The comparison of all these approaches is as follows-

Approach	**Average Precision**	**Corpus**	**Average Accuracy**	**Baseline**
Supervised Decision Lists	96.1%	Tested on a set of 12 highly polysemous English words	63.9%	
Semi-Supervised Decision Lists	96.1%	Tested on a set of 12 highly polysemous English words	63.9%	

Table 3.4: Comparison of Semi-Supervised approaches

3.2.5 Unsupervised Algorithms

Unsupervised learning identifies patterns in a large sample of data, without the benefit of any manually labeled examples or external knowledge sources. These patterns are used to divide the data into clusters, where each member of a cluster has more in common with the other members of its own cluster than any other. If one may remove manual labels

from supervised data and cluster, one may not discover the same classes as in supervised learning. In this way, Supervised Classification identifies features that trigger a sense tag and Unsupervised Clustering finds similarity between contexts.

If Sense tagged text is available, it can be used for evaluation. But these sense tags aren't used for clustering or feature selection. Now, assume that sense tags represent 'true' clusters, and then compare these to discover clusters in such a way that find mapping of clusters to senses that attains maximum accuracy. The pseudo words are especially useful, since it is hard to find data that is discriminated. So, pick two words or names from a corpus, and conflate them into one name. Then see how well we can discriminate. The baseline Algorithm is that, that group all instances into one cluster; this will reach 'accuracy' equal to majority classifier.

3.2.5.1 Major Algorithms

The major algorithms under this category are-

- **Lin's Algorithm**

It is a general purpose broad coverage approach. It can even work for words which do not appear in the corpus.

- **Hyperlex**

In it, instead of using 'dictionary defined senses', we extract the 'senses from the corpus' itself. These 'corpus senses' or 'uses' corresponds to clusters of similar contexts for a word. It is a word-specific classifier. The algorithm would fail to distinguish between finer senses of a word (*e.g.* the medicinal and narcotic senses of 'drug').

- **Yarowsky's Algorithm**

It is a broad coverage classifier. It can be used for words which do not appear in the corpus but it was not tested on an 'all word corpus'.

- **WSD using Parallel Corpora**

By this algorithm we can distinguish even between finer senses of a word because even finer senses of a word get translated as distinct words. It needs a word aligned parallel corpora which is difficult to get but in it, an exceptionally large number of parameters need to be trained.

3.2.5.2 Drawbacks of Unsupervised Algorithms

The problems with this approach are as follows-

1. The unsupervised methods may not discover clusters equivalent to the classes learned in supervised learning.

2. The evaluation which is based on assuming that sense tags represent the 'true' cluster is likely a bit harsh. The alternatives are as-

- Humans could look at the members of each cluster and determine the nature of the relationship or meaning that they all share
- Use the contents of the cluster to generate a descriptive label that could be inspected by a human

3. The first order feature sets may be problematic with smaller amounts of data since these features must occur exactly in the test instances in order to be 'matched'.

3.2.5.3 Comparison of Unsupervised Approaches

The comparison of unsupervised approaches are as follows-

Approach	**Precision**	**Average Recall**	**Corpus**	**Baseline**
Lin's Algorithm	68.5%. The result was considered to be perfect if the similarity between the predicted sense and the actual sense was greater than 0.27	Not reported	Trained using WSJ corpus containing 25 million words. Tested on 7 SemCor files containing 2832 polysemous nouns	64.2%
Hyperlex	97%	82%	Tagged on a set of 10 highly polysemous French words	73%
WSD using Roget's Thesaurus	92% (average degree of polysemy was 3)	Not reported	Tested on a set of 12 highly polysemous English words	Not reported

Table 3.5: Comparison of Unsupervised approaches

3.3 Hybrid Approaches

These combine information obtained from multiple knowledge sources and it uses a very small amount of tagged data. The major algorithms regarding these approaches are explained in following sections-

3.3.1 An Iterative Approach to WSD

The main points regarding this approach is as follows -

- Uses semantic relations (synonymy and hypernymy) form WordNet.
- Extracts collocational and contextual information form WordNet (gloss) and a small amount of tagged data.
- Monosemic words in the context serve as a seed set of disambiguated words.
- In each iteration, new words are disambiguated based on their semantic distance from already disambiguated words.
- It also exploits other semantic relations available in WordNet.

3.3.2 SenseLearner

It uses some tagged data to build a semantic language model for words seen in the training corpus. It uses WordNet to derive semantic generalizations for words which are not observed in the corpus. The semantic language model used is as -

- For each POS tag, using the corpus, a training set is constructed.
- Each training example is represented as a feature vector and a class label which is word sense
- In the testing phase, for each test sentence, a similar feature vector is constructed.
- The trained classifier is used to predict the word and the sense**.**
- If the predicted word is same as the observed word then the predicted sense is selected as the correct sense.
- It improves Lin's algorithms by using semantic dependencies from WordNet *e.g.* if 'drink water**'** is observed in the corpus then using the hypernymy tree we can derive the syntactic dependency 'take-in liquid**'**. 'take-in liquid**'** can then be used to disambiguate an instance of the word tea as in 'take tea**'**, by using the hypernymy-hyponymy relations.

3.3.3 Structural Semantic Interconnections (SSI)

- It is an iterative approach.
- It uses the following relations -
    1. Hypernymy (car is a kind of vehicle) denoted by (kind-of)
    2. Hyponymy (the inverse of hypernymy) denoted by (has-kind)
    3. Meronymy (room has-part wall) denoted by (has-part)
    4. Holonymy (the inverse of meronymy) denoted by (part-of)
    5. Pertainymy (dental pertains-to tooth) denoted by (pert)
    6. Attribute (dry value-of wetness) denoted by (attr)
    7. Similarity (beautiful similar-to pretty) denoted by (sim)
    8. Gloss denoted by (gloss)
    9. Context denoted by (context)
    10. Domain denoted by (dl)
- Monosemic words serve as the seed set for disambiguation.

3.3.4 Comparison of Hybrid Approaches

The comparison of all the above approaches is as follows-

Approach	**Precision**	**Average Recall**	**Corpus**	**Baseline**
Iterative approach	92.2%	55%	Trained using 179 texts from SemCor. Tested using 52 texts created from 6 SemCor files	Not reported
SenseLearner	64.6%	64.6%	SenseEval-3 All Words Task	60.9%
SSI	68.5%	68.4%	SenseEval-3 Gloss Disambiguation Task	Not Reported

Table 3.6: Comparison of Hybrid approaches

Chapter 4 Problem Statement

In the field of computational linguistics, some results have already been obtained however, a number of important research problems have not been solved yet. Some of these problems are as follows-

4.1 Ambiguity

Ambiguity is one of these problems which have been a great challenge for computational linguistics. In general, people are unaware of the ambiguities in the language they use because they are very good at resolving them using 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. [28]

Something is ambiguous when it can be understood in two or more possible ways or when it has more than one meaning. Lexical semantics ambiguity occurs when a single word is associated with multiple senses. In this thesis, we will focus on developing a tool used to resolve semantic ambiguity. Examples of lexical ambiguity are everywhere. In fact, almost any word has more than one meaning. For example, consider the word '**तत्व**'.It can refer to following 4 meanings-

1. **जगत का मूल कारण ।**

   Here '**तत्व**' means origin of world.

2. **एक पिंड जिसमें अन्तर-संरचना होती है, द्रव्यमान सीमित होता है और जिन्हें और भी सरल रूप में विघटित नहीं किया जा सकता ।**

   Here '**तत्व**' means chemical element.

3. **पंचतत्वों में से कोई एक ।**

   Here '**तत्व**' means the one of the fundamental elements of universe according to Sanatan Dharma.

4. **किसी पदार्थ आदि का वास्तविक या मुख्य भाग या गुण ।**

Here '**तत्व**' meaning summary.

For various applications, such as information retrieval or machine translation, it is important to be able to distinguish between the different senses of a word. In a machine translation application, different senses of a word may be represented with different words in the target language. In order to correctly translate a text in one language to another, firstly we have to know the senses of the words and then find the best translation equivalent in the target language.

The key concern in machine translation, whose purpose is to convert documents from one language to another, is the language divergence problem. This problem arises from the fact that languages make different lexical and syntactic choices from expressing an idea [29]. Language divergence needs to be tackled not only for translating between language pairs from distant families (*e.g.* English and Hindi) but also for pairs which are close siblings (*e.g.* Marathi, Punjabi, Hindi, and Gujarati). The solution to language divergence problem lies in building and using knowledge networks like WordNets. In this thesis, we present an approach on how to solve word sense disambiguation problem using the Hindi WordNet and we are using Hindi WordNet as resource to solve the ambiguity.

The language divergence problem arise when a particular word in a language has multiple senses and so can be translated to more than one word in the target language. This type of problem is called the ambiguity problem.

For example, the word **'खग'** in Hindi language has different meanings like planet [**ग्रह, खग**], stars [**तारा, सितारा, तारका, तारक, रोचनी, खग, ऋक्ष**], arrow for bow [**धातु आदि का बना वह पतला लम्बा हथियार जो धनुष द्वारा चलाया जाता है**] and bird [**पंख और चोंचवाला द्विपद जिसकी उत्पत्ति अंडे से होती है और जो नियततापी होता है**] are the meanings when translated to English language. So for machine translation of such a word to English language, it'll require resolving this ambiguity that which sense of the word is referred to in the particular context. We address the formulation of the WordNet for the WSD task. If we know the correct semantic meaning of each word in the source language, we could more accurately determine the appropriate words in the target language.

Chapter 5 Design of the Problem

WordNet is like a dictionary in that it stores words and meanings. However, it differs from traditional ones in many ways. For instance, words in WordNet are arranged semantically instead of alphabetically.

5.1 Hindi WordNet

The idea of Hindi WordNet is inspired by the English WordNet that was created and being maintained at the Cognitive Science Laboratory of Princeton University under the direction of psychology professor George A. Miller [30]. Its development began in 1985. English WordNet is organized by semantic relations. Since a semantic relation is a relation between meanings, and since meanings can be represented by synsets, it is natural to think of semantic relations as pointers between synsets.

Unlike most dictionaries, WordNet contains only open-class words (nouns, verbs, adjectives, and adverbs). WordNet doesn't contain closed class words such as pronouns, conjunctions and prepositions. WordNet are organized semantically (as parts-of-speech). The central object in WordNet is a synset, a set of synonyms. The Hindi WordNet is being used in this thesis. This was produced by the researchers in the centre for Indian Language Technology (CFILT), IIT-B, directed by Prof. Pushpak Bhattacharya. [27] The famous English WordNet inspires its design. It organizes the lexical information in terms of word meanings and can be termed as lexicon based on psycholinguistic principles.
As shown in Figure, the interface of Hindi WordNet contains a keypad that is used to type the Hindi word to be search in textbox.

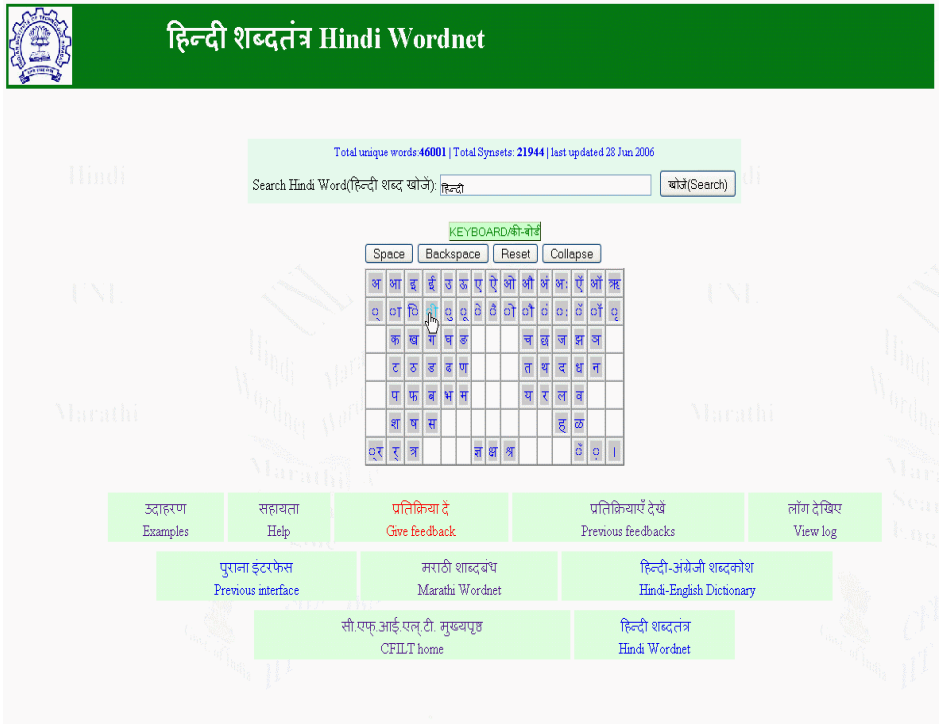

Figure 5.1 Hindi WordNet Interface [27]

5.2 Constituents of Information Provided by the WordNet

WordNet groups sets of synonymous word senses into synonyms sets or synsets. A word sense is a particular meaning of a word. For example, the word '**मूल**' has 8 meanings as noun and 4 as adjective. As a noun, it can refer to following meanings-

1. **वनस्पतियों आदि का जमीन के अंदर रहनेवाला वह भाग जिसके द्वारा उन्हें जल और आहार मिलता है**
2. **वह जिसके प्रभाव से या फलस्वरूप कोई काम हो**
3. **मकान आदि बनाने के समय उसका वह मूल भाग जो दीवारों की दृढ़ता के लिए ज़मीन खोदकर और उसमें से दीवारों की जोड़ाई आरंभ करके बनाया जाता है**
4. **वह काल जब चंद्रमा मूल नक्षत्र में होता है**
5. **वह असल धन जो किसी के पास हो या लाभ आदि के लिए व्यापार में लगाया जाए**
6. **सत्ताईस नक्षत्रों में से उन्नीसवाँ नक्षत्र**

7. **किसी वस्तु या कार्य का आरंभिक भाग**

8. **मूलभूत सिद्धान्त, प्रथा आदि**

As an adjective it can refer to following meanings –

1. **जो किसी का अनुवाद, नकल या आधार पर न हो,बल्कि अपनी उद्भावना से निकला हो**

2. **किसी वस्तु के मूल या तत्व से संबंध रखनेवाला;**

3. **जो आवश्यक हो;**

4. **जो वहीं उत्पन्न या पैदा हुआ हो जहाँ पाया जाता हो .**

The synset for the word '**मूल**' in the first set as noun as-{**जड़, मूल, सोर**}**.** The synset is the basic organizational unit in WordNet. If a word has more than one sense, it will appear in more than one synset. Synsets are organized in a hierarchy via super-class/sub-class relationship (referred to as hypernymy/hyponymy).

Each synset has a gloss (definition) associated with it. The gloss for the synset {**जड़, मूल, सोर**} is '**वनस्पतियों आदि का जमीन के अंदर रहनेवाला वह भाग जिसके द्वारा उन्हें जल और आहार मिलता है**'. The synsets also have an example in addition to the gloss e.g. for the above synset the example is '**आयुर्वेद में बहुत प्रकार की जड़ों का प्रयोग होता है'.**

Each entry in the Hindi WordNet consists of synset, gloss and ontology. An ontology is a hierarchical organization of concepts, more specifically, a categorization of entities and actions. For each syntactic category namely noun, verb, adjective and adverb, a separate ontological hierarchy is present. Each synset is mapped into some place in the ontology. A synset may have multiple parents.

Each word may have one or more senses and these are classified as-

- **Homonyms –** Two senses of a word are said to be homonyms when they mean entirely different things but have the same spelling *e.g.* the two senses of the word '**मूल**' are '**जड़**' and '**मूलनक्षत्र**' , are homonyms because they aren't related to each other.

- **Monosemous-** Words with only one sense are said to be monosemous *e.g.* '**मूलधन**' has only one sense as '**पूँजी**' so it appears in only one synset.
- **Polysemous –** They are the words with multiple senses. In WordNet, each word occurs in as many synsets as it has senses *e.g.* the word **'मूल'** occurs in 8 synsets as nouns and 4 synsets as adjectives.
- **Compound Words –** Besides single words, WordNet contains some compound words. But they are treated as single words in all respects *e.g.* WordNet has two word compounds like '**मेदिनीपोर शहर**' and 3 word compounds like '**वियतनाम समाजवादी गणराज्य**'.

5.3 Parts-of-Speech in WordNet

WordNet stores information about words that belong to four parts-of-speech: nouns, verbs, adjectives and adverbs. These are arranged in their respective synsets. Prepositions and conjunctions don't belong to any synset.

5.3.1 Nouns in WordNet

Noun words have various relations defined in WordNet for the noun part of speech. These relations are as follows-

- **Hypernymy and Hyponymy:** These are two most common relations for nouns. These are semantic relationships that connect two synsets if the entity referred to by one is a kind of or is a specific example of the entity referred to by other .Specifically, if synset *A* is a kind of *B* synset, then B is a hyponym of *A*, and *A* is the hypernym of *B e.g.* { **जड़, मूल, सोर**} is the hypernym of { **पेड़, वृक्ष, पादप, द्रुम, तरु, तरुवर, विटप, रुक्ष, रूख, विटपी, रूँख, अघ्रिप, अग, अनोकह, साखी, साखि** }**.** The number of hypernym links is equal to the number of hyponym links since for every hypernym links there is a corresponding hyponym link.
- **Meronymy and Holonymy:** These are also semantic relationships that connect two synsets if the entity referred to by the other. Specifically, synset B is a hypernym of *A* synset, if *A* is a part of *B.* Conversely, *B* is a holonym of *A* if *A* has *B* as a part.

Holonyms can be of three types: Member-Of, Substance-Of and Part-Of. Conversely there are three types of Meronyms: Has-Member, Has-Substance and Has-Part *e.g.* **'सिर'** is a part of '**शरीर**'. The synset {**जन, लोग, लोक**} is the holonym of {**जुलूस, जलूस, मार्च**} while {**व्यक्ति, मानस, आदमी, शख़्स, शख्स, जन, बंदा, बन्दा**} is its meronym.

- **Antonymy and Attribute:** Antonymy is a lexical relationship that links together two noun words that are opposites of each other. Thus the noun '**सुख'** is the antonym of the noun '**दुख'**. Since antonymy is a lexical relationship; it is defined between the words and not between the synsets in which those words occur. Thus, although the words {**सुख, चैन, आराम, ख़ुशहाली, खुशहाली, खुशाल**} share a synset with '**सुख'**, they aren't related to '**दुख'** through the antonymy relation. The attribute relation is a semantic relation that links together a noun synset with an attribute synset when *A* is a value of *B*.
- **Attribute:** This denotes the properties of noun. It is a linkage between noun and an adjective. This is a semantic relation. For instance, for word sense [**पक्षी**, **चिड़िया**, **पंछी**, **खग**, **परिंदा**, **विहंग**, **विहंगम**, **पखेरू**, **विहग**, **पर्णवी**, **दिवाचर**] of word '**खग**', the attribute exists as [**पंखदार**, **पाँखदार**, **पंखयुक्त]**.
- **Ability verb:** This link specifies the inherited features of a nominal concept. This is a semantic relation. This is a relation between noun and the verb *e.g.* for noun '**खग'**, the ability verb exists as **[उड़ना**, **उड़ान भरना**]
- **Function Verb:** This shows a linkage between nominal and verbal concept. This link specifies the function of a nominal concept. This is a semantic relation *e.g.* for the word sense **[अध्यापक, शिक्षक, आचार्य, गुरु, मास्टर]** of the word '**गुरु'** the function verb exists as **[दुखाना]** which has meaning **[कुछ ऐसा करना, कहना आदि जिससे किसी का कोई मर्म स्थान आहत हो].**

5.3.2 Verbs in WordNet

Verb words have various relations defined in WordNet for the verb parts-of-speech. These relations are as follows-

- **Hypernymy and Troponymy:** These are semantic relations and are analogous to the noun hypernymy and hyponymy respectively. Synset *A* is the hypernym of *B*, if *B* is one way to *A*; *A* is then the troponym of *B*. Thus the verb synset **{खाना, जीमना, भोजन करना}** is the troponym of **{सेवन करना, लेना, उपभोग करना}.** These two relationships form the lion's share defined for verbs in WordNet.
- **Antonymy:** Like nouns, verbs are also related through the relationship of antonymy that links two verbs that are opposite to each other in the meaning. Thus the word '**जाना'** is the antonym of the verb '**आना'**. This is a lexical relationship and doesn't belong to the other words in the synsets that both belong to.
- **Entailment and Cause:** Other relations defined for verbs include those of entailment and cause, both of which are semantic relations. A synset *A* is related to synset *B* through the entailment relationship if *A* entails doing *B*. Thus the verb synset {**चाटना, खाना**} has an entailment relationship with the [**बोलना, कहना, उचारना, उच्चारना, उच्चारण करना]**. *A* synset is related to synset *B* by the cause relationship, if *A* causes *B* [26] *e.g.* for the word **'दुखाना'** with meaning as **[किसी के घाव आदि को ऐसे छूना कि वह दर्द करने लगे],** the entailment relationship exists as **[छूना, स्पर्श करना, परसना].**

5.3.3 Adjectives and Adverbs in WordNet

The various relations concerning them are-

- **Similar-to:** It is defined for Adjectives. This is a semantic relationship that links two adjective synsets that are similar in meaning but not close enough to be put together

in the same synset *e.g.* the similar to relation of '**होशियार**' with another word-forms is [**जागरूक, चैतन्य**]

- **Also-see:** This relation is common to both adjective and adverb. All links of this type of adjective are semantic in nature but they aren't lexical relations *e.g.* for word '**होशियार**' the also Also-see relation exists for [**बुद्धि, अक्ल, प्रज्ञा, विवेक, धी, धी शक्ति, मति, मनीषा, मेधा, दिमाग़, दिमाग, मस्तिष्क, बूझ, बुझ, अक़ल, अक़्ल, अकल, समझ, जिहन, ज़िहन, जेहन, ज़ेहन, संज्ञा, मनीषिका, उद्ग्रहण, अभिबुद्धि**]
- **Modifies Noun:** This shows a linkage between nominal and adjectival concepts. It shows certain adjectives can only modify certain nouns. Such adjectives and nouns are linked in the Hindi WordNet by the relation Modifies Noun *e.g.* for '**होशियार**' it exists as [**व्यक्ति, मानस, आदमी, शख़्स, शख्स, जन, बंदा, बन्दा**].
- **Modifies Verb:** This shows a linkage between adverbial and verbal concepts. It shows certain adverbs can only go with certain verbs. Modifies Verb is a relation to show connection between such words *e.g.* for word '**अंदर**', the relation exists as [**काम करना, कार्य करना, करना**].
- **Derived form:** This relation specifies the root form from which a particular word is derived. This relation can go from noun to adjective or vice versa, noun to verb and adjective to verb and aims to handle derivational morphology. This is a lexical relation. This is also applicable to adverb *e.g.* for '**अकेला**' the relation exists for [**एक, इक**] [26]

5.4 Steps for Word Sense Disambiguation

WSD involves the association of a given word in a text or discourse with a definition or meaning which is distinguishable from other meanings potentially attributable to that word. The task therefore necessarily involves two steps-

- The first step is to determine all the senses for every word *i.e.* to choose a sense inventory, *e.g.* knowledge networks like WordNet and get the different senses of the word from WordNet or similar resources.
- The second step involves a means to assign the appropriate sense to each occurrence of a word in context. All disambiguation work involves matching the context of an instance of the word to be disambiguated either with information from external knowledge sources (like WordNet) or with contexts of previously disambiguated instances of the word.

In this thesis, we discuss the approach which involves looking for overlap between the words that are related to the word to be disambiguated through the relations like Synonymy, Antonymy, Meronymy, Holonymy *etc.* as obtained from the WordNet and the words from the text surrounding the words to be disambiguated. The sense definition chosen is the one that has the largest number of words in common with the surrounding words.

5.5 WSD algorithm

The basic idea for WSD algorithm is based on Lesk algorithm as explained in Chapter 2. The database that has been used for solving the Hindi Word Sense Disambiguation problem is Hindi WordNet. The working is explained for the Hindi WordNet as follows-

Step 1: The words surrounding the word to be disambiguated are collected and this forms the context bag in the given algorithm. These words include the nouns, verbs, adjectives and adverbs found in the sentence containing the word, the sentence previous to the sentence containing the word and the sentence following the sentence containing the word.

Step 2: This step works in two parts as follows-

Step 2a: The information regarding the word or its each sense is collected from the WordNet *i.e.* the words from the various relations like Synonymy, Antonymy, Hypernymy, Meronym etc. occurring in these relations are collected.

Step 2b: This step try to do the overlap between the context and the semantic bag for each sense of the word to be disambiguated is found.

Step 3: The sense that has the maximum sense for the overlap is the winner case.

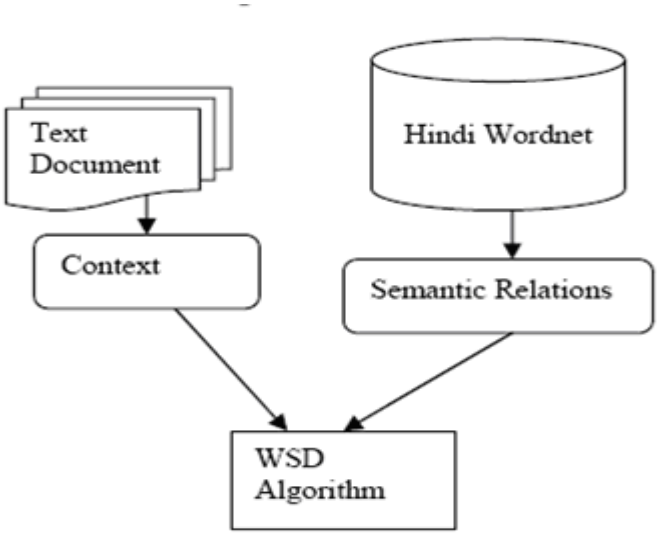

Figure 5.2: Showing the basic idea behind Hindi WordNet

Now we explain the formation of context bag as follows

The context of the word to be disambiguated is collected from a window around it. In the present case the window is the-

- Sentence in which the word occurs.
- The previous sentence.
- The following sentence.

For example, consider the following paragraph of text-

"ज्योतिर्विज्ञान प्रस्तुत समय को जो 1850 ईसवी सदी से आरम्भ होकर 2005 ईसवी सदी में समाप्त होगा- संधि काल, परिवर्तन काल, कलियुग के अंत तथा सतयुग के आरम्भ का काल मानता चला आया है । इसीलिए इस समय को **युगसंधि** की वेला कहा गया है । कुछ रुढ़िवादी पण्डितों के अनुसार नया युग आने में अभी 3 लाख 24 हजार वर्ष की देरी है, किन्तु यह प्रतिपादन भ्रामक है, यह वास्तविक काल गणना करने पर पता चलता है । हरिवंश पुराण के भविष्य पर्व से

लेकर श्रीमद्भगवत गीता का उल्लेख कर परम पूज्य गुरुदेव ने इसमें यह प्रमाणित किया है कि वास्तविक काल गणना के अनुसार सतयुग का समय आ पहुँचा ।"

In the above text, the word to be disambiguated (here **युगसंधि)** is underlined. The words in the window will include the sentence in which the word occurs, the previous sentence and the following sentence. Hence the window will include the following text of the paragraph.

"ज्योतिर्विज्ञान प्रस्तुत समय को जो 1850 ईसवी सदी से आरम्भ होकर 2005 ईसवी सदी में समाप्त होगा- संधि काल, परिवर्तन काल, कलियुग के अंत तथा सतयुग के आरम्भ का काल मानता चला आया है । इसीलिए इस समय को **युगसंधि** की वेला कहा गया है । कुछ रुढ़िवादी पण्डितों के अनुसार नया युग आने में अभी 3 लाख 24 हजार वर्ष की देरी है, किन्तु यह प्रतिपादन भ्रामक है, यह वास्तविक काल गणना करने पर पता चलता है ।"

This context provides what we call as context bag. Context Bag consists of the words occurring around the word to be disambiguated. Then, the semantic bag for the word to be disambiguated is constructed. The WordNet is mined with a view to find the semantic associations of the given word. A set of word is collected by traversing the WordNet graph. This set consists of the words that occur as Hypernyms, Hyponyms, Meronyms, Holonyms, and Synonyms etc. of the word that is to be disambiguated. This set is called Semantic Bag.

5.6 Formation of Semantic Bag with the help of Hindi WordNet

The semantic bag for the word '**युग**' can be prepared as follows-

Step 1: In this step, we will first search the word in Hindi WordNet which will give the output for the word '**युग**' as

There are 6 senses of '**युग**' as NOUN (संज्ञा):

1. **युग, जुग**; पुराणानुसार काल के यह चार परिमाण या विभाग सतयुग,त्रेता,द्वापर और कलि में से प्रत्येक; "भगवान राम का जन्म त्रेता युग में हुआ था"

2. **युग, काल, जुग**; इतिहास का कोई ऐसा बड़ा कालमान जिसमें एक ही प्रकार के कार्य, घटनाओं आदि की प्रमुखता हो; "भक्ति युग हिंदी साहित्य में स्वर्ण युग के नाम से जाना जाता है"

3. **जुआ, जूआ, जुआठ, जुआठा, युग, जूड़, माची**; गाड़ी, हल आदि के आगे की वह लकड़ी जो बैलों के कंधे पर रहती है; "बैल जुआ तोड़ कर भाग गया"

4**. जोड़ी, जोड़ा, जोड़, जोट, युग्म, युगल, जुगल, युगम, यमल, युग**; एक ही तरह की दो चीज़ें; "यह कबूतर की जोड़ी अच्छी है"

5. **जोड़ी, जोड़ा, जोड़, जोट, युग्म, युगल, जुगल, युगम, यमल, युग, मिथुन**; नर और मादा का युग्म; "बहेलिये ने क्रौंच पक्षी के जोड़े में से एक को मार दिया"

6. **जोड़ी, जोड़ा, जोड़, जोट, युग्म, युगल, जुगल, युगम, यमल, युग**; दो व्यक्ति, वस्तु आदि जो एक-दूसरे के सहयोगी या सम्बद्ध हों; "उनकी जोड़ी बड़ी अच्छी लगती है"

These different word senses, their glosses and meanings can be saved into a text file. This also shows that it has two other links for hypernym and hyponym.

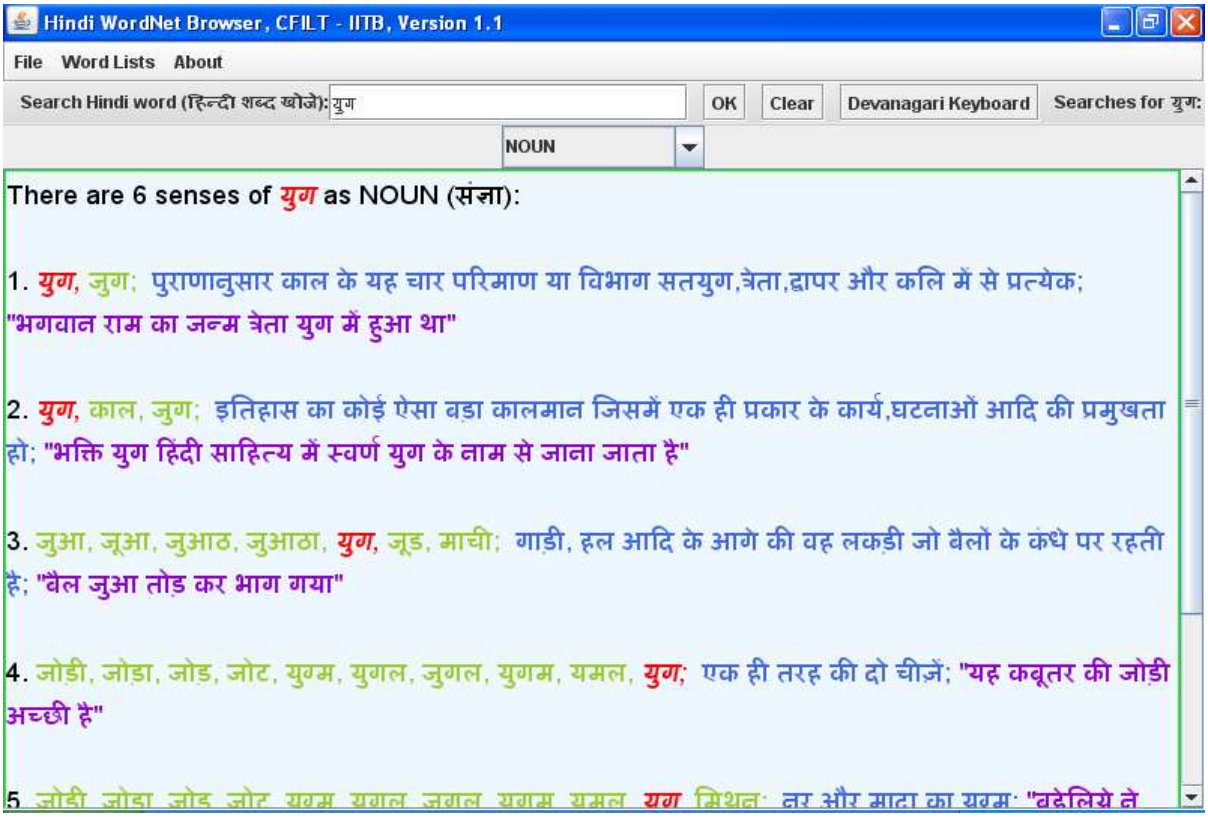

Figure 5.3: Different senses of word '**युग**'

Step 2: The hypernyms, hyponyms, meronyms *etc.* are searched corresponding to that specific word and saved into the text file in a similar way by going in the File menu and saving everything all the data in the text file and naming them correspspondinly according to their property like of above Hypernyms, Hyponyms etc., so that, these words can be used for the formation of semantic bag.

e.g. for the word '**युग**', the hypernyms are as-

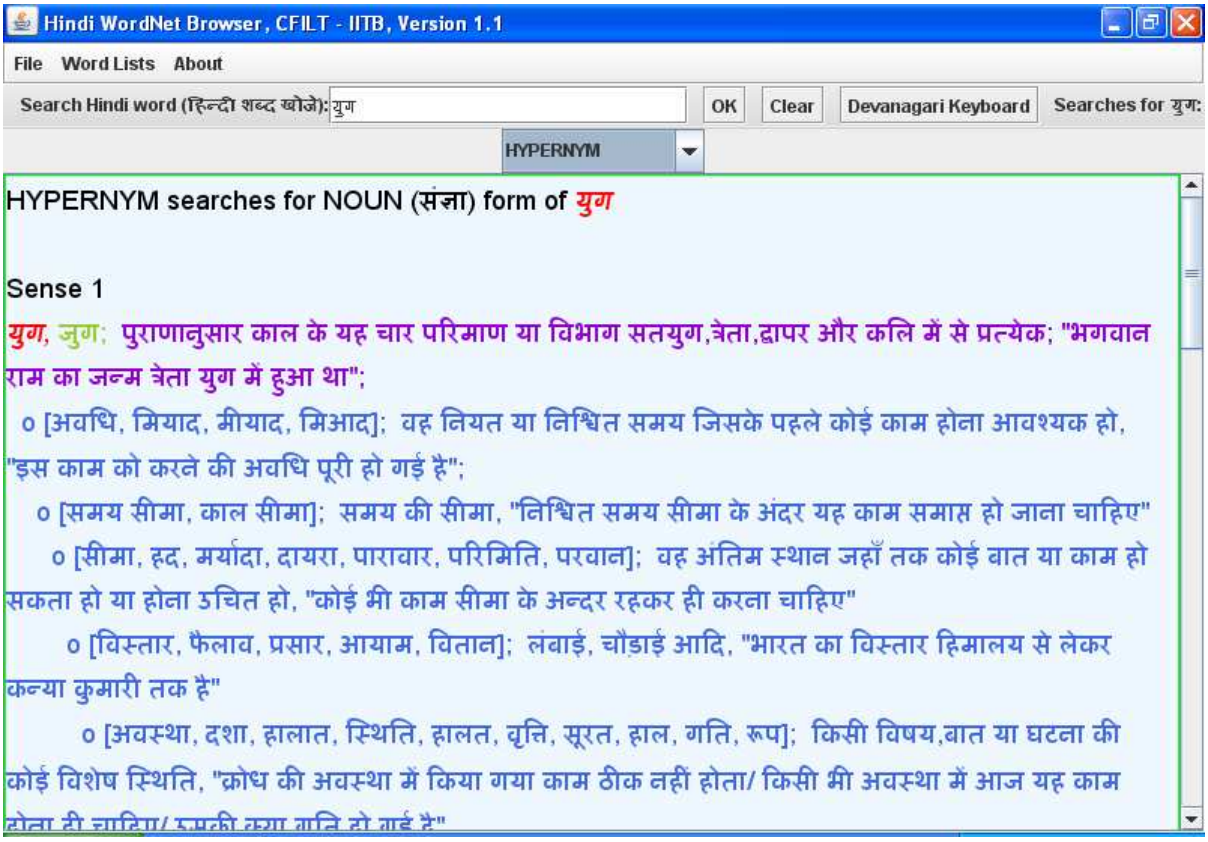

Figure 5.4: Different Hypernyms for '**युग**'

5.7 Demonstration of algorithm with Hindi WordNet

Consider the text written below in Hindi-

"हाल के खेल में एमटेल मास्टर्स का खिताब जीतने वाले उक्रेन के इवानचुक ने 2.5-1.5 अंकों के साथ जीत दर्ज करके आनंद को आठवां खिताब जीतने से रोक दिया। पहली ***बाजी*** ने ही इवानचुक की जीत की नींव रख दी जबकि इस उक्रेनी ग्रैंड मास्टर ने काले मोहरों से खेलते हुए बढ़त बना ली। आनंद ने अन्तिम बाजी की 38वीं चाल में गंभीर गलती की जिसके बाद उन्हें तुरंत हार माननी पड़ी।"

Context Bag

Context Bag will contain the words as-

“हाल,एमटेल,मास्टर्स,खिताब,जीतने,उक्रेन, इवानचुक, ,अंकों, जीत, दर्ज, करके, आनंद, आठवां, खिताब, जीतने, रोक,दिया,पहली,***बाजी*** ,इवानचुक,नींव,ग्रैंड , मास्टर,काले ,मोहरों, खेलते,अन्तिम ,बढ़त,38वीं,चाल,गंभीर,गलती,उन्हें,तुरंत,हार”

Semantic Bag

The semantic Bag is formed for each sense of the word ‘**बाजी’** by collecting the data from the WordNet. The various senses of the word as shown by the WordNet are-

Different senses of the word ‘**बाजी’** are as follows-

1. बाज़ी, बाजी: आदि से अंत तक कोई ऐसा पूरा खेल जिसमें हार-जीत हो या दाँव लगा हो; "श्याम ने हारते-हारते अंतिम समय में बाज़ी जीत ली"

2. पारी, नंबर, बाज़ी, बाजी, बारी, दाँव, दांव, नम्बर: कोई कार्य करने या खेल खेलने का वह अवसर जो सब खिलाड़ियों को बारी-बारी मिलता है; "अब राम की पारी है"

3. शर्त, बाज़ी, बाजी, दाँव, होड़, दांव: किसी विषय के ठीक होने के संबंध में दृढ़ता पूर्वक कुछ कहने का वह प्रकार जिसमें सत्य या असत्य सिद्ध होने पर हार-जीत व कुछ लेन-देन भी हो; "राहुल शर्त जीत गया"

4. बाजी, बाज़ी: जुए में जुआरियों द्वारा दाँव पर लगाया हुआ कुल धन; "बाजी जीतनेवाला जुआरी बहुत प्रसन्न था"

Hypernyms Sequences as taken from the WordNet

The WordNet provides the different Hypernymy sequences possible for each sense of the word. These are as follows-

Sense 1

बाज़ी, बाजी; आदि से अंत तक कोई ऐसा पूरा खेल जिसमें हार-जीत हो या दाँव लगा हो; "श्याम ने हारते-हारते अंतिम समय में बाज़ी जीत ली";

- [खेल, खेल-कूद, क्रीड़ा, खिलवाड़, खेलवाड़]; मन बहलाने या व्यायाम के लिए उछल-कुद,दौड़-धूप या और कोई मनोरंजक कृत्य, "खेल में हार जीत होती रहती है";
- [गतिविधि, क्रिया कलाप, हरकत, कार्य कलाप]; किसी की चाल-ढाल या उसके द्वारा किए जाने वाले कार्यों आदि का रंग-ढंग, "आपको अपने पुत्र की गतिविधियों पर ध्यान रखना चाहिए"
- [काम, कार्य, कर्म, करम, करनी, कृत्य]; वह जो किया जाए, "वह हमेशा अच्छा काम ही करता है"
- [क्रिया]; किसी कार्य के होने या किए जाने का भाव, "दूध से दही बनना एक रासायनिक क्रिया है"

Sense 2

पारी, नंबर, बाज़ी, बाजी, बारी, दाँव, दांव, नम्बर; कोई कार्य करने या खेल खेलने का वह अवसर जो सब खिलाड़ियों को बारी-बारी मिलता है; "अब राम की पारी है";

- [अवसर, मौक़ा, मौका, वक्त, समय, मुहूर्त, औसर, काल, घड़ी, नौबत, बेला, वेला, योग]; ऐसा समय या परिस्थिति जिसमें कोई कार्य या उद्देश्य सहज में, जल्दी या सुविधा से हो सके, "इस काम को करने का अवसर आ गया है";
- [समय, काल, वक्त, जमाना, ज़माना, दिन, वेला, अनेहा, अवसर, अर्सा]; मिनटों,घंटों,वर्षों आदि में नापी जानेवाली दूरी या गति जिससे भूत,वर्तमान आदि का बोध होता है, "समय किसी का इंतजार नहीं करता / आप किस ज़माने की बात कर रहे हैं / समय कैसे बीतता है, कुछ पता ही नहीं चलता"
- [बोध, संज्ञान, ज्ञान, भान, संज्ञा, बोधि, अवबोध]; वस्तुओं और विषयों की वह पूर्ण जानकारी जो मन या विवेक को होती है, "कन्याकुमारी में आत्मचिंतन करते समय स्वामी विवेकानंद को आत्म बोध हुआ"

Sense 3

शर्त, बाज़ी, बाजी, दाँव, होड़, दांव; किसी विषय के ठीक होने के संबंध में दृढ़ता पूर्वक कुछ कहने का वह प्रकार जिसमें सत्य या असत्य सिद्ध होने पर हार-जीत व कुछ लेन-देन भी हो; "राहुल शर्त जीत गया";

- [काम, कार्य, कर्म, करम, करनी, कृत्य]; वह जो किया जाए, "वह हमेशा अच्छा काम ही करता है";
- [क्रिया]; किसी कार्य के होने या किए जाने का भाव, "दूध से दही बनना एक रासायनिक क्रिया है"

Sense 4

बाजी, बाज़ी; * जुए में जुआरियों द्वारा दाँव पर लगाया हुआ कुल धन; "बाजी जीतनेवाला जुआरी बहुत प्रसन्न था";

- [धन-दौलत, दौलत, धन, रूपया-पैसा, पैसा, वित्त, अर्थ, वैभव, विभव, द्रव्य, कौड़ी, इक़बाल, इकबाल, नियामत, नेमत]; रुपया-पैसा,सोना-चाँदी आदि , "धन दौलत का उपयोग अच्छे कार्यों में ही करना चाहिए / उसने अपनी जायदाद में से फूटी कौड़ी भी किसी को नहीं दी";
- [आर्थिक साधन, द्रव्यात्मक साधन, आर्थिक संपत्ति]; अर्थ संबंधी साधन, "केवल आर्थिक साधन से ही सुख नहीं मिलता"
- [साधन, माध्यम, जरिया, ज़रिया, माध्य]; वह जिसके द्वारा या जिसकी सहायता से कोई कार्य आदि सिद्ध होता है, "वाहन यात्रा का साधन है"
- [वस्तु, चीज़, चीज]; वास्तविक या कल्पित सत्ता, "हवा एक अमूर्त वस्तु है"
- [अस्तित्व, मौज़ूदगी, मौजूदगी, वज़ूद, वजूद, संभूति, विद्यमानता, सत्ता, हस्ती]; सत्ता का भाव, "कभी-कभी हमारे मन में यह प्रश्न उठता है कि क्या ईश्वर का अस्तित्व है"

Hyponyms Sequences as taken from the WordNet

Sense 2

पारी, नंबर, बाज़ी, बाजी, बारी, दाँव, दांव, नम्बर; कोई कार्य करने या खेल खेलने का वह अवसर जो सब खिलाड़ियों को बारी-बारी मिलता है; "अब राम की पारी है";

- [हाथ]; हाथ से खेले जानेवाले खेलों में हर खिलाड़ी के खेलने की बारी, "अभी किसका हाथ है?"

The context bag is formed for the given text can be formed by saving the individual words in the array as data structure. Similarly, the semantic bag is prepared by storing each of the synset, hypernyms, hyponyms, meronyms, glosses etc. corresponding to their word sense in their different arrays. In this way, the semantic bag contains the words as follows.

- For sense 1 – {**खेल, खेल-कूद, क्रीड़ा, खिलवाड़, खेलवाड़, हार, जीत, गतिविधि…**}
- For sense 2 – {**अवसर, मौक़ा, मौका, वक्त, समय, मुहूर्त, औसर, काल, घड़ी, नौबत, बेला, वेला, योग, समय, काल, वक्त, जमाना…**}
- For sense 3 – {**काम, कार्य, कर्म, करम, करनी, कृत्य, क्रिया…**}
- For sense 4 – {**धन-दौलत, दौलत, धन, रूपया-पैसा, पैसा, वित्त, अर्थ, वैभव, विभव…**}

The algorithm for the implementation of Lesk algorithm corresponding to our thesis, in the form of Pseudo code, is as follows-

Step 1: Open the files containing words of synsets, hypernyms, hyponyms, holonyms etc

Step 2: Store these files in their corresponding buffers so that words can be extracted from it.

Step 3: Open the file containing words for the formation of context bag.

Step 4: Store this file in the buffer

Step 5: Declare the strings corresponding to each sense and context bag, so that they can be used for taking words related to semantic bag and context bag.

Step 7: Declare the arrays corresponding to each sense as well as for context bag

Step 8: Declare the variables corresponding to each sense for matching.

Step 9: Collect the words corresponding to each sense in their respective arrays.

Step 10: Collect the words corresponding to context bag in its arrays.

Step 11: Compare the arrays of each sense with the context bag and store the number of matched words in each matching variables.

The whole algorithm that is implemented as per the above shows the following overlapping words – **{खेल, हार, जीत, अन्तिम}** corresponding to sense 1 and the number of matching variables corresponding to each sense variables are as follows-

Sense 1 = 4

Sense 2 = 0

Sense 3 = 0

Sense 4 = 0

Chapter 6 Implementation of Problem

The different examples corresponding to the implementation of the Lesk Algorithm corresponding to Hindi WordNet and the word to be disambiguated are as follows-

6.1 Example 1

The algorithm is implemented for the different senses corresponding to the word '**आम**' which is to be disambiguated in the following text-

'आम एक फल है । आम एक विश्वप्रसिद्द स्वादिष्ट फल है। आम का पेड़ भारत विश्व मे सबसे पहले भारत मे पाया गया।आम को फलो का बादशाह भी कहा गया हैं। तमिलनाडू के कृष्णगिरि के आम बहुत ही स्वादिष्ट होते है और दुनिया भर में मश्हूर है।'

Here first of all context bag is prepared by taking three sentences in such a way that it includes-

- The sentence with the word
- The previous sentence
- The following sentence

In this way, the example paragraph to prepare context bag is as follows-

'आम एक विश्वप्रसिद्द स्वादिष्ट फल है।आम का पेड़ भारत विश्व मे सबसे पहले भारत मे पाया गया।आम को फलो का बादशाह भी कहा गया हैं।तमिलनाडू के कृष्णगिरि के आम बहुत ही स्वादिष्ट होते है और दुनिया भर में मश्हूर है।'

The context bag is prepared for the above text in the NetBeans environment as-

आम,एक,विश्वप्रसिद्द,स्वादिष्ट,फल,है,आम,का,पेड़,भारत,विश्व,मे,सबसे,पहले,भारत,मे,पाया,गया,
आम,को,फलो,का,बादशाह,भी,कहा,गया,हैं,तमिलनाडू,के,कृष्णगिरि,के,आम,बहुत,ही,स्वादिष्ट,होते,
है,और,दुनिया,भर,में,मश्हूर,है

The above output is shown as-

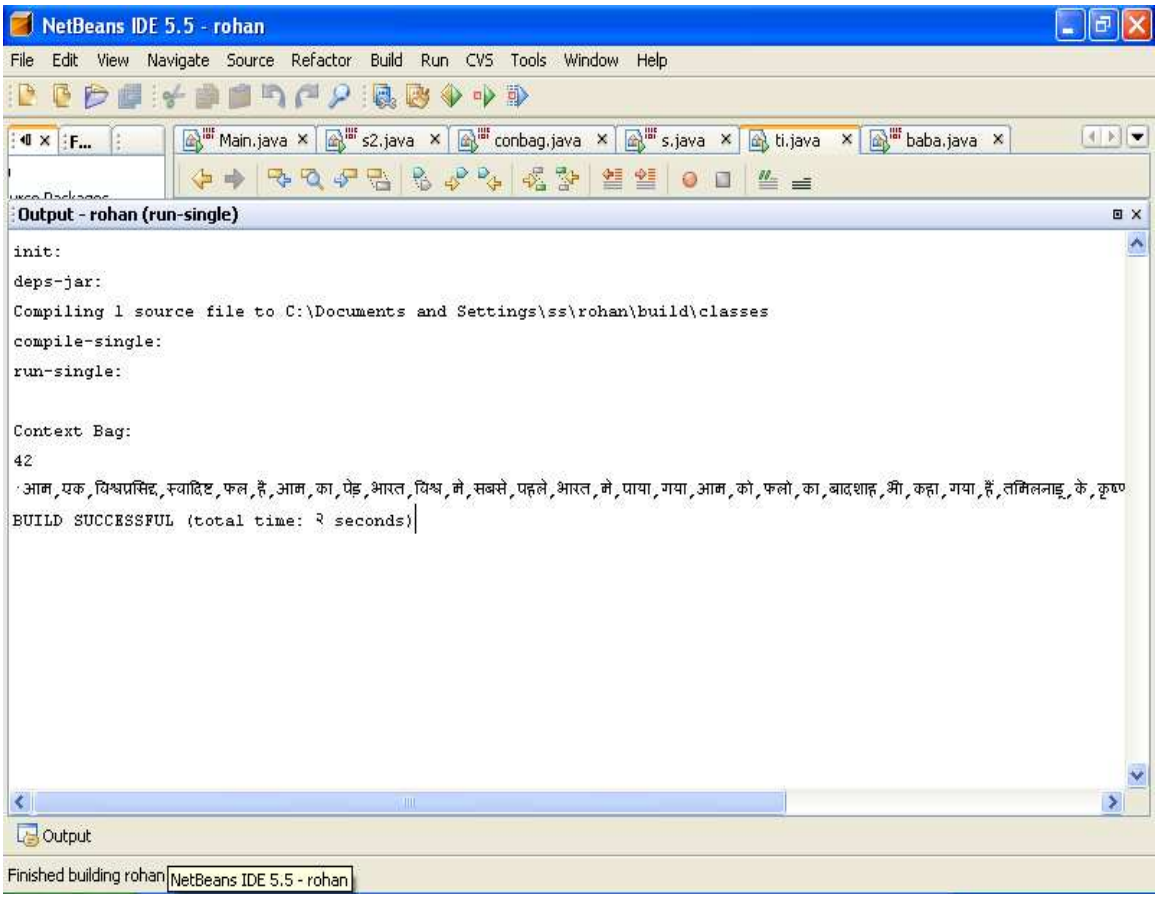

Figure 6.1: Context bag for the word '**आम**'

Later, the semantic bag is created corresponding to each of the sense as follows-

Sense 1:

आम,आम वृक्ष, पिकप्रिय, पिकदेव, पिकबंधु, पिकबन्धु, पिकबंधुर, पिकबन्धुर, पिकराग, मधूली, च्यूत, माकंद, माकन्द, अमरस,अमावट,अंबापोली,खाद्य फल,खाद्य-फल, खाद्यफल, फल, फर, प्रसून, वनस्पति अंग,वनस्पति अवयव,वनस्पति भाग,पेड़-पौधे का भाग,वनस्पति का भाग, प्राकृतिक वस्तु,नैसर्गिक वस्तु, वस्तु, चीज़, चीज, अस्तित्व, मौज़ूदगी, मौजूदगी, वज़ूद, वजूद, संभूति, विद्यमानता,सत्ता,हस्ती,खाद्य वस्तु,खाद्य पदार्थ,खाद्यवस्तु,खाद्यपदार्थ,आहार,खाद्य,भोज्य पदार्थ,आहार पदार्थ, अन्न, पदार्थ, वस्तु, चीज, चीज़, द्रव्य, वस्तु, चीज़, चीज, अस्तित्व.....

Sense 2:

सफेदा,सफेदा आम,सफ़ेदा,सफ़ेदा आम,दशहरी,दशेरी,दशहरी आम,दशेरी आम, दसहरी, दसेरी, दसहरी आम,दसेरी आम,बीजू,बीजू आम,बीज्जू,बीज्जू आम,क़लमी,कलमी,क़लमी आम, कलमी आम, पायरी,पैरी,पायरी आम,पैरी आम,पहेरी,पीरी,नदुसलाई,रसपुरी,पहेरी आम,पीरी आम,नदुसलाई आम,रसपुरी आम,बैंगनपल्ली,बैंगनपल्ली आम,हापुस,हापुस आम,हापूस,हापूस आम,अल्फांसो,अलफांसो,अल्फॉन्सो,तोतापरी,तोतापरी आम,तोतापुरी,तोतापुरी आम,लँगड़ा आम,लँगड़ा,लंगड़ा,लंगड़ा आम,सुंदरी,सुन्दरी,सुंदरी आम,सुन्दरी आम,सुवर्णरेखा,सुवर्णरेखा आम,राजापुरी,राजापुरी आम,सेंदूरी,सेंदरी,सेन्दूरी,सेन्दरी,सेंदूरी आम,सेंदरी आम,सेन्दूरी आम,सेन्दरी आम,सेंदुरियाआम,सेन्दुरिया……..

This is shown in the following figure-

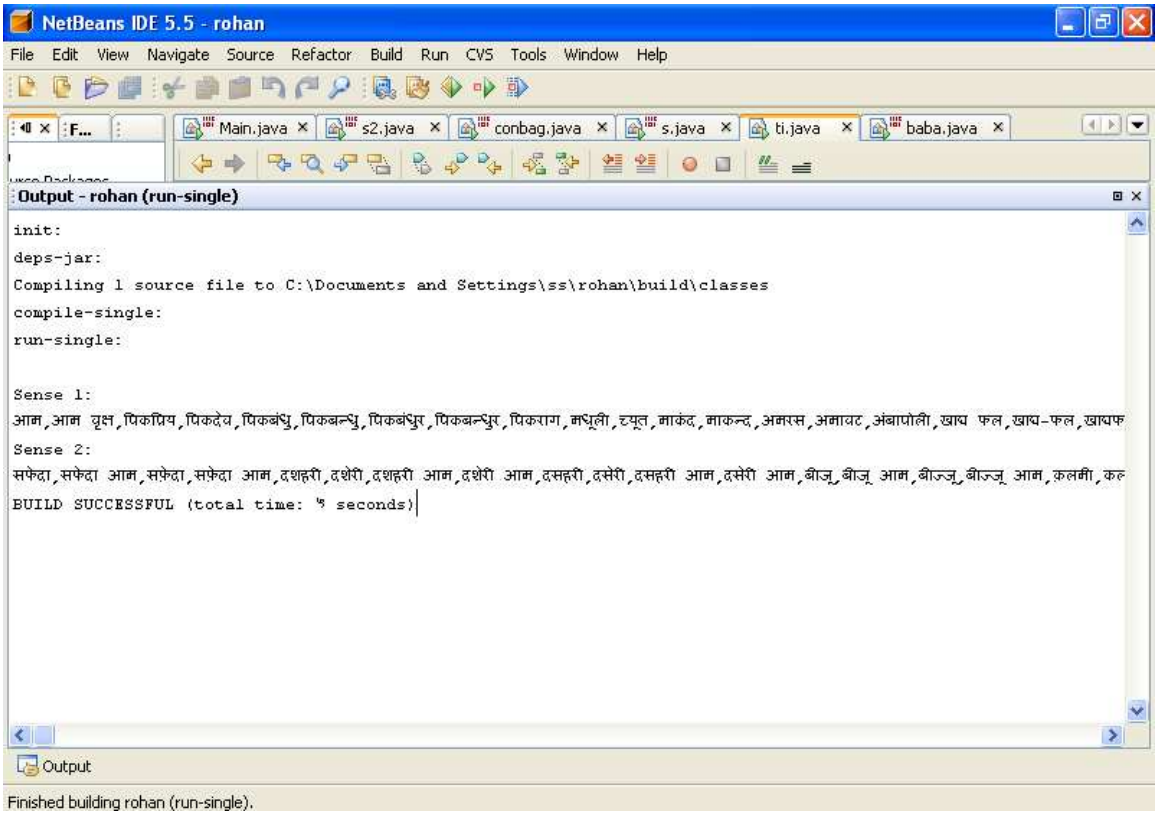

Figure 6.2: Semantic bag for the word '**आम**'

Then we had done comparison of both the bags corresponding to the senses and the result is find out as –

No. of matches (Sense 1) = 2

No. of matches (Sense 2) = 0

This output is shown in the following figure-

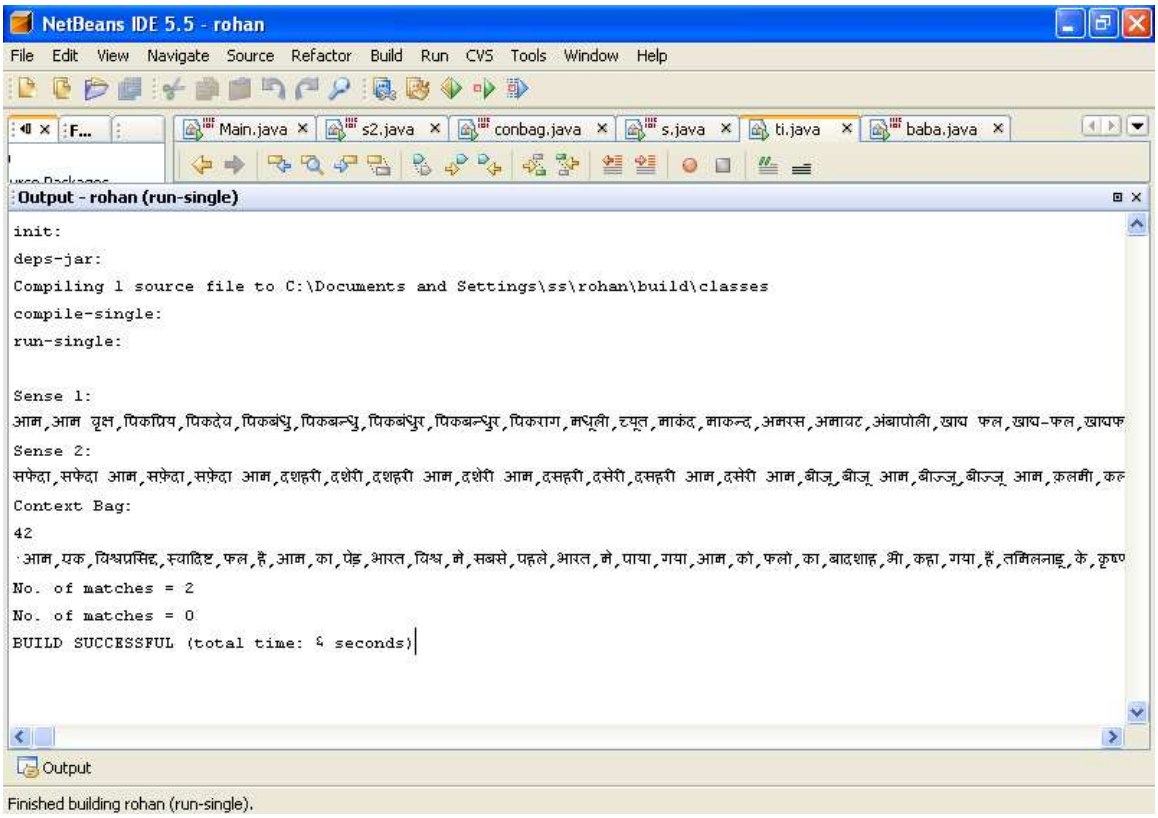

Figure 6.3: Result after matching of arrays corresponding to each sense with the array of context bag for the word '**आम**'

6.2 Example 2

The algorithm is implemented for the different senses corresponding to the word '**योग**' which is to be disambiguated in the following text-

'योग के निपुण अभ्यासी पुरुष को योगी कहा जाता है, स्त्री को योगिनी । आज योग एक विकसित आध्यात्म, दर्शन तथा चिकित्सा पद्धति है जो भारत से फैलकर कई देशों में

पहुंच चुका है । चित वृतियों पर नियंत्रण और उस का विरोध ही दर्शन शास्त्र में योग शब्द से विभूषित हुआ है।'

The context bag is prepared for the above text in the NetBeans environment as-

'योग,के,निपुण,अभ्यासी,पुरुष,को,योगी,कहा,जाता,है,स्त्री,को,योगिनी,आज,योग,एक,विकसित, आध्यात्म, दर्शन,तथा,चिकित्सा,पद्धति,है,जो,भारत,से फैलकर, कई, देशों, में, पहुंच, चुका, है, चित, वृतियों,पर,नियंत्रण,और,उस,का,विरोध,ही,दर्शन,शास्त्र,में,योग,शब्द,से,विभूषित,हुआ,है'

Later, the semantic bag is created corresponding to each of the sense as follows-

Sense 1:

'अंकगणित, अंकविद्या, अंकशास्त्र, अंक-गणित, अंक-विद्या, हिसाब, अंक-शास्त्र, अंक गणित, अंक विद्या, अंक शास्त्र, काम, कार्य, कर्म, करम, करनी, कृत्य,....'

Sense 2:

'जोड़, योग, जोड़ कर्म, योगकरण, जुड़ाई, जोड़ाई, संख्या, अंक, अङ्क,....'

Sense 3:

'समय, काल, वक्त, जमाना, ज़माना, दिन, वेला, अनेहा, अवसर, अर्सा, बोध, संज्ञान, ज्ञान, भान, संज्ञा, बोधि, अवबोध,...'

Sense 4:

'समय, काल, वक्त, जमाना, ज़माना, दिन, वेला, अनेहा, अवसर, अर्सा, बोध, संज्ञान, ज्ञान, भान, संज्ञा, बोधि, अवबोध,...'

Sense 5:

'शास्त्र, धर्मशास्त्र, धर्म ग्रंथ, धर्मग्रन्थ, धर्मग्रंथ, धार्मिक ग्रंथ, धर्म-ग्रंथ, धार्मिक-ग्रंथ,...'

Sense 6:

'काम, कार्य, कर्म, करम, करनी, कृत्य, वह,जो,किया,जाए, वह,हमेशा,अच्छा,काम,ही,करता,है...'

Sense 7:

‘अपयोग, क्रिया, योग, संयोग, अम्ल,और,क्षार,के,योग,से,लवण,बनता,है…’

Sense 8:

‘दर्शन शास्त्र, तत्वशास्त्र, दर्शनशास्त्र, दर्शन, दर्शन-शास्त्र, तत्वज्ञान,वह,शास्त्र,जिसमें,विविध, दर्शनों,का,विवेचन,होता,है,हमारे,गुरुजी,दर्शन शास्त्र,के,अच्छे,ज्ञाता,हैं,…’

Then we had done comparison of both the bags corresponding to the senses and the result is find out as –

No. of matches (Sense 1) = 0

No. of matches (Sense 2) = 0

No. of matches (Sense 3) = 0

No. of matches (Sense 4) = 0

No. of matches (Sense 5) = 3

No. of matches (Sense 6) = 0

No. of matches (Sense 7) = 0

No. of matches (Sense 8) = 2

Chapter 7 Conclusion and Future work

In this thesis, we have developed a tool for Hindi WordNet that can be used for Word Sense Disambiguation (WSD) as part of Natural Language Processing (NLP) tasks for the Hindi Language. We believe that the work presented here is a step in the direction towards the achievement of Natural Language Processing tasks for the Hindi Language. To eliminate the barrier of communication between human beings, the ultimate goal of constructing WordNet is to link all languages in the world together.

7.1 Conclusion

Here, we have done comparison of different approaches that are being used for Word Sense Disambiguation (WSD), these approaches are as knowledge based approaches, machine learning based approaches and hybrid approaches. These approaches are further explained and the different approaches used underneath them are also explained. The major approaches that are being used today are discussed here.

We found the best approach is knowledge based approaches and therefore one of their approach of Michael Lesk *i.e.* Lesk's algorithm is being taken here as an example to show its applicability for Word Sense Disambiguation (WSD) of Hindi language. In order to use this approach, we have taken an example paragraph and created its context bag and then extracted the semantic bag for the word to be disambiguated and we have done the overlap between both bags corresponding to each sense of the word and then the appropriate sense of the word is find out. We have found that our approach is successfully able to resolve the Synonymy, Antonymy, Hypernymy, Hyponymy, Meronymy and Holonymy relations for the different categories of Part-of-Speech.

In this way, Lesk's algorithm can be used for disambiguation and its application will encourage and enable knowledge sharing and translation. If knowledge sharing between Hindi and other languages will be possible, it'll help to cross the language barrier among the regional people. The accuracy can be improved by consulting different linguists of Hindi Language in order to resolve the relations between various synsets of different words.

7.2 Future Work

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

- The context bag that has been used by taking example texts must take many sentences so that accuracy of the word to be disambiguated increased.
- The more examples must be taken so that its accuracy can be tested for different words.
- A more detailed study of the other relationships of verbs, adverbs and adjectives.
- The performance can be surely improved if morphological inflections are handled exhaustively. The system doesn't detect the underlying similarity in presence of morphological variations.
- In this thesis, 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.
- The accuracy of the Lesk's algorithm must be checked on other languages.

References

[1] Prof. Nancy Ide & Jean Véronis , 1998 'Word Sense Disambiguation: The State of the Art.A comprehensive overview'.
http://sites.univ-provence.fr/~veronis/pdf/1998wsd.pdf

[2] Bar-Hillel's,1960 http://www.hutchinsweb.me.uk/Bar-Hillel-2000.pdf

[3] ALPAC report, 1966 'Language and Machines, Computers in Translation and Linguistics), National Academy of Sciences'
http://www.nap.edu/openbook.php?isbn=ARC000005

[4] Mark Stevenson, 'Word Sense Disambiguation: Natural Language Processing Group', University of Sheffield, UK.
http://research.microsoft.com/india/nlpsummerschool/data/files/MarkStevenson%20-%20WSD%20tutorial.pdf

[5] Torbjörn Lager, Semantics.
http://www.ling.gu.se/~lager/kurser/GSLT_Semantics/NLP1_semantics.pdf

[6] Rada Mihalcea and Ted Pedersen, 'Slides from the AAAI 2005 Tutorial - Advances in Word Sense Disambiguation'
http://www.d.umn.edu/~tpederse/WSDTutorial.html

[7] Robert Gaizauskas and Yorick Wilks, 'Information Extraction: Beyond Document Retrieval'
http://citeseer.ist.psu.edu/cache/papers/cs/3650/http:zSzzSzrocling.iis.sinica.edu.twzSzROCLINGzSzCLCLPzSzv5zSz3-2-2.pdf/gaizauskas98information.pdf

[8] Lucia Specia1, Ashwin Srinivasan, Ganesh Ramakrishnan, Maria das Graças V. Nunes, Word Sense Disambiguation using ILP
http://www.dcs.shef.ac.uk/~lucia/publications/Speciaetal_ILP-2006.pdf

[9] Dalal, Nagaraj, Sawant, Shelke, Hindi Part-of-Speech Tagging and Chunking: A Maximum Entropy Approach
http://www.cse.iitb.ac.in/~uma/download/HindiPosTagChunk.pdf

[10] Sinha, Reddy, Bhattacharyya, An Approach towards Construction and Application of Multilingual Indo-WordNet
www.cse.iitb.ac.in/~pb/papers/gwc06_IITB_IndoWN.pdf

[11] Fellbaum, C. 1997, Analysis of a hand tagging task. Proceedings of ANLP-97 Workshop on Tagging Text with Lexical Semantics: Why, What, and How? Washington D.C., USA.

[12] Word Sense Disambiguation By Mitesh M. Khapra

[13] Maximizing Semantic Relatedness to Perform Word Sense Disambiguation by Ted Pedersen,Satanjeev Banerjee,Siddharth Patwardhan.

[14] Manish Sinha, Mahesh Kumar, Prabhakar Pande, Lakshmi Kashyap and Pushpak Bhattacharyya, November, 2004, Hindi Word Sense Disambiguation, International Symposium on Machine Translation, Natural Language Processing and Translation Support Systems, Delhi, India
http://www.cse.iitb.ac.in/~pb/papers/HindiWSD.pdf

[15] Basak Mutlum, Word Sense Disambiguation
http://www.denizyuret.com/students/bmutlum/index.htm

[16] K. Sparck Jones, Synonymy and Semantic Classification, Edinburgh University Press, Edinburgh, 1986.

[17] M. Quillian, Semantic memory, in: M. Minsky (Ed.), Semantic Information Processing, the MIT Press, Cambridge, MA, 1968, pp. 227–270.

[18] Y. Wilks, D. Fass, C. Guo, J. McDonald, T. Plate, B. Slator, Providing machine tractable dictionary tools, Machine Translation 5 (1990) 99–154.

[19] J. Cowie, J. Guthrie, L. Guthrie, Lexical disambiguation using simulated annealing, in: Proceedings of the 14th International Conference on Computational Linguistics, Nantes, France, 1992, pp. 359–365.

[20] J. Veronis, N. Ide, Word sense disambiguation with very large neural networks extracted from machine readable dictionaries, in: Proceedings of the 13th International Conference on Computational Linguistics, Helsinki, 1990, pp. 389–394.

[21] H. Kozima, T. Furugori, Similarity between words computed by spreading activation on an english dictionary, in: Proceedings of the 6th Conference of the European Chapter of the Association for Computational Linguistics, Utrecht, 1993, pp. 232–239.

[22] Y. Niwa, Y. Nitta, Co-occurrence vectors from corpora versus distance vectors from dictionaries, in: Proceedings of the Fifteenth International Conference on Computational Linguistics, Kyoto, Japan, 1994, pp. 304–309.

[23] M. Sussna, Word sense disambiguation for free-text indexing using a massive semantic network, in: Proceedings of the Second International Conference on Information and Knowledge Management, 1993, pp. 67–74.

[24] E. Agirre, G. Rigau, Word sense disambiguation using conceptual density, in: Proceedings of the 16th International Conference on Computational Linguistics, Copenhagen, 1996, pp. 16–22.

[25] S. Banerjee, T. Pedersen, Extended gloss overlaps as a measure of semantic relatedness, in: Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, Acapulco, 2003, pp. 805–810.

[26] Hindi WordNet hwn_docs.doc

[27] Hindi WordNet from Center for Indian Language Technology Solutions, IIT-B, India, http://www.cfilt.iitb.ac.in/WordNet/webhwn

[28] Ramakrishnan G., Prithviraj B. and Bhattacharya P.: 2004, A Gloss Centered Algorithm for Word Sense Disambiguation, Proceedings of the ACL SENSEVAL Conference, Barcelona, Spain.

[29] Dave Shachi, Parikh Jignashu and Bhattacharya Pushpak, 2001, 'Interlingua based English Hindi machine translation and language divergence', Journal of Machine Translation (JMT) ,16(4), pp. 251-304.

[30] English WordNet http://WordNet.princeton.edu

[31] Philip Resnik , Selectional Preference and Sense Disambiguation
http://209.85.175.104/search?q=cache:1_XWLzsoiG0J:www.ims.uni-stuttgart.de/~light/tueb_html/resnik2.ps+Philip+Resnik.+Selectional+preference+and+sense+disambiguation&hl=en&ct=clnk&cd=1&gl=in&client=firefox-a

[32] Sin-Jae Kang, Corpus based Ontology for Word Sense Disambiguation,
http://dspace.wul.waseda.ac.jp/dspace/bitstream/2065/12296/1/PACLIC17-399-407.pdf

[33] David Martinez Iraola, Supervised Word Sense Disambiguation: Facing Current Challenges, http://www.sepln.org/revistaSEPLN/revista/34/13.pdf

[34] Regina Barzilay, Word Sense Disambiguation, MIT. http://people.csail.mit.edu/regina/6864-2005/slides/lec17-4.pdf

Paper Communicated

[1] Parteek Bhatia, Rohan Sharma, Word Sense Disambiguation for Hindi Language, Communicated at National Conference on Emerging Trends In Information Technology, NCEIT-2008, Institute of Engineering and Emerging Technologies, Baddi, Solan