Word Sense Disambiguation Using Unsupervised Methods

Project report submitted to

Indian Institute of Information Technology, Nagpur, in

partial fulfillment of the requirements for the award of

the degree of

Bachelor of Technology In

Department of Computer Science and Engineering

by

Tushita Singh(BT16CSE006)

Under the guidance of

Dr.Pooja Jain

HoD(CSE), IIITN

Department of Computer Science And Engineering

Indian Institute of Information Technology, Nagpur 440 006(India)

2020

Specimen- B

Department of Computer Science and Engineering

Indian Institute of Information Technology, Nagpur

Declaration

I, **Tushita Singh** , hereby declare that this project work titled “**Word Sense Disambiguation Using Unsupervised Methods** ” is carried out by me in the Department of Computer Science and Engineering (CSE) of Indian Institute of Information Technology, Nagpur. The work is original and has not been submitted earlier whole or in part for the award of any degree/diploma at this or any other Institution /University.

DATE : 20th June, 2020

BT16CSE006
Tushita Singh

Tsingh

Specimen- C

Declaration

I / We,Tushita Singh, Enrollment No (BT16CSE006), understand that plagiarism is defined as any one or the combination of the following:

1. Uncredited verbatim copying of individual sentences, paragraphs or illustrations (such as graphs, diagrams, etc.) from any source, published or unpublished, including the internet.

2. Uncredited improper paraphrasing of pages or paragraphs (changing a few words or phrases, or rearranging the original sentence order).

3. Credited verbatim copying of a major portion of a paper (or thesis chapter) without clear delineation of who did or wrote what. (Source: IEEE, the institute, Dec.2004) I have made sure that all the ideas, expressions, graphs, diagrams, etc. that are not a result of my own work, are properly credited. Long phrases or sentences that had to be used verbatim from published literature have been clearly identified using quotation marks.

I affirm that no portion of my work can be considered as plagiarism and I take full responsibility if such complaint occurs. I understand fully well the guide of the thesis may not be in a position to check for possibility of such incidences of plagiarism in this body of work.

Date:20th June,2020

BT16CSE006

Tushita Singh

Computer Science And Engineering(CSE)

IIIT,Nagpur

TSingh

भारतीय सूचना प्रौद्योगिकी संस्थान, नागपुर
INDIAN INSTITUTE OF INFORMATION TECHNOLOGY, NAGPUR
"An Institution of National Importance by an Act of Parliament"
RTTC, BSNL Near TV Tower, Beside Balaji Temple, Seminary Hills, Nagpur - 440 006
Website: www.iiitn.ac.in, Email: director@iiitn.ac.in, registrar@iiitn.ac.in Phone: 0712 - 2985010

Thesis Approval Certificate

This is to certify that the project titled "**Word Sense Disambiguation**", is submitted by **Tushita Singh** with enrollment number **BT16CSE006** in the partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science and Engineering**. The work is found to be fit for final evaluation.

Dr. Pooja Jain

(Supervisor Name and Sign)

Name: Dr. Pooja Jain
(Head, Computer Science and Engineering)

Date: 03 / 07 / 2020

Place: IIITN

ACKNOWLEDGEMENTS

I observe my deepest gratitude to Dr.Pooja Jain who has supported and encouraged me in my academic research and career endeavours over the years. Without her, I would not be where I am today professionally nor would this dissertation have ever come to fruition.

I thank her for the opportunities she provided me to work with her in many pioneering efforts developing a working theory of learning and applying it to many different areas of application. Her insights and in-depth knowledge of Current scenario of WSD , as well as her personal encouragement and our conversations, have shaped the direction of this research. The Knowledge imparted by her to me inspired the idea of using Skip-Gram as a learning system for inducing word senses.

Special thanks go to my immediate family. I could not have accomplished this work without them. I am forever grateful for the love and patience of my Family members.

ABSTRACT

Word Sense Disambiguation (WSD) is the identification of the particular meaning for a word based on the context of its usage. WSD is a complex task that is an important component of language processing and information analysis systems in several fields. The best current methods for WSD rely on human input and are limited to a finite set of words. Complicating matters further, language is dynamic and over time usage changes and new words are introduced. Static definitions created by previously defined analyses become outdated or are inadequate to deal with current usage. Fully automated methods are needed both for sense discovery and for distinguishing the sense being used for a word in context to efficiently realise the benefits of WSD across a broader spectrum of language. Skip-gram model is a powerful automated unsupervised learning system that has not been widely applied in this area. The research described in this proposal will apply Skip-Gram techniques in a novel way to the WSD tasks of sense discovery and distinguishing senses in use.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

In the task associated with Natural language processing, the bulk of the engineered time is endowed for text processing. Every document consists of a structure, which are usually Title and paragraph having sentences. When processing these sentences for any kind of further process, we also need to perceive the meaning behind the sentences. Since the sentences are made up words, a number of these words will have either a unique meaning or multiple meanings to it.

Word Sense Disambiguation (WSD) is the process to assign those words having multiple senses with the appropriate meaning, according to the context of the sentence. If we can resolve the subsequent process then it might eliminate the major intermediate activity concerned to attain higher performance of other successive tasks such as Machine Translation(MT), Question-Answers analysis, Sentiment analysis, improving rising searches and Language understanding. Word Sense Disambiguation (WSD) was declared as a definite process task throughout the first days of AI within the Forties, creating it one in all the oldest computer linguistics.

1.1 Word Sense Disambiguation

To Automate the task of word sense disambiguation (WSD) is a difficult as a result of a word does not have distinct senses, they're generally clustered in keeping with the part-of- tag of the word in a given sentences. Even in these cluster they aren't separate from one another and a lot of them are more similar than being totally different. These words having more than one meaning associated to them are called polysemy. Word senses may be thought-about as either coarse-grain or fine-grain, this is based on the information required that may be different enough to divide them [1]. Coarse-grain senses have a lot unrelated distinction between its meaning. Words that have coarse-grain sense are called homographs. Fine-grain senses have little or subtle differences between their meaning, and can pose as a difficulty in assigning the sense due their interrelatedness. Words that have fine grained senses are called polysemous [2]. Mostly words have both coarse and fine grained senses associated with them. Lets us take an example to better understand the need of word sense disambiguation. The most commonly used example is that of "bank", there are both fine grained and coarse grained meaning to it as a noun. The following are the meaning as given by the word net :

Fine - Grained senses	Coarse-Grained senses
a financial institution that accepts deposits and channels the money into lending activities	sloping land (especially the slope beside a body of water
the funds held by a gambling house or the dealer in some gambling games	a financial institution that accepts deposits and channels the money into lending activities
a container (usually with a slot in the top) for keeping money at home	a slope in the turn of a road or track; the outside is higher than the inside in order to reduce the effects of centrifugal force
a building in which the business of banking transacted	a supply or stock held in reserve for future use (especially in emergencies)

Table 1. differentiation of fine-grained and coarse-grained senses using '*bank'* as a example

Now a phrase "I am going to the bank to draw out some money" can have several interpretations to it when presented with its senses. The word "bank" in the above sentence could mean that the person is going to a financial institute or is he taking it out from a piggy bank(i.e usually a container at home) or is he referring to a building dealing in business. Now depending on the context where the word is being used, we decide which sense might be close to appropriate for the word.

Lexical disambiguation in its broadest definition is nothing less than determining the meaning of the word given a context, human readers seem to do this differentiation easily. As a computational problem it is often described as "AI-complete", that is, a problem whose solution pre-supposes a solution to complete natural-language understanding or common-sense reasoning [3]. Sometimes even for a human expert determining the senses of the word can be a challenging task.

1.1.1 A brief history of WSD Research

As we had already mentioned that WSD was identified during the research of Machine Translation in the late 1940s, hence it could be considered as one of the oldest problems in the computer linguistics. Warren Weaver, in his famous 1949 memorandum on translation, first introduced the problem in a computational context as:

> *If one examines the words in a book, one at a time through an opaque mask with a hole in it one word wide, then it is obviously impossible to determine, one at a time, the meaning of words. "Fast" may mean "rapid"; or it may mean "motionless"; and there is no way of telling which.*

Weaver had acknowledged that the context during which the target word is be disambiguated is crucial, and recognised the fundamental statistical character of the problem in proposing that "statistical semantic studies should be undertaken, as a necessary primary step."

The 1950s then saw abundant work in estimating the degree of ambiguity in texts and bilingual dictionaries, and applying simple statistical models. Zipf (1949) published his "Law of Meaning"[4] that accounts for the skewed distribution of words by variety of senses, that is, that additional common words have more senses than less frequent words in a power-law relationship; the relationship has been confirmed for the *British National Corpus* (Edmonds 2005). Kaplan (1950) determined that two words of context on either side of an ambiguous word was reminiscent to entire sentence of context in resolving power [4].

Bar-Hillel (1960) argued that "no existing or imaginable program will enable an electronic computer to determine that the word *pen*" is employed in its 'enclosure' sense within the passage below, due to the necessity to model, in general, all world knowledge like, for example, the relative sizes of objects:

Little John was looking for his toy box. Finally he found it. The box was in the pen. John was very happy.

Ironically, the very "statistical semantics" that Weaver proposed, might have applied in cases such as this: Yarowsky (2000) notes that the trigram *in the pen* is very strongly indicative of the enclosure sense, since one almost never refers to what is in a writing pen, apart for ink.

In the 1970s, WSD was a subtask of semantic interpretation systems developed within the field of artificial intelligence, beginning with Wilks (1975) he developed "preference semantics". The system used selectional restrictions and a frame-based lexical semantics to find a consistent set of word senses for the words contained within a sentence. However, since WSD systems were at the time for the most part rule-based and hand-coded they were prone to a knowledge acquisition bottleneck.

By the 1980s large-scale lexical resources and corpora, such as the Oxford Advanced Learner's Dictionary of Current English (OALD), became available: hand-coding was replaced with knowledge automatically extracted from these resources, but disambiguation was still knowledge-based or dictionary-based. Albeit lexicon methods proved to be useful, the major drawback with them was that they weren't robust since dictionaries lack complete coverage of information on the sense detection on the basis of the context provided.

In the 1990s, the statistical revolution swept through computational linguistics, and WSD became a paradigm problem on which to apply supervised machine learning techniques.It saw three major developments: WordNet[32] became available, the statistical revolution in NLP swept through and the Senseval began. WordNet[32] drove the research forward leaps and bounds. As it was both computationally accessible and the hierarchically organised into word senses called synsets. Today, English WordNet (together with wordnets for other languages) is the most-used general sense inventory in WSD research. Statistical and machine learning methods have been successfully applied to the sense classification problem. Before Senseval, it was extremely difficult to compare and evaluate different systems because of disparities in test words, annotators, sense inventories, and corpora.

The 2000s saw supervised techniques reach a plateau in accuracy, and so attention has shifted to coarser-grained senses, domain adaptation, semi-supervised and unsupervised corpus-based systems, combinations of different methods, and the return of knowledge-based systems via graph-based methods. Still, supervised systems continue to perform best.

1.1.2 Solution Approaches

The problem of WSD can be divided into roughly two aspects : sense definition, that is giving possible senses of the word and then sense identification, where we have to understand the given context of the word. According to Aggirre and Edmonds [4], there are four basic approach to WSD :
1)Knowledge-based : This approach makes extensive use of the Dictionaries or some prior knowledge base available, such as WordNet [5], that has been built manually either by crowd sourcing or available manuals of the language to inform the processing decisions for the identification task.
2)Unsupervised corpus-based : Unsupervised methods for WSD use no external information and work directly with raw non-annotated corpora to induce the senses for words.
3)Supervised corpus-based : Supervised methods for WSD use sense-annotated corpora produced by human lexicographers for training an automated system which is then used for the identification

task. These heavily reply on the training the model, hence having a good sense annotated is crucial part of this approach.

4)Combination approach or Semi-Supervised-based : Semi-or-minimally-supervised methods involve building a disambiguation model based on small amount of human annotated text or word-aligned bilingual corpora and then bootstