Lexical Disambiguation using English WordNet with Natural Language Toolkit

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

Master of Engineering

in

Software Engineering

Submitted By

Swati Kukreja

(Roll No. 801231027)

Under the supervision of:

Dr. Shalini Batra

Assistant Professor

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

THAPAR UNIVERSITY

PATIALA – 147004

June 2014

Certificate

I hereby certify that the work which is being presented in the thesis entitled, **"Lexical Disambiguation using English WordNet with Natural Language Toolkit"**, in partial fulfillment of the requirements for the award of degree of Master of Engineering in Software Engineering submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Dr.Shalini Batra* and refers other researcher's work which are duly listed in the reference section.

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

Swati Kukreja

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

Dr. Shalini Batra
Assistant Professor,
Computer Science and Engineering Department,
Thapar University,
Patiala.

Countersigned by

(Dr. Deepak Garg)
Head
Computer Science and Engineering Department
Thapar University
Patiala

(Dr. S. K. Mohapatra)
Dean (Academic Affairs)
Thapar University
Patiala

Acknowledgment

It is a great pleasure for me to acknowledge the guidance I have received from **Dr. Shalini Batra**, Assistant Professor, Computer Science and Engineering Department, Thapar University, Patiala. I am thankful for her continual support throughout the thesis.

I am also thankful to **Dr. Deepak Garg**, Head of Computer Science & Engineering Department and **Dr. Damandeep Kaur**, P.G. Coordinator, for the motivation and inspiration that triggered me for the thesis work.

I would also like to thank the staff members and my colleagues who were always there at the need of an hour and provided with all the help and facilities, which I required, for the completion of my thesis work.

Most importantly, I would love to express my heartiest gratitude to my parents and the almighty for showing me the right direction out of the blue, to help me stay calm in the oddest of the times and keep moving even at times when there was no hope.

Swati Kukreja

(801231027)

Abstract

The expansion of the Information Technology, has given rise to the emergence of the great amounts of the unstructured data like the Web pages, document warehouses, blog corpora and many more. Consequently, there is arising an increasing demand to treat the massive information through the means of automated methods of lexical disambiguation i.e. Word Sense Disambiguation (WSD). It is a tedious task to deal with, as to resolve this issue one need to overcome the complexities of language and it is a complicated affair to recognize a semantic layout from the unstructured sources of the text and still the researches are continued in this field so as to resolve the issue at the best possible level of accuracy. WSD is considered as an artificial intelligence problem having the capability to recognize the meaning of the words, which are in the context of the given text. The issue of lexical disambiguation existing in a sentence is resolved here with the help of the Lesk Algorithm, with the modification that, the Part of Speech (POS) of the ambiguous word is predicted with the help of Decision Tree Classifier, which helps in resolving the issue of accuracy to determine the correct POS to a great extent, and this even aided the Lesk Algorithm to limit its effort to just one Part-of-Speech of the ambiguous word only. The output is yielded in the form of the 'sense' which gives a best match with the context of the sentence in which it is mentioned. Experimental results showed that the accuracy to determine the sense of a word was improved.

The resultant sense obtained as an output was further judged by the computation of its similarity score (i.e. Wu-Palmer similarity score and Jiang-Conrath similarity score) with the words in the context bag. The modified Lesk Algorithm further facilitated in getting the correct translation of the ambiguous words to the languages named Punjabi and Hindi.

Table of Contents

List of Figures

List of Tables

Chapter1
Introduction

There are some words in human language which exhibit ambiguity and this is the reason these words can be explicated in multiple senses depending upon the sentence context in which they are found. For example, if the following two sentences are taken into consideration:

1. The workers at the **plant** were over worked.
2. The **plant** was not getting enough sunlight.

The meaning of the word "Plant" varies according to the context of the sentences, in the very first sentence the word plant can be interpreted as an industrial unit where as the plant in the second sentence refers to vegetation. Human mind can easily interpret the difference in the senses reflected in each sentence, but machines require textual analysis in order to ascertain the underlying sense of the ambiguous word [1]. This technique of computational identification of the meaning of the words in the context is called Word Sense Disambiguation i.e. the words in the sentence should be tagged ideally with the meaning they reflect. WSD is counted as an AI- complete problem which means that this task is as tough as the most tedious problems in the domain of Artificial Intelligence [1, 2].

1.1 More about Word Sense Disambiguation

Word Sense Disambiguation is coined as one of the important issues in the field of Machine Translation since 1940s. This problem can be called as a mandatory subtask for most of the NLP applications [3, 4]. Research has been conducted to unearth the rudiments needed for lexical disambiguation which is in terms of: the local context in which a target word resides, the statistical distribution of all the words and its senses, and the knowledge sources [5]. But because of lack of availability of the computational resources, congestion was reached and appreciable progress was not observed. So, with the advancement in lexical resources, furtherance was observed in WSD and more and more number of subtle AI based approaches were introduced to tackle the problem. With the breakthrough in Machine Learning and statistical modeling, statistical methods were employed for automatic WSD. Also, significant

advancement was established on the problem through the creation of regular ranking and analysis campaigns, which served the process of monitoring the improvement being created in the performance of the automated WSD systems, and also directed to the new concerns with respect to the task. WSD is still counted as a tedious problem, mainly because of lack of availability of knowledge bases which are mandatory from a sense disambiguation view point.

Firstly, the task lends itself to distinct fundamental questions, like its outlook in the context in which it is mentioned i.e. delineation of sense (ranging from finite set of senses to the rule-based generation of the new senses), the granularity of various sense inventories (varying from precise distinctions to the homonyms), the domain-oriented versus the unbounded nature of content, the set of target words to be disambiguated (one target word per sentence vs. "all-words" settings), etc. Second, WSD is heavily dependent on the knowledge sources. In fact, the outlook of the view of approach of any WSD system can be outlined as follows: a bag of words or a sentence is given, technique is applied which employs one or more knowledge sources to link the most suitable senses with the words in context. There exist wide varieties of knowledge sources i.e. these may range from either unlabeled or annotated with word senses, to more structured resources, like semantic networks, machine readable dictionaries etc. Without knowledge base, it would become unfeasible for both humans and machines to recognize the sense of a word. So, it can be said that Knowledge is considered as a fundamental part of WSD [2]. Knowledge sources provide data which are essential to associate senses with words. There exist various forms of knowledge sources for instance the corpora of the text, either annotated or labeled with the word sense or the structured resources like the semantic networks. So, these provide the opportunities for resolving the disambiguation issues to a great extent [6].

So far, domain information for sense disambiguation is discussed. Current methods for WSD take advantage of a knowledge source named WordNet 2.0 .A novel framework is proposed which models information from this knowledge source into the constraints and use them collectively to resolve the issue of disambiguation. In all the methods, knowledge base named English WordNet is used as a sense inventory.

1.2 WordNet

WordNet lies in the category of the knowledge sources and will serve the purpose of resolving the issue of WSD here. It is a large online lexical database for English language which entails nouns, verbs, adverbs and adjectives grouped into the sets of the synsets which are interlinked by the means of conceptual lexical and semantic relations [7]. It gives broad coverage of lexical of English words, which are organized as taxonomic hierarchies. Synset constitutes the smallest unit of WordNet which carries all the information regarding the words, like the concept, offset, pointer etc. The synsets in the WordNet hiererarchy are connected to one another by the number of relationships such as hypernym, homonym, meronym, synonym, antonym and many more. Amongst the synsets the most frequently mentioned relation is a hypernymy –hypnomy relation also called as IS-A relation [8].

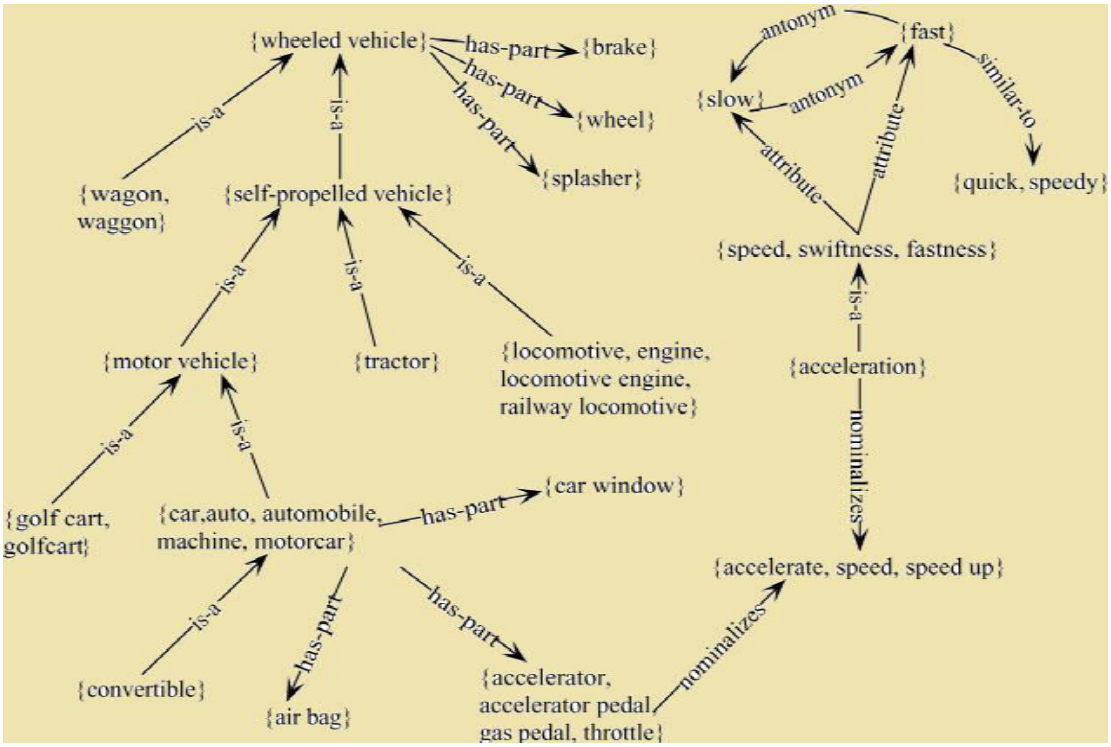


Figure 1.1: Synsets of the word 'automobile' [6]

Figure 1 and Figure 2 illustrates the manner in which the term 'automobile' is expressed with the synsets, it can be determined that each sense of word unambiguously determines the single synset and a synset comes up with the information such as Part-of-Speech, gloss and all the lexical relations.

Senses in WordNet are elucidated by synsets (i.e. the synonyms sets). Each synset comprises of the set of synonyms which represent a sense defined by a gloss [9]. Some examples are listed below:

>>> wn.synsets ('automobile')

[Synset ('car.n.01'), Synset ('automobile.v.01')]

Detailed description of synsets of word "automobile"

>>> wn.synset ('car.n.01').definition

'a motor vehicle with four wheels; usually propelled by an internal combustion engine'

>>> wn.synset ('car.n.01').examples

['he needs a car to get to work']

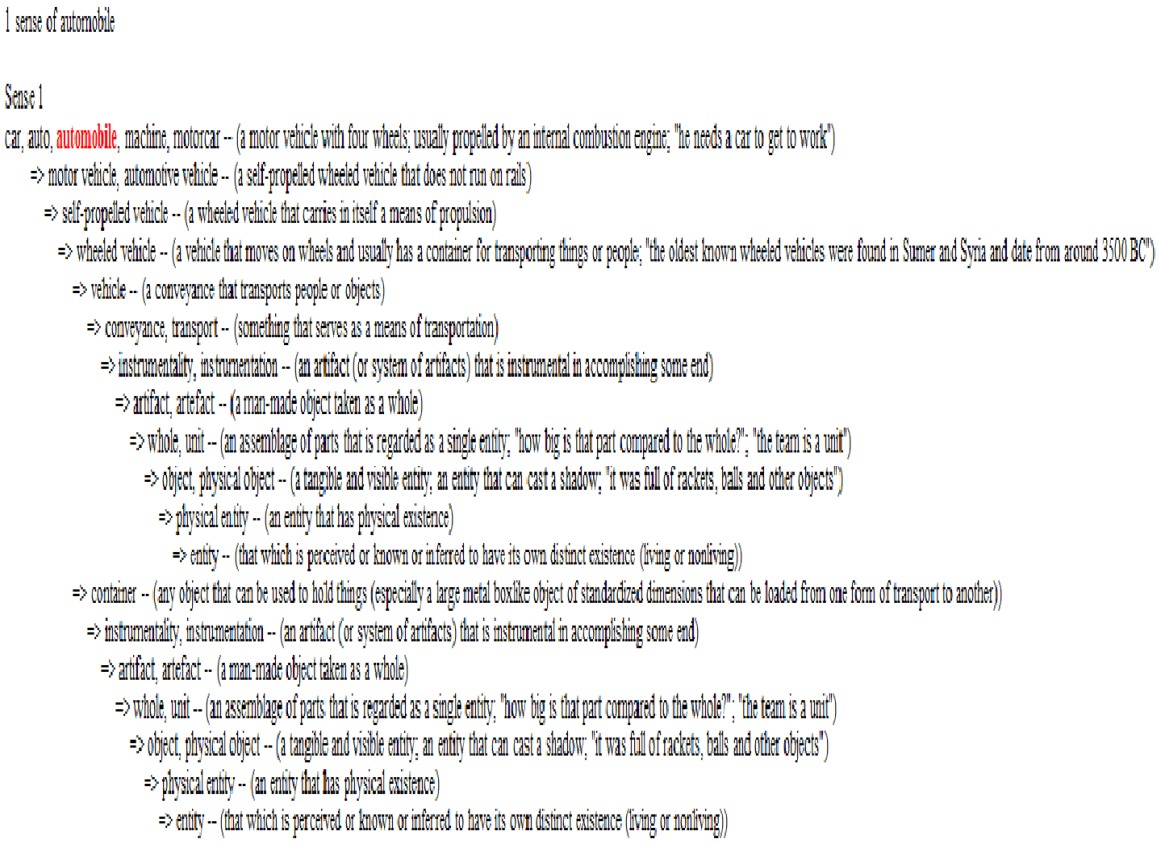


Fig 1.2: Hypernym hierarchy of first synset of word 'automobile'

1.3 Steps for handling of WSD

Word Sense Disambiguation (WSD) incorporates the involvement of association of a word mentioned in a text with the meaning or definition which suits well to the context in which it is mentioned [10]. This task is inevitably accomplished in two steps mentioned as follows:

Step 1: Ascertainment of the different senses of every word present in the context;

- Listing of all the senses reflected by a word;
- Listing of all the associated words or categories (e.g., synonyms, as mentioned in thesaurus);
- An entry which comprises of translation of words to different languages etc.

Step 2: Assignment of the relevant sense to every word.

- Text or a discourse in which an ambiguous word is present provides great help in disambiguation of its sense as it is the context which will determine the exact sense and sentiment reflected by a word.
- Knowledge sources provides an insight of senses that can reflected by an ambiguous word, so without knowledge source a WSD system cannot get a wide exposure of all the senses reflected by a word and hence pick the most relevant sense which best complement the context or discourse [10] .

So, the process of WSD relies heavily on the knowledge to resolve the lexical ambiguity i.e. knowledge source will help in figuring out the most appropriate sense which matches well to the context in which it is mentioned.

1.4 Approaches of Disambiguation

There exist following approaches of Word Sense Disambiguation:

1.4.1Knowledge-BasedDisambiguation

Knowledge based disambiguation has emerged as a powerful mechanism to overcome WSD. It fetches the information from the manually developed lexical resources like dictionaries, thesauri etc for the purpose of disambiguation. This is further comprised of two approaches:

1.4.1.1 WSD using Selection Preferences

This technique is quoted as the most recurrently utilized approach for WSD. The technique widely used in this domain is named as 'Katz and Fodor's use of Boolean selection restrictions to constrain semantic interpretation. This applies restriction to the selection of senses for a word mentioned in the context sentence. For example, the word the 'bank' in a sentence 'She goes to bank to deposit her money' exhibits the second sense i.e. 'a financial institution', so the scope is restricted to this sense exclusively.

1.4.1.2 Overlap- based approaches

The techniques lying under the category of this approach employs the Machine Readable Dictionary (MRD) to resolve the issue of Word Sense Disambiguation. The synonym set which provides the maximum overlap with the words in the context is assumed to exhibit maximum similarity with the context [10].

1.4.2 Machine Learning Based Approaches

It is further categorized into three different approaches:

1.4.2.1 Supervised Approaches

It is based on a convention that the context provides sufficient indications about the sense reflected by an ambiguous word present in a context sentence. The methods following supervised approach rely on considerable amount of manually sense-tagged i.e. labeled corpora for training, which requires great amounts of inputs in terms of time and efforts [11].

1.4.2.2 Semi-supervised Algorithms

It is a learning paradigm which makes the use of both labeled and unlabeled data for learning purpose, it follows the belief that making use of both types of corpora (i.e. labeled and unlabelled) can lead to the change in the learning behavior which can give rise to new and improved results [12]. The semi-supervised learning approaches are of great interest to the researchers as these can utilize the readily available unlabeled data to upgrade the supervised learning methods when there exists scarcity in labeled data sources.

1.4.2.3 Unsupervised Algorithms

This approach combines the primacy of both knowledge based and supervised approaches. Like supervised approaches, evidence is extracted from the corpus and like that of knowledge based approaches these algorithms doesn't require the tagged corpus [12].

1.4.3 Hybrid Approaches

This approach is an amalgamation of rule based techniques and statistics based techniques which come under the domain of machine learning. Overall it can be said that these approaches are both knowledge-based and corpus based for instance Sin-Jae Kang used an approach in which he considered secured dictionary information as

context information and semi-automatically constructed ontology as an external source. Through this hybrid approach he was able to overcome the lexical disambiguation of the ambiguous words present in a context sentence [12].

1.5 Contributions

There exist four-fold key contributions to this field. Taking algorithmic aspect in consideration, novel approaches for the lexical disambiguation are introduced. Secondly, the efficacy of semantic similarity measures is described for the lexical distinctions. Furthermore, a novel technique is proposed for the domain specific lexical disambiguation revealing the significance of information of domain in lexical distinction. Finally, novel framework is derived for modeling of information from the knowledge source named WordNet and the information retrieved is collectively used for resolving the issue of WSD.

1.6 Thesis Overview

The rest of the thesis is structured as follows: In chapter 2, brief review of literature is given, which entails the information about various approaches meant to overcome the issue of WSD. In chapter 3, Gap Analysis and Problem Statement are described and in chapter 4, Objectives and Methodology proposed to achieve these objectives are presented. In chapter 5, domain specific WSD method is discussed and is evaluated and finally novel framework is presented for the usage of information from the knowledge source (i.e. WordNet here) with the intent to overcome WSD. Chapter 6 gives the detailed description of the results obtained after the application of the mechanism. Chapter 7 provides the conclusion, listing all the main points, along with an insight of potential directions for the further research in future.

Chapter 2

Literature Review

There exists a wide variety of approaches to overcome Word Sense Disambiguation. Different algorithms developed under these approaches are discussed below.

2.1 Knowledge Based Approaches

WSD relies heavily on the knowledge to resolve the lexical ambiguity i.e. knowledge source will help in figuring out the most appropriate sense which matches well to the context, in which it is mentioned. Knowledge is considered as an elemental part of WSD [2]. Knowledge sources yield data which is significant to equate senses with words. There exist various forms of knowledge sources for instance the corpora of the text, either annotated or labeled with the word sense or the structured resources like the semantic networks. So, these provide the opportunities for resolving the disambiguation issues to a great extent [6]. WordNet lies in the category of such knowledge sources and will serve the purpose of resolving the issue of WSD here.

Knowledge sources may take grammar rules into consideration or they may employ hand coded rules for disambiguation. Nowadays, knowledge sources and thesaurus are available in Machine Readable Dictionaries format (MRD) like that of Oxford English Dictionary, Collins, Longman Dictionary of Ordinary Contemporary English (LDOCE) [13]. These knowledge sources comes up with different forms, for instance, Roget Thesaurus adds synonyms information; and Knowledge sources like WordNet, EuroWordNet take semantic networks into consideration and these knowledge rich sources are available in English language [14]. Dictionaries available in Machine Readable Format entails list of denotations, definitions (for all the synsets), and the example sentences for most of the words mentioned. While thesauri includes various explicit synonymy relations existing between the meaning of words and the semantic network takes into consideration all the relations such as Hypernyms/hyponyms (IS-A), meronyms/holonyms (PART-OF), entailment, antonyms etc.

The different approaches discussed as follows:-

2.1.1 Using Selection Preferences and Arguments

Example using word named 'provide'-

Sense 1 -This airline provides lunch services during the flight.

- Provide(verb)
- Agent
- Object - edible

Sense 2 - This airline provides the services for the sector between Agra and Delhi.

- Provide(Verb)
- Agent
- Object - Sector

This approach involves comprehensive enumeration of:

- Arguments – i.e. the structure of verb forms of a word here
- Argument's various preferences
- Description of relation existing between various words in a sentence

2.1.2 Using Overlap Based Approaches

This approach makes the use of Machine Readable Dictionary [15].

- It takes into consideration the overlap between various senses of an ambiguous word (i.e. the sense bag) and the attributes of the words available in the context (called as context bag).
- These features could be gloss, example sentences, hyponyms, hypernym etc.
- These features can also be allotted some weights.
- The sense which provides maximum overlap is coined as the contextually appropriate sense.

WordNet, Thesaurus etc are counted as Machine Readable Dictionaries. Thesaurus provides the information about all the relationships existing between words. Disambiguation based on thesaurus makes the use of labeling of semantic category which is provided by a dictionary with the subject categories. The thesaurus used most frequently with the intent to overcome WSD is Roget's International Thesaurus and it was transformed into machine-tractable form in the year 1950. Thesaurus based Disambiguation assigns semantic category to the ambiguous word which is relevant to the context of the sentence and this category facilitates in determining the

correct output. There exist many algorithms which takes Overlap Based Approaches into the consideration. 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.

2.1.2.1 Lesk's Algorithm

The Lesk algorithm follows the principle of Overlap Based Approaches which is explained as follows-

1. For a polysemantic word with different senses needing disambiguation, a set of context words surrounding is collected. Let this collection be C, the context bag contains all the words with which a polysemantic word is surrounded.

2. For each sense's' of word say, 'w' here:

- Let B be the bag of words obtained from the
    - Synonyms and Glosses
    - Example Sentences
    - Hypernyms
    - Glosses of Hypernyms
    - Example Sentences of Hypernyms
    - Hyponyms
    - Glosses of Hypernyms
    - Example Sentences of Hypernyms
    - Meronyms
    - Glosses of Meronyms
    - Example Sentences of Meronyms
- Extent of overlap existing between C and B is measured by making the use of the intersection similarity measure. Output is obtained in the form of a sense's' which provides the maximum overlap [16].

2.1.2.2 Walker's Algorithm

In this algorithm, thesaurus category for each sense of an ambiguous word is found out. Similarity score of each sense and the context is computed. A context word will

increment the score of the sense by 1 in the thesaurus grouping, if the word shows a match with the sense [15].

Black applied this technique to five discrete words and attained accuracy around 50%.

2.1.2.3 WSD using Conceptual Density

Sense is selected on the basis of relatedness to the context. Similarity is coined in the terms of conceptual distance here. Various senses reflected by an ambiguous word present in the context are taken into consideration and the similarity is computed in terms of the conceptual distance (i.e. closeness between the senses reflected by the ambiguous word and represented by its context words are). This approach makes the use of the structured hierarchical semantic net (WordNet) for the computation of the conceptual distance. Smaller the conceptual distance is, 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].

2.1.3 Development in the field of Knowledge Based Approaches

Dictionaries are considered as the best possible sources of information for Computational methods concerned with word meanings [17]. For instance, technique introduced by Sparck–Jones identified synonyms by clustering terms based on the content words that occurred in their glosses [18]. Brief descriptions of these approaches are discussed further.

2.1.3.1 Major Algorithms

Brief Description of some of the major algorithms is mentioned below:

2.1.3.1.1 Quillian's Approach

Quillian described the strategic means through which the content of a machine readable dictionary can be used to make conclusion about the meaning of the words. Under this approach it was proposed that the contents of a thesaurus or a dictionary can be represented in the form of semantic network. Different meanings associated with a word were represented as a node of a graph, and that node is connected to the set of words that defined the concept in the dictionary. This interconnection leads to the creation of a large network of words [15].

Once the structure is defined for a variety of concepts, spreading activation is used to find the intersecting words or concepts in the definitions of a pair of words, the intersection of the various senses of the ambiguous word with the context is carried

out and this facilitates in giving an insight of the relation existing between various senses and the context. For instance, it was found that word 'cry' and 'comfort' have the word 'sad' as a common word in their glosses, which describes that these words hold a relation with this emotion. This mechanism describes the early usage of gloss overlap to figure out the best sense out of all the senses.

2.1.3.1.2 Lesk's Approach

The Lesk algorithm has been consistently employed as the solution to the issue of Word Sense Disambiguation. It determines the meaning for ambiguous word by comparing the gloss of various senses of a word with the words present in the context bag [15]. It is based on the fact that the sense which best suits to the context will provide maximum amount of intersection. In particular, it can be mentioned that Lesk algorithm treats the glosses of various senses as an unordered bag of words, and simply tally the number of the words that overlapping between the words residing in the sense bag and the other words lying in the context bag. So, the winner sense is the one which provides maximum overlap with the words in the context bag.

There are still existing some grounds which illustrates the fact that Lesk algorithm despite of exhibiting accurate results still require some refinement for instance, the Lesk algorithm should be used to disambiguate all the words in a sentence at once and it should proceed sequentially, from one word to another. If it proceeds sequentially, then to what extent the previously assigned senses should influence the output of the algorithm. The words located farther from the target word are given less importance than the words residing nearby; this may affect the accuracy to output the correct sense [19]. Lesk also postulated that the length of the glosses may affect the accuracy of the determining the winner sense to a great extent.

2.1.3.1.3 Wilks' Approach

Wilks conveyed that the dictionary glosses may not be the reliable mode always to overcome Lexical Disambiguation. A context vector approach was developed which expanded the glosses with all the related words, which caused the intersection between larger set of words and consequently the accuracy to determine the correct sense was improved. This algorithm became a standard for WSD in the early 1990s, and Longman's Dictionary of Contemporary English (LDOCE) was employed as a knowledge source [18].

2.1.3.1.4 Cowie's Approach

Cowie suggested that while the Lesk algorithm is capable of disambiguating all the words in a sentence simultaneously, but the computational complexity of this technique is higher which makes it difficult to implement in practice. So in this technique, simulated annealing is employed to simultaneously search for the senses of all the words present in the sentence. A positive point associated with simulated annealing is that it is capable of finding an optimized solution to the assignment of senses to the words present in the sentence that too without an exhaustive search [20].

2.1.3.1.5 Veronis and Ide's Approach

It is introduced as an improvement over Quillian's approach to resolve WSD. Graphical approach is employed here in a manner that the words of the sentence are treated as a node and each node is connected to its sense nodes, which are further connected with their respective definition nodes. So, the process of disambiguation is performed by spreading activation, so 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 [21].

2.1.3.1.6 Kozima and Furugori's Approach

Kozima and Furugori constructed a network from the glosses that consisted of nodes representing the controlled vocabulary, and links to show the co-occurrence of these words in glosses. They defined a measure based on spreading activation that results in a numeric similarity score between two concepts [22].

2.1.3.1.7 Nitwa and Nitta's Approach

Under this approach, context vectors obtained from the co–occurrence statistics of large corpora were compared with the vectors obtained from the path lengths (i.e. the count of the edges) in a network that represented their co– occurrence in the dictionary definitions. In the latter case, Quillian–style network is constructed where words that occur along with each other in a definition are inter-connected. Wilk's context vector method of disambiguation was evaluated, and it was found that the dictionary content is more suitable source of co–occurrence information than the other corpora [23].

But mostly WordNet is considered as a reliable knowledge source for the purpose of WSD [24].

2.1.3.1.8 Banerjee and Pedersen's Approach

Under this approach Lesk algorithm was modified in order to avail the benefit of the network of connections provided in WordNet. The glosses of words present in the context along with its hypernyms are taken into consideration which helps in getting accuracy in results in the form of correct sense [25].

2.1.3.1.9 Pedersen et al. Approach

Pedersen et al. introduced an algorithm that uses measures of semantic similarity to conduct lexical disambiguation [26]. This algorithm makes the use of 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. The overlaps in the definitions will indicate the relatedness. This algorithm performs disambiguation using any measure that returns a relatedness or similarity score for pairs of word senses.

2.1.4 Comparison and Drawbacks of Knowledge Based Approaches

The comparisons and drawbacks of the above explained approaches are mentioned in table 2.1 and table 2.2 respectively.

Table2.1: Comparison of various Knowledge based approaches [19]

Algorithm	Accuracy
WSD using selection preferences	44%on brown corpus
Lesk's Algorithm	50-60% on short sentences
WSD using conceptual Density	54% on brown corpus
Walker's Algorithm	50% when tested on 10 highly polysemous English words

Table2.2: Drawback of Knowledge Based Approaches [19]

Algorithm	Drawbacks
WSD using selection preferences	Requirement of exhaustive knowledge sources
Overlap based approaches	• The dictionary definitions are small • The distributional constraints of different word senses are rarely taken into account. • Proper nouns are not found in the MRDs, so the sense exhibited by the proper noun can't be used for disambiguation.

2.2 Machine Learning Based Approaches

Three different kinds of algorithms are covered under this approach [12]:

- Supervised Algorithms- It takes labeled training set into consideration. The learning system consists of training set of the 'feature-encoded inputs' and the appropriate sense label.
- Semi-supervised Algorithms-These sorts of algorithms take unlabeled corpora into the consideration. The learning system comprises of training set of 'feature-encoded inputs' but doesn't consist of label for the senses.
- Unsupervised Algorithms- Earlier approaches disambiguated each word in segregation. But under this approach, connections existing between the words in a sentence can facilitate in conducting disambiguation. Graphical approach is followed which provides a way to study the relations existing between the different entities, considering the relations existing between senses of various words.

2.2.1 Supervised Learning

Under this type of learning mechanism, a set of examples is collected which illustrates the different possible outcomes or classification of an event. This collection makes a system recognize the patterns in the different instances related with each class of an event. These patterns are then generalized into rules and these rules are further executed to categorize the new event [28]. In this way, a classifier is induced from manually sense-tagged text using machine learning techniques. The resources such as

Sense Tagged Text, Syntactic Analysis (POS tagger, Chunker, Parser), Dictionary (implicit source of sense inventory) make use of this mechanism.

The methodology of supervised learning is as follows-

- A sample of training data is created where a specified target word is annotated manually with a sense from the set of possibilities which are already pre-determined i.e. one tagged word per instance/lexical sample disambiguation
- Then, a set of features is collected with the help of which context can be represented. Co-occurrences, POS tag, collocations, verb-object relations, etc. Sense-tagged training instances are converted to feature vectors.
- Machine learning algorithm is applied to induce a classifier.
- A held out sample of test data is converted into feature vectors.
- Classifier is applied to test the instances for the assignment of a sense tag.
- Once data gets transformed into the feature vector form, any supervised learning algorithm can be employed to use. Some of the famous algorithms are listed below :
    - Support Vector Machines
    - Nearest Neighbor Classifiers
    - Decision Trees
    - Decision Lists
    - Naïve Bayesian Classifiers
    - Perceptrons
    - Neural Networks
    - Graphical Models
    - Log Linear Models

2.2.1.1 Major Algorithms

Brief description of these algorithms is as follows-

2.2.1.1.1 Naive Bayesian Classifiers

- Naïve Bayesian Classifier is quite famous in Machine Learning sector for its performance across different range of tasks and WSD is also not an exception to this facet [27].

- Under this, conditional independence among the features is assumed, when the sense of a word is given. The structure of the model is assumed and the parameters are estimated from the training instances.
- When it comes to the application of this classifier to resolve the issue of WSD, features are often termed as "a bag of words" that emerges right from the training dataset. This usually entails thousands of features which are binary in nature and hence indicate whether a word is present in the context of the target word or not.
- This algorithm suffers from the problem of sparseness of data.
- The scores are procured as a result of the product of probability values, but some factors may lower the value of overall score for a sense.
- This requires training of large number of parameters [30].

$$\hat{s} = argmax_{s \in senses} P_{r(s|V_w)} \quad (2.1)$$

where 'V_w' is a feature vector consisting of-

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

2.2.1.1.2 Decision Lists and Trees

- This is widely used in Machine Learning. It is employed as a word-specific classifier and a separate classifier is required to be trained for each word. It uses the single most predictive feature which eliminates the drawback of Naïve Bayes [27, 31].
- It is established on the basis of 'One sense per collocation' property.
  - It is capable of providing accurate clues in terms of sense of the target word.
- The disambiguation problem is represented as a series of questions (presence of feature) which makes a system predict the sense exhibited by an ambiguous word with respect to the context.

- Tree form of the problem statement helps is predicting the best sense from the set of the senses exhibited by a word.

- A small and much more refined set of features are employed to predict the sense.
- It is easy to implement and more descriptive in nature.

The algorithm to create decision List for the process of Lexical Disambiguation was provided by Yarowsky. The algorithm for the creation of decision lists [32] is as follows -

- First step involves the identification of collocation features from the sense tagged data.
- Large set of collocations is collected for the ambiguous word.
- Word-sense probability is calculated for distributions of all such collocations.
- Calculation of the log-likelihood ratio

$$AB_s(Log\left(\frac{P(Sense_A|Collocation_i)}{P(Sense_B|Collocation_i)}\right)) \quad (2.2)$$

- The higher the log-likelihood is, the more predictive the evidence will become.
- Collocations are kept in a sorted manner in a decision list, with the most predictive collocations ranked highest, for instance, ID3 and C4.5 constitute to the well known decision tree learning algorithms.

2.2.1.1.3 Exemplar Based WSD (K-NN)

It is also known as word-specific classifier. This algorithm doesn't work for the words which are not present in the corpus. Diverse set of features are employed which includes noun-subject-verb pairs and morphological pairs [29]. This classifier is used for the disambiguation process, following steps are followed:

Step 1: A training set is constructed for each sentence comprising of the ambiguous word:

- Part-Of-Speech of an ambiguous word and rest of words of the words in the sentence.
- Local collocations
- Co-occurrence vector
- Morphological features
- Subject-verb syntactic dependencies

Step 2: A test example similar to the input sentence comprising the ambiguous word is constructed.

Step 3: Comparison of the test example with the training set is conducted and in this manner the k-closest training examples are picked.

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

2.2.1.1.4 WSD Using Support Vector Machines

It is also regarded as a word-sense specific classifier. It has evolved as an improvement over the baseline accuracy. Diverse set of features are employed in this case. SVM classifier is binary in nature which finds a hyper plane with the largest margin which further facilitates in the separation of training examples into 2 classes [29]. Since SVM classifiers are binary in nature, so there arise a necessity to create separate classifier for each sense of the word:

Training Phase:

SVM is trained for every sense of a word using the following features [29, 30]:

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

Testing Phase:

A test example similar to the test sentence and is fed as an input to the classifier. Then correct sense is chosen on the basis of the label returned by each classifier.

2.2.1.1.5 WSD Using Perceptron trained HMM

Under this approach, Word Sense Disambiguation is treated as a sequence labeling task. A discriminative Hidden Markov Model is employed which is trained using the following features[16]:

- POS of an ambiguous word as well as POS of neighboring words.
- Local collocations
- Shape of the word and neighboring words

This technique lends itself to the Named Entity Recognition (NER) and assigns labels like ‘person’, ‘location’, ‘time’ etc. which are already included in the super sense tag set.

2.2.1.2 Comparison of Supervised Approaches

Comparison of various supervised approaches is represented in table 2.3 below:

Table 2.3: Comparison of various supervised approaches [28]

Approach	Average Precision	Average Recall	Corpus	Average Baseline Accuracy
Naïve Bayes	64.13%	Not Reported	Senseval-3	60.9%
Exemplar based (K-NN)	68.6%	Not Reported	WSJ6 containing 191 words	63.7%
Decision Lists	96%	Not Applicable	Test conducted on set of 12 ambiguous words	63.9%
Support Vector Machine	72.4%	72.4%	Senseval-3	55.2%
Perceptron Trained HMM	67.6%	73.74%	Senseval-3	60.9%

2.2.2 Semi-Supervised Algorithms

This technique requires significantly fewer amounts of tagged data. This learning algorithm has the characteristics of learning from the annotated data, with minimal human supervision [27].

- It is capable of conducting automatic bootstrapping of a corpus, by taking human annotated examples into consideration [30].
- It makes use of monosemous relatives / dictionary definitions for the automated construction of the sense tagged data.
- It also relies on the Web-users for the purpose of corpus annotation

These algorithms take bootstrapping approaches into consideration. The pre-requisites for the bootstrapping approach includes –

- Labeled data
- Unlabelled data
- One or more than one basic classifiers

A new classifier which is far more improved than the basic classifiers is obtained as a result of the application of the above mechanism [33].

This makes the use of Yarowsky's supervised Bootstrapping algorithm that takes Decision Lists into the consideration. The Yarowsky's approach is dependent on a decision list and two heuristics.

- One sense per collocation
    - The words surrounding the ambiguous word provide sufficient hints about its sense.
- One sense per discourse:
    - The sense of a target word is highly consistent within a single document

The learning algorithm based on this approach makes the use of decision lists to categorize the different instances of an ambiguous word. The sense getting highest rank gives maximum match to the context.

The process can be described in 3 words: initialization, progress and convergence. In this very first 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. Comparison of these techniques is mentioned in table 2.4 below.

Table 2.4: Comparison of Semi-Supervised approaches [28]

Approach	Average Precision	Corpus	Average Baseline Accuracy
Supervised Decision Lists	96.1%	Tested on set of 12 ambiguous words	63.9%
Unsupervised Decision Lists	96.1%	Tested on 12 ambiguous words	63.9%

2.2.3 Unsupervised Algorithms

Through the means of unsupervised algorithms one can identify patterns in a large sample of data that too without the assistance of any knowledge source or manually labeled examples. The patterns recognized divides data into the clusters, where each member of a cluster exhibits more similarity with the other members belonging to its own cluster rather than the members belonging to any other clusters. If manual labels are removed from supervised cluster and data, then it becomes difficult to unearth the classes of similar nature. In this way, Supervised Classification recognizes the features that activate a sense tag and the unsupervised Clustering detects sameness between the contexts.

A sense tagged text is used for the evaluation purpose, but cannot be used for the feature selection or clustering. To attain maximum amount of accuracy, sense tags are assumed to be the clusters which are mapped against the available senses to obtain resultant output.

2.2.3.1 Major Algorithms

The major algorithms covered under this category are-

2.2.3.1.1 Lin's Algorithm

This algorithm provides general purpose broad coverage approach. It is capable of working for the words which does not even appear in the corpus [34].Just as supervised approaches,it extracts evidence from the corpus and like knowledge based approaches this algorithm doesn't require tagged corpus

2.2.3.1.2 Hyperlex

Under this technique, the senses from the corpus are extracted. 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').

2.2.3.1.3 Yarowsky's Algorithm

This is another broad coverage classifier which can be employed for the words which do not appear in the corpus but it is not tested on 'all word corpuses'. The key idea behind the working of this algorithm is that, instead of making the use of dictionary defined senses, it extracts the senses from the corpus itself [33].

2.2.3.1.4 WSD using Parallel Corpora

This algorithm facilitates in fetching the difference 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 are required to be trained [35].

2.2.3.2 Comparison of Unsupervised Algorithms

Comparative analysis is mentioned in the table 2.5:

Table 2.5: Comparison of Unsupervised Approaches

Approach	Precision	Average Recall	Corpus	Average Accuracy
Lin's Algorithm	68.5%	--	WSJ, SemCor	64.2%
Hyperlex	97%	82%	Tested on set of 10 ambiguous words of French Language	73%
WSD using Roget's Thesaurus	92%	--	Experiment performed on 12 ambiguous words of English language.	--

2.3 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 [36]:

2.3.1 Methodologies under Hybrid Approaches

Brief description of some of the major approaches is mentioned below:

2.3.1.1 An Iterative Approach to WSD

The main points regarding this approach are as follows -

- Uses semantic relations (synonymy and hypernymy) form WordNet.
- Extracts collocational and contextual information from WordNet (gloss) and from 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.

2.3.1.2 Sense Learner

Tagged data is used to develop a semantic language model for the words found in the training corpus. WordNet is used to obtain the semantic generalizations for those words which are not present in the corpus.

- For each word of a sentence, a separate training set is developed, for the purpose of assignment of POS.
- Each training example is represented as a feature vector.
- 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 word predicted is same as the observed word, then the corresponding sense predicted, is selected as the correct sense.

2.3.1.3 Structural Semantic Interconnections (SSI)

- It is an iterative approach [13].
- It makes the use of following relations -
    - Hypernymy (banyan is kind of tree) denoted by (kind-of)
    - Hyponymy (the inverse of hypernymy) denoted by (has-kind)
    - Meronymy (room has-part wall) denoted by (has-part)
    - Holonymy (the inverse of meronymy) denoted by (part-of)
    - Pertainymy (dental pertains-to tooth) denoted by (pert)
    - Similarity (beautiful similar-to pretty) denoted by (sim)
    - Gloss denoted by (gloss)
    - Context denoted by (context)
    - Domain denoted by (dl)
- Monosemic words serve as the seed set for disambiguation.

2.3.2 Comparison of Hybrid Approaches

Comparison of various hybrid approaches is shown in the table 2.6 :

Table 2.6: Comparison of Hybrid Approaches

Approach	Precision	Average Recall	Corpus	Baseline Accuracy
Iterative Approach	92.2%	55%	SemCor	Not Reported
Sense Learner	64.6%	64.6%	SenseEval-3	60.9%
SSI	68.5%	68.4%	SenseEval-3	Not Reported

2.4 Chapter Summary

This chapter gives a brief description of various approaches meant to overcome the issue of Word Sense Disambiguation. It is found that the techniques discussed throughout the chapter fall under these three different approaches, i.e., The Knowledge Based Approaches, Machine Learning Approaches, and The Hybrid Approaches. So, it can be said that a large number of advances have been conducted in this domain, but still there exist some gaps which leaves a scope for further advancement in this field. The gap analysis and the problem statement are described in the next chapter.

Chapter3
Problem Statement

3.1 Gap Analysis

Based on the literature review of various Word Sense Disambiguation Techniques following gaps have been identified:

- No unified technique has been developed to map ambiguous words to local languages [3, 4, 14].
- The objective to acquire the results in terms of correct sense of an ambiguous word to the best possible level of accuracy has not been accomplished yet [2,11,13,16].
- There is a need to resolve the issue of determination of correct Part of Speech of an ambiguous word for the assignment of correct semantic category [28, 30, 34, 35].

3.2 Problem Statement

In the field of computational linguistics, some outcomes have already been derived; however, the target to achieve the results to the best possible level of accuracy has not been accomplished yet. In applications, such as machine translation or information retrieval, it is mandatory to be able to differentiate between the different senses of a word, in order to obtain the correct translation of a word to another language. So, it can be said that it is necessary to be familiar with the different senses of the words and then detect the best conversion which is equivalent in the target language. The phenomenon of the language divergence problem holds key concern in the area of Information Retrieval and Machine Translation and this problem arises from the only fact that the languages make different lexical and syntactic choices from expressing an idea.

Chapter 4

Objectives and Methodology

4.1 Objectives

The accurate description of specific actions taken in order to provide solution to the problem statement described in previous chapter is given in the steps that follow:

- To study the various approaches of Word Sense Disambiguation with the intent to get the clear insight of the researches conducted so far in this domain.
- To resolve the issue of determination of the correct Part of Speech of the target word present in the context sentence.
- Implementation of the sense evaluation algorithm to determine the sense which exhibits the best match with the context sentence.
- Translation of ambiguous words to the local languages named Punjabi and Hindi.

4.2 Methodology

A framework is proposed which is aimed at fetching information from a knowledge source named WordNet so as to disambiguate a target word in a sentence.

Following steps have been followed to achieve the objectives mentioned above:

1. A sentence or a paragraph inputted undergoes the process of Tokenization, Part of Speech Tagging, Lemmatization, Chunking and Parsing.

    1.1 The text inputted by the user is ruptured into the set of tokens.

    1.2 The grammatical category is assigned to each word with the help of POS Tagger.

    1.3 Inflectional forms are reduced to base forms with the help of process of lemmatization.

    1.4 Syntactically correlated words are grouped together in the process of chunking.

    1.5 Syntactic layout of a string is studied under the process of parsing.

2. After this processing, the target word i.e. the ambiguous word is figured out from the sentence. This is done with the help of the steps as follows:

2.1 The output obtained as a result of pre-processing of sentence is stored as an array of the words, which is compared with an array comprising of the most ambiguous words in English language

2.2 As a consequence of this comparison, an ambiguous word i.e. the target word is discovered from the sentence, which will be passed further for the disambiguation process.

3. The target word and the context sentence or paragraph is passed as an input to the Decision Tree Classifier, which will serve the purpose of predicting the Part-of-Speech of a target word.

   3.1 Decision Tree Classifier is employed as a POS Classifier, which examines the internal look-up of sentence to determine the POS of a target word.

   3.2 This classifier decides the Part-of-Speech of a target word, with the help of the information obtained from the suffixes of a word (a function named FreqDist( ) found in the NLTK python module is used to fetch the set of various suffixes).

   3.3 pos_features( ) function helps in determining the suffix (if any) of a target word, and the classifier decides the POS only on the basis of the information obtained from this function, it makes a simple flowchart that selects labels for input values. This flowchart consists of decision nodes, which check feature values, and leaf nodes, which assign labels.

4. The target word, the Part-of-Speech of the target word and the context sentence are passed as an input to the Gloss Overlap algorithm named Lesk Algorithm.

   4.1 The Lesk algorithm with the help of a knowledge source named WordNet, obtains all the senses of a target word which come under the POS predicted by Decision Tree Classifier.

   4.2 The sense which exhibits maximum overlap with the words in the context bag will be the winner sense and is assumed to be the best suited sense as per the context.

5. The other senses and the resultant sense obtained as an output was further judged by the computation of its similarity score (i.e. Wu-Palmer similarity score and Jiang-Conrath similarity score) with the words in the context bag. Winner sense is supposed to acquire maximum similarity score, this further confirmed to the fact that the winner sense provided by the Lesk Algorithm is the best suited sense to the context.

6. The target word after disambiguation is translated to the local languages named Hindi and Punjabi.
   6.1 The data containing the translation is stored in an excel sheet and python module named ‘xlrd’ will serve the purpose of providing access and retrieval from this exc

Chapter- 5

Implementation

5.1 Basic Architecture Description

The issue of lexical disambiguation existing in a sentence is resolved here with the help of the Lesk Algorithm, with the modification that, the Part of Speech (POS) of the ambiguous word is predicted with the help of Decision Tree Classifier, which helps in resolving the issue of accuracy, to determine the correct POS to a great extent, and this even aided the Lesk Algorithm to limit its effort to just one Part-of-Speech of the ambiguous word only. Experimental results showed that the accuracy to determine the sense of a word was improved. The resultant sense obtained as an output was further judged by the computation of its similarity score (i.e. Wu-Palmer similarity score and Jiang-Conrath similarity score) with the words in the context. The modified Lesk Algorithm further facilitated in getting the correct translation of the ambiguous words to the languages named Punjabi and Hindi. Figure 5.1 gives an insight about the basic architecture of the implementation.

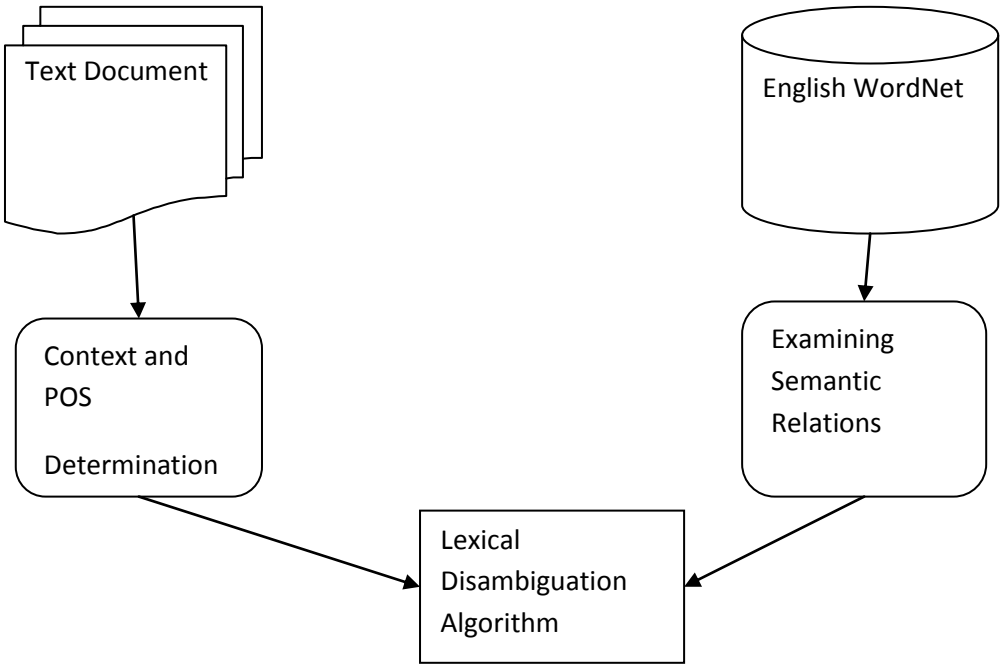


Figure 5.1: Architecture of the tool meant to resolve issue of Lexical Disambiguation

5.2 Description of Task

The task is divided into the following sub-sections mentioned below:

5.2.1 List of Python Modules

The implementation incorporating this concept is done with the help of a python module named 'Natural Language Toolkit (NLTK)'. It is a leading platform which provides python programs a privilege to work with the human language data.

The snapshot in the figure 5.2 entails all the modules and functions required for the accomplishment of the task. NLTK provides major contribution in performing tasks such as:

- Importing the reference inventory i.e. WordNet here ,
- Sentence pre-processing steps like Tokenization ,Stemming etc,
- Establishing the Decision Tree Classifier.

Other functions such as creation of the Graphical User Interface, calculations, access to the Excel sheets for the translation purpose are supported with the modules such as 'Tkinter','re' ,'xlrd' etc.

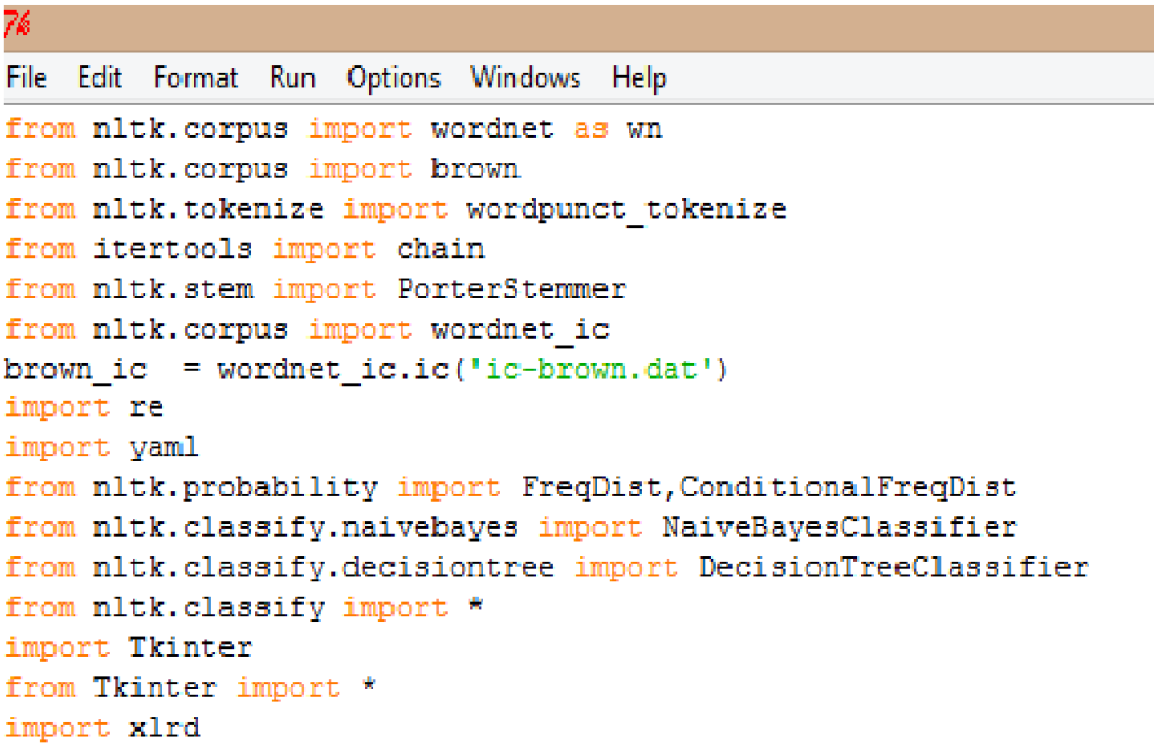


Figure 5.2: Modules required for the accomplishment of the task

5.2.2 Sentence Pre-Processing Techniques

The very first step begins with the processing of the sentence entered by the user. The input text i.e. the sentence entered by the user is subjected to the following steps [34]:

- Tokenization i.e. the text is split into the set of the tokens.
- Tagging of the Part-of-Speech i.e. the grammatical category is assigned to each word.
- Lemmatization – the inflectional forms are reduced to the base form.
- Chunking- syntactically correlated parts are clubbed into groups.
- Parsing - syntactic layout of a sentence is analyzed under this process.

Flow of processing is presented in figure 5.3. As a consequence of this text processing activity, the sentence can be shown as a graph or a tree, which provides the insight of the relations existing between the words [38]. The main task is to assign the most appropriate sense to the word from the reference inventory (i.e. WordNet) that best reflects the context of the sentence.

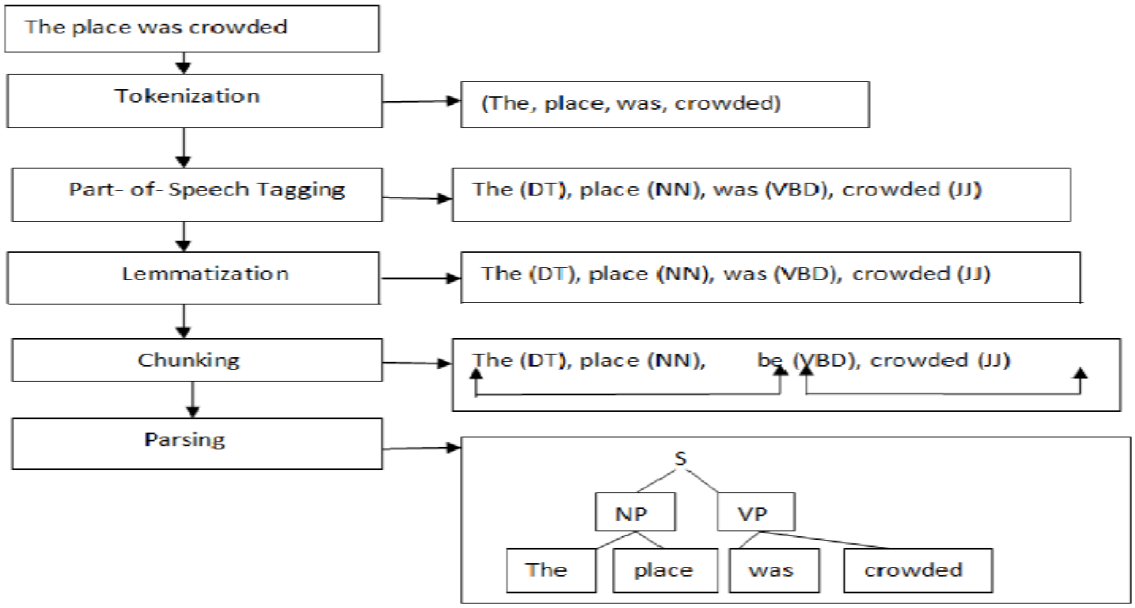


Figure 5.3: Sentence Pre-processing steps

5.2.3 Process of determination of POS of the Target Word

After the sentence is processed, the ambiguous word present in the sentence is figured out, which is then passed as an input, along with the context sentence to the Decision Tree Classifier to predict its Part of Speech [27]. The classifier is trained in such a manner that the Part of Speech of the target word is predicted by examining the internal look up of a word. In other words, it works on a principle that suffixes

provide enough information about a word. The snapshot mentioned in the figure 5.4 lists all the most common suffixes a given target word can possess. This list helps the function named pos_features( ) in the implementation, to learn, which suffix a word is attached with.

```
>>> common_suffixes = suffix_fdist.keys()[:100]
>>>  print common_suffixes
['e', ',', '.', 's', 'd', 't', 'he', 'n', 'a', 'of', 'the',
 'y', 'r', 'to', 'in', 'f', 'o', 'ed', 'nd', 'is', 'on', 'l',
 'g', 'and', 'ng', 'er', 'as', 'ing', 'h', 'at', 'es', 'or',
 're', 'it', '``', 'an', "''", 'm', ';', 'i', 'ly', 'ion', ...]
```

Figure 5.4: Suffix set

The classifier makes decisions exclusively on the basis of the suffixes possessed by the target word. This type of feature extraction mechanism (which is conducted with the help of a function named pos_features( ) in the implementation) is used to train the decision tree classifier here. The snapshot of a code snippet in the figure 5.5 describes the way through which the features extracted by the function named pos_features( ), can be used to the train the Decision Tree Classifier.

```
suffix_fdist = FreqDist()
for word in brown.words():
    word = word.lower()
    suffix_fdist.inc(word[-1:])
    suffix_fdist.inc(word[-2:])
    suffix_fdist.inc(word[-3:])

common_suffixes = suffix_fdist.keys()[:100]
print common_suffixes
def pos_features(word):
    features = {}
    for suffix in common_suffixes:
        features['endswith(%s)' % suffix]=word.lower().endswith(suffix)
    return features

tagged_words = brown.tagged_words(categories='news')
featuresets = [(pos_features(n), g) for (n,g) in tagged_words]
size = int(len(featuresets) * 0.1)
train_set,test_set = featuresets[size:] , featuresets[:size]
global classifier
classifier = DecisionTreeClassifier.train(train_set)
```

Figure 5.5: Code Snippet representing Decision Tree Classifier and Feature extraction function

A decision tree is actually a simple flowchart which picks labels for the inputted values. This flowchart comprises of decision nodes, which checks feature values, and

the leaf nodes, which assign labels. The value is passed at the flowchart's initial decision node, which is also known as the root node. This node contains a condition that checks one of the input value's features (these features are provided by the feature extraction function i.e. pos_features( ) here), and consequently a branch is selected on the basis of that feature's value. Following the branch describing the feature value, a new decision node is obtained, which further comes up with a new condition on inputted feature values. The snapshot in the figure 5.6 below describes the decision tree printed in the form of pseudocode, the feature value inputted undergoes the following set of conditions for the determination of correct POS.

```
>>> print classifier.pseudocode(depth=4)
if endswith(the) == False:
  if endswith(,) == False:
    if endswith(s) == False:
      if endswith(.) == False: return '.'
      if endswith(.) == True: return '.'
    if endswith(s) == True:
      if endswith(is) == False: return 'PP$'
      if endswith(is) == True: return 'BEZ'
  if endswith(,) == True: return ','
if endswith(the) == True: return 'AT'
```

Figure 5.6: Decision tree as a Pseudocode

5.2.4 Sense Evaluation Algorithm -The Lesk Algorithm

The ambiguous word derived and its predicted Part Of Speech are passed along with the context sentence as an input to the Lesk algorithm which takes the help of Knowledge based approaches (i.e. WordNet) to resolve the issue of WSD. This algorithm incorporates the calculation of overlap of the ambiguous word's gloss with the context sentence's gloss. So, this algorithm is also known as Gloss Overlap algorithm.

Let's assume S1 denotes the gloss of the ambiguous word (can be called as Sense Bag) and S2 denotes the gloss of Context sentence (can be called as Context Bag) [15].

$$\text{Similarity Score}_{Lesk}(S_1, S2) = |\text{gloss}(S_1) \cap \text{gloss}(S_2)| \quad (5.1)$$

where gloss (S_i) refers to the set of words in the definition of the different senses of S_1 . Gloss of best sense will give maximum overlap to the gloss of the context i.e.:

$$Score_{LeskVar}(S) = |context(w) \cap gloss(S)| \quad (5.2)$$

where context (w) denotes the bag which contains all the words of the content in the context window and gloss(S) denotes all the synsets of the sense i.e. the sense bag[15]

The similarity score of each sense of the ambiguous word can be computed as [25]:

$$Score_{ExtLesk}(S) = \sum_{S'':S \to S' \text{ or } S \equiv S'} |context(w) \cap gloss(S')| \quad (5.3)$$

where context (w) denotes the words residing in the context bag around the ambiguous word 'w' and gloss(S') refers to the sense bag which contains the definition of various senses of an ambiguous word. The more the score is, the more that particular sense will match to the context of the sentence. Semantic similarity measure can be defined as:

$$Score: Senses_A \times Senses_A \to [0, 1] \quad (5.4)$$

where $Senses_A$ refers to all the set of senses listed within the reference lexicon. The sense which exhibits the maximum score will be considered as the best sense and hence will suit well to the context of the sentence.

5.2.5 Semantic Similarity Metrics- Wu-Palmer and Jiang-Conrath Measures

The semantic similarity relatedness measures further confirms the accuracy of the algorithm by the comparing the value of the similarity score of the different senses belonging to the same Part-of-Speech of an ambiguous word with the context. If the winner sense outputted by the Lesk Algorithm scores maximum out of all the senses then the output exhibited is true and accurate. The similarity metrics employed for this purpose are: Jiang- Conrath Similarity measure and Wu-Palmer semantic relatedness measure[37]. Brief description of these measures is as follows:

5.2.5.1 Wu and Palmer Similarity measure

The Wu and Palmer (shortly speaking, wup similarity computation) measures semantic similarity by taking into consideration the depths of the two synsets (say, C_1, C2 here) and the distance of their LCS from the root entity in the WordNet taxonomical hierarchy.

$$Sim(C1, C2) = \frac{2*depth(LCS)}{(depth(C1)+depth(C2))} \quad (5.5)$$

The expression exhibits the value in the range (0, 1] [38].

5.2.5.2 Jiang-Conrath Similarity measure

Jiang-Conrath (shortly speaking, jcn) semantic relatedness measure uses child node to calculate semantic distance when there exists a common parent between the two synsets (say, C_1, C2). The value of semantic similarity in this case is inversely proportional to the value of semantic path length [39].

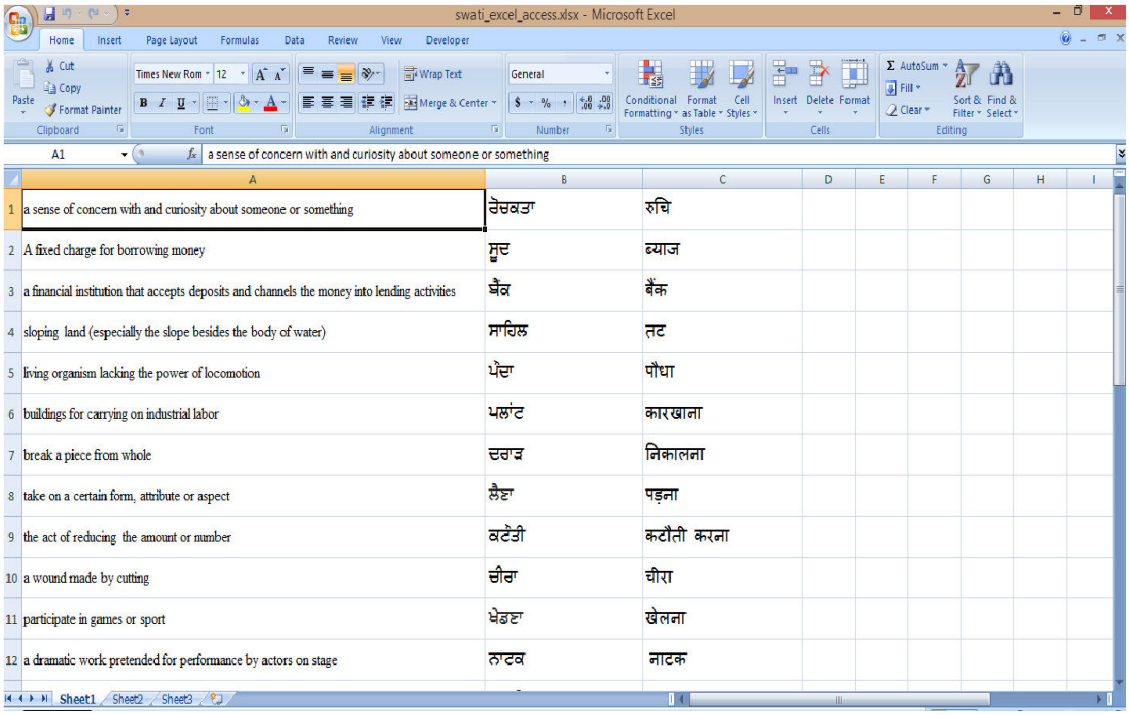
$$Sim(C1, C2) = \frac{1}{IC(C1)+IC(C2)-2*IC(lcs(C1,C2))} \quad (5.6)$$

5.2.6 Mapping to Local Languages

Now that the winner sense is derived accurately, it becomes easy to determine the correct translation of the target words to Local Languages named Punjabi and Hindi. This is accomplished with the help of the mechanism mentioned below:

- The various definitions of an ambiguous word are listed in a column in an excel sheet and their translation is mentioned in the cells neighboring them.
- The gloss of the winner sense obtained as a result of the application of the above algorithm is compared with the elements residing in different cells of this column.
- On detection of the match, translations are fetched from the neighboring cells and printed on the Graphical User Interface.

The snapshot in figure 5.7 shows that the definition of various synsets is mentioned in a column with their translation to Punjabi and Hindi in the neighboring cells. Python module named ‘xlrd’ provides access to the excel sheet and hence facilitates in fetching the correct translation.



swati_excel_access.xlsx - Microsoft Excel

	A	B	C
1	a sense of concern with and curiosity about someone or something	ਰੋਚਕਤਾ	रुचि
2	A fixed charge for borrowing money	ਸੂਦ	ब्याज
3	a financial institution that accepts deposits and channels the money into lending activities	ਬੈਂਕ	बैंक
4	sloping land (especially the slope besides the body of water)	ਸਾਹਿਲ	तट
5	living organism lacking the power of locomotion	ਪੌਦਾ	पौधा
6	buildings for carrying on industrial labor	ਪਲਾਂਟ	कारखाना
7	break a piece from whole	ਦਰਾੜ	निकालना
8	take on a certain form, attribute or aspect	ਲੈਣਾ	पड़ना
9	the act of reducing the amount or number	ਕਟੌਤੀ	कटौती करना
10	a wound made by cutting	ਚੀਰਾ	चीरा
11	participate in games or sport	ਖੇਡਣਾ	खेलना
12	a dramatic work pretended for performance by actors on stage	ਨਾਟਕ	नाटक

Figure 5.7: Excel sheet showing the definition of various synsets along with their translation to Hindi and Punjabi

5.3 Chapter Summary

The steps mentioned above gives an insight of the inter-modular interaction taking place in order to determine the output.

The sentence or paragraph inputted undergoes the process of Tokenization, Part of Speech tagging, Lemmatization, chunking and parsing. After this processing, the ambiguous word is figured out from the sentence which is further passed as an input along with the context sentence to the Decision Tree Classifier to determine its Part of Speech accurately. The ambiguous word and its POS obtained as the result of the following implementation is passed as an input along with the context sentence to the Gloss Overlap Algorithm. The output is derived in the form of the sense of an ambiguous word which suits well to the context. This 'winner sense' obtained, is supposed to exhibit maximum similarity score (i.e., wup and jcn similarity score) amongst all the senses to prove that the Modified Lesk Algorithm is capable of exhibiting results with accuracy and this finally leads to correct translation of the target words to the languages named Punjabi and Hindi. The results are discussed in next chapter.

Chapter 6

Results and Discussion

The snapshot in figure 5.1 shows the best sense of an ambiguous word outputted with the help of the mechanism described above and its translation to the local languages named Hindi and Punjabi.

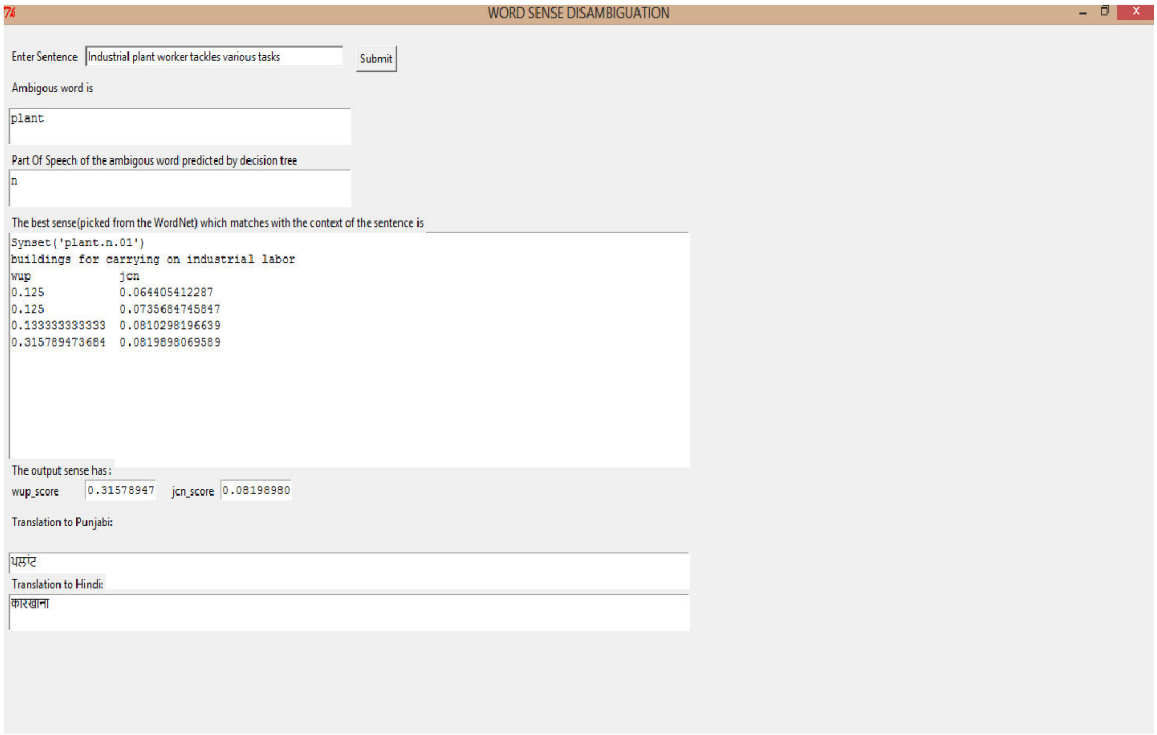


Figure 6.1: Interface showing winner sense with translations and similarity scores

The sentence or paragraph inputted undergoes the process of Tokenization, Part of Speech tagging, Lemmatization, chunking and parsing [34]. After this processing, the ambiguous word is figured out from the sentence which is further passed as an input along with the context sentence to the Decision Tree Classifier to determine its Part of Speech accurately depending upon the context of the sentence. The ambiguous word and its POS obtained as the result of the following implementation is passed as an input along with the context sentence to the Gloss Overlap Algorithm, where the definition sets of different synsets under a common POS of the ambiguous word (i.e. the one residing in the sense bag) are compared with the definition set of the synsets of the words lying in context bag. The sense which gives the maximum overlap to the context bag will be the best suited sense.

Table 5.1 shows the WUP similarity and the JCN similarity score of various senses (coming under POS ‘n’ here) of word plant with the context. It is observed that the synonym set i.e. synset (‘plant.n.01’) shows maximum match to the sense of the context. This comparison also illustrates the fact that the output sense given by the Lesk Algorithm is correct.

Table 6.1: Comparison of similarity scores of various sense of an ambiguous word ‘plant’ with the context

Senses / Scores	sense 1	sense 2	sense 3	sense 4
WUP Similarity Score of senses with the context	0.315789	0.125	0.1333	0.125
JCN Similarity Score of senses with the context	0.08198	0.064405	0.081029	0.073568

The incorporation of Decision Tree Classifier reduced the efforts of the gloss overlap algorithm by limiting its scope to just one Part-of-Speech of the target word, so the senses which are covered under that Part-of-Speech will be called for comparison with the context and the rest of the senses which come under other Part-of-Speech of the target word will not be counted. This confinement of scope increases the accuracy of the Lesk Algorithm to an appreciable extent and the chances to get correct output gets maximizes accordingly. The scores obtained finally confirm the fact that the output results produced by the modified Lesk Algorithm are valid. In this way, the winner sense which best complements the context of the sentence is obtained, similarity score further strengthen the output giving the clear insight that the sense obtained as a result of implementation is a solution to the lexical disambiguation issue lying in the sentence. Table 6.2 lists some of the sentences (with the ambiguous words

highlighted) taken for the experiment with the results in the form of the winner sense and translations mentioned.

Table 6.2: Experiment showing resultant sense and its translation to Punjabi and Hindi

SENTENCE (With ambiguous word highlighted)	WINNER SENSE (With definition, WUP similarity score and JCN similarity score)	Translation to Punjabi	Translation to Hindi
Interest in a subject	Synset('interest.n.01') a sense of concern with curiosity or concern about something wup:0.2857 jcn:0.08362	ਰੋਚਕਤਾ	रुचि
Interest in bank	Synset ('interest.n.04') A fixed charge for borrowing money wup:0.125 jcn:0.0552213	ਸੂਦ	ब्याज
She goes to **bank** deposit her money	Synset('bank.n.02') A financial institution that accepts, deposits and channels the money into lending activities wup : 0.2666 jcn : 0.0603479	ਬੈਂਕ	बैंक

The river **bank** was full of dead fishes	synset('bank.n.01') sloping land (especially the slope besides the body of water) wup:0.42105 jcn:0.6956	ਸਾਹਿਲ	तट
Green **plant**	synset('plant.n.02') living organism lacking the power of locomotion wup:0.3875 jcn: 0.07295	ਪੌਦਾ	पौधा
The workers at the **plant** were overworked	synset('plant.n.01') buildings for carrying on industrial labor wup: 0.3529411 jcn: 0.06590352	ਪਲਾਂਟ	कारखाना
Break a piece from the whole	synset('break.v.43') break a piece from whole wup:0.3942 jcn:0.0482	ਦਰਾੜ	निकालना
Let's **take** a 10 minute break.	synset('assume.v.03') take on a certain form, attribute or aspect wup:0.25 jcn: 0.063805	ਲੈਣਾ	पड़ना
Tom **cut** classes again	synset('cut.v.02') the act of reducing the amount or number	ਕਟੌਤੀ	कटौती करना

	wup:0.37401 jcn:0.06945		
I have a **cut** on my right elbow.	synset('cut.n.05') a wound made by cutting wup: 0.1000 jcn: 0.047898	ਚੀਰਾ	चीरा
She is **playing** in the courtyard	synset('play.v.01') participate in games or sport wup:0.1333 jcn:0.0563801	ਖੇਡਣਾ	खेलना
She participated in a **play**	synset('play.n.01') a dramatic work pretended for performance by actors on stage wup: 1.0 jcn:0.153349	ਨਾਟਕ	नाटक
I know you can **make** it	synset('make.v.05') give rise to; cause to happen or occur wup:0.3942 jcn:0.05374	ਬਣਾਉਣਾ	गठन
Feather is **light** (v)	synset('light.a.01') of comparatively little physical weight or density wup: 0.5 jcn: 0.0617	ਹਲਕਾ	भार रहित
A shrewd **light** entered his eyes	synset('light.n.01') (physics) electromagnetic radiation that can produce a visual	ਪ੍ਰਕਾਸ਼	रौशनीई

	sensation wup : 0.57142857 jcn : 0.08510379		
The Home Secretary **set** in motion a review of the law	synset('set.v.04') establish as highest level or best performance wup: 0.5283 jcn:0.03717	ਤੈਅ	निर्धारित
Set of cards	synset('set.n.01') a group of things of the same kind that belong together and are so used. wup : 0.307692 jcn : 0.860804	ਸੈੱਟ	समुच्चय
He caught **hold** of her arm	synset('clasp.n.02') the act of grasping wup : 0.1111 jcn: 0.056364	ਫੜਨਾ	मजबूती से पकड़ना
The police were **holding** him on a murder charge	synset('hold.v.02') keep in a certain state, position or activity wup : 0.4761 jcn : 0.0782610	ਕਾਬੂ	हिरासत में रखना
Clear and precise directions	synset('clear.n.01') the state of being free of suspicion wup : 0.571428 jcn : 0.10788	ਸਪੱਸ਼ਟ	बिना शंका के

He had time to get **clear** away	synset(‘clear.v.08’) go away or disappear wup: 0.585901 jcn:0.03741	ਫ਼ੁੱਟਣਾ	साफ़ हो जाना
Karen **cleared** the dirty plates	synset(‘clear.v.02’) make a way or path by removing objects wup: 0.4731 jcn:0.03563	ਨਿਖਾਰਨਾ	साफ करना
he **took** an envelope from his inside pocket	synset(‘take.v.04’) get into one’s hands, take physically wup: 0.5832 jcn:0.02721	ਲੈਣਾ	लेना
the **take** from commodity taxation	synset(‘take .n.01’) the income or profit arising from such transactions as the sale of land or other property wup: 0.361 jcn:0.146106	ਛਕਣਾ	ग्रहण करना
he **passed** through towns and villages	synset(‘pass.v.01’) go across or through wup: 0.29757 jcn:0.03684	ਲੰਘਣਾ	पास से निकल जाना
she **passed** her driving test	synset(‘pass.n.01’) (baseball) an advance to the first base by a batter who receives four balls wup: 0.31578	ਪਾਸ	उत्तीर्ण

	jcn:0.06811267		
the **head** waiter	synset('chief.n.01') a person who is in charge wup: 0.6 jcn:0.0759	ਮੁੱਖ	प्रमुख
The St George's Day procession was **headed** by the mayor	synset('head.v.04') be the first or leading member of (a group) and excel wup:0.47643 jcn:0.05974	ਪ੍ਰਧਾਨਗੀ	नेतृत्व करना
An article **headed** 'The Protection of Human Life	synset('head.n.08') the top of something wup:0.428571 jcn:0.07338	ਸਿਰਲੇਖ਼	अग्रभाग
She heard Terry **calling** her	synset('call.v.10') utter a characteristic note or cry wup:0.43071 jcn:0.0759	ਬੁਲਾਰਾ	आवाहन

6.1 Chapter Summary

The application of the above mechanism to the example sentences lead to the determination of an ambiguous word present in the sentence, along with this, it also helped in the determination of the Part -of- Speech and the best sense exhibited by the ambiguous word. The Wu-Palmer and the Jiang – Conrath similarity scores further revealed that the winner sense scored maximum out of all the senses which confirms to the fact that the winner sense obtained shows the best match to the context. This implementation helped in obtaining the correct translation of the target word to the local languages named Punjabi and Hindi.

Chapter 7

Conclusions and Future Scope

In this thesis, a tool is developed for English WordNet that can be used for Word Sense Disambiguation (WSD), as part of Natural Language Processing (NLP) tasks, especially for the translation of the ambiguous words to local languages named Hindi and Punjabi.

7.1 Conclusion

WSD is a tedious task to deal with, as to resolve this issue one need to overcome the complexities of language and it is a complicated affair to recognize a semantic layout from the unstructured sources of the text and still the researches are continued in this field so as to resolve the issue at the best possible level of accuracy. The issue of Word Sense Disambiguation is resolved here with the help of a knowledge source named WordNet and the modified Lesk Algorithm. The ambiguous word figured out from the sentence and the Part of Speech predicted with the help of Decision Tree Classifier is passed as an input along with the context sentence to the Gloss Overlap Algorithm. The output is yielded in the form of a 'sense' which gives the best match with the context of the sentence in which it is mentioned. The results are further confirmed by comparing the similarity scores of different senses with the context and it is found that the resultant output 'sense' gets maximum of the similarity score. This facilitated in exhibiting the correct translation of the ambiguous words to Punjabi and Hindi. In future, the work can be extended to improve and advance the accuracy of Machine Translation, speech recognition, information retrieval and other fields with lesser ambiguity in results.

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 can lead to further advancement of the project.
- 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.
- The accuracy of the Lesk's algorithm can be checked on other languages.

References

[1] Navgili, R., "Word Sense disambiguation: A survey," *ACM Computation Survey,* Article 10, 2009.

[2] Mihalcea, R., "Knowledge-based methods for WSD. In Word Sense Disambiguation: Algorithms and Applications", *Agirre, E., and Edmonds (Eds) Springer,* Vol.33 ,New York, USA, 2006, pp. 107–131

[3] Carpuat, M., and Wu, D., "Improving statistical machine translation using word sense disambiguation," *In Proceedings of Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning,* 2007, pp. 61–72

[4] Vickrey, D., Biewald, L., Teyssier, M., and Koller, D., "Word-sense disambiguation for machine translation," *In Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing,* Association for Computational Linguistics, Morristown ,NJ, USA, 2009, pp. 771–778

[5] Resnik, P., "Word Sense Disambiguation in Natural Language Processing Applications," *Agirre, E., and Edmonds, P. Eds., Springer,* chap.11, 2006, pp. 299– 337

[6] Navgili, R., "A structural approach to the automatic adjudication of word sense disagreements" *Journal Natural Language of Engineering,* Vol. 14, 2008, pp. 547–573

[7] Fellbaum, C., "WordNet: an electronic lexical database." Bradford Books, Bradford. , 2008.

[8] Fellbaum, C., "WordNet - A lexical DataBase for English", Available at http://wordnet.princeton.edu/wordnet/man2.1/wnstats.7WN.html, [As Accessed on May 5, 2014].

[9] Alan, D. C., "*Lexical semantics*." Cambridge University Press, 2000.

[10] Banarjee, S., and Pedersen, T., "Extended gloss overlaps as a measure of semantic relatedness," *In Proceedings of 18th International Joint Conference on Artificial Intelligence*, (IJCAI), Acapulco, Mexico, 2003, pp. 805-810

[11] Mihalcea, R., and Moldovan. , "An automatic method for generating sense tagged corpora," *In Proceedings of the 16th National Conference on Artificial Intelligence*, AAAI, Orlando, 1999, pp. 461–466

[12] Lee, Y. K., and Ng, H. T., "An empirical evaluation of knowledge sources and learning algorithms for Word Sense Disambiguation," *In EMNLP '02: In Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, Morristown, NJ, USA. Association for Computational Linguistics, 2002, pp. 41–48

[13] Navgili, R., and Velardi, P., "Structural Semantic Interconnections: A knowledge based approach to Word Sense Disambiguation," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 27, No. 7, 2005, pp.1075-1085

[14] Kilgarriff, A., Reddy, S., and Pomiklek, J., "A corpus factory for many languages," *In Proceedings of the Seventh conference on International Language Resources and Evaluation* ,(LREC'10), Valletta, Malta, 2010 ,pp. 143-148

[15] Lesk, M., "Automatic Word Sense Disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone," *In Proceedings of the 5th annual international conference on Systems documentation*, New York, 1986, pp. 24-29

[16] Mitesh, K., "Words Sense Disambiguation", Available at: http://www.cse.iitb.ac.in/~nlp-ai/WSD.ppt ,2007 [As accessed on April 2014].

[17] Ide, N., and Wilks, Y., "Making Sense About Sense. In Word Sense Disambiguation: Algorithms And Applications," *Springer*, Vol. 15, chap. 18, 2006, pp. 322-354

[18] Wilks, Y., Fass, D., Guo, C., McDonald J., Plate, Slator, B., "Providing machine tractable dictionary tools," *Machine Translation,* Vol. 5, No. 2, 1990, pp. 99–154

[19] Agirre, E., and Mart´ınez, D., "Knowledge sources for word sense disambiguation" *In Proceedings of 4th International Conference on Artificial Intelligence*, Springer, Zelezna Ruda, Czech Republic, 2001, pp.287-295

[20] Cowie, J., Guthrie, J., and Guthrie, L., "Lexical disambiguation using simulated annealing." *In Proceedings of the 14th International Conference on Computational Linguistics*, Nantes, France, 1991 pp. 359–365

[21] Veronis, J., and Ide, N., "Word Sense Disambiguation with very large neural networks extracted from Machine Readable Dictionaries," *In Proceedings of the 13th International Conference on Computational Linguistics,* Helsinki, 1991, pp. 389–394

[22] Kozima, H., and Furugori, T. , "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

[23] Nitwa, Y., and Nitta, Y., "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

[24] Sussna, M., "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

[25] Banerjee, S., and Pedersen, T., "An adapted Lesk Algorithm for Word Sense Disambiguation WordNet," *In Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing*, Springer-Verlag, London, UK, 2002, pp.136-145

[26] Patwardhan, S., Banerjee, S., and Pedersen, T., "Using measures of semantic relatedness for word sense disambiguation," *In Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*, Mexico City, Mexico, 2003, pp. 241–257

[27] Florian, R., Cucerzan, S., Schafer, C., and Yarowsky, D., "Combining Classifier for Word Sense Disambiguation," *Journal Natural Language and Engineering* ,Vol. 8, No. 4, 2002, pp. 1-14

[28] Marquez, L., Exsudero, G., Martınez, D., and Rigau, G., "Supervised corpus-based methods for WSD," *Word Sense Disambiguation,* Springer, Netherlands, Vol. 33, 2006, pp.167-216

[29] Toutanova, D. K., Ilhan, K., Kamvar, H. T., and Manning C. D., "Combining heterogeneous classifiers for word-sense disambiguation," *In Proceedings of the*

ACL workshop on Word Sense Disambiguation: Recent Successes and Future Directions ,Philadelphia, PA., 2002,pp. 74–80

[30] Boser, B. E., Guyon, I. M., and Vapnik, V. N., "A training algorithm for optimal margin classifiers," *In Proceedings of the 5th Annual Workshop on Computational Learning Theory*, Pittsburgh, 1992,pp. 144–152

[31] Mihalcea, R., and Pedersen, T., "Slides from the AAAI Tutorial – Advances in Word Sense Disambiguation," 2007 Available at: http://www.d.umn.edu/~tpederse/WSDTutorial.html [As accessed on Jan 2014].

[32] Yarowsky, D., "Word-sense disambiguation using statistical models of Roget's Categories trained on large corpora," I*n Proceedings of COLING-92*, 1992, pp. 454–460

[33] Yarowsky, D., and Florian, R., "Evaluating sense disambiguation across diverse parameter spaces,", *Natural Language Engineering*, 2002, pp. 321-334

[34] Lin, D., "Using syntactic dependency as local context to resolve word sense ambiguity," *In Proceedings of ACL/EACL-97,* 1997, pp. 64–71

[35] McRoy, S. W., 1992. "Using multiple knowledge sources for word sense discrimination," *Computational Linguistics,* 1992, pp. 1–30

[36] Bharati, A., Husain, S., Ambati, B., Jain, S., Sharma, D. M., and Sangal, R., "Two semantic features make all the difference in parsing accuracy," *In Proceedings of International Conference on Natural Language Processing*, (ICON-08), Pune, India, 2008.

[37] Pedersen, T., Patwardhan, S., and Michelizzi, J., "Wordnet::similarity: measuring the relatedness of concepts," *Demonstration Papers at HLT-NAACL 2004*, Association for Computational Linguistics, NJ, USA., 2004 pp. 38–41

[38] Wu, Z., and Palmer, M., "Verb semantics and lexical selection." *In 32nd Annual Meeting of the Association for Computational Linguistics*, 1994, pp.133–138

[39] Jiang J., and Conrath D., "Semantic similarity based on corpus statistics and lexical taxonomy," *In Proceedings of International Conference on Research in Computational Linguistics,* Talwall, 1997.

List of Publications

Research Paper Accepted

Swati Kukreja and Shalini Batra, "Resolving Issue of Lexical Disambiguation using WordNet" First International Conference on Networks & Soft Computing(ICNSC-2014), IEEE, 2014.

Research Paper under Review

Swati Kukreja and Shalini Batra, "Comparison of Semantic Relatedness Techniques in WordNet", Third International Conference on Advances in Computing, Communications and Informatics (ICACCI-2014), IEEE, 2014.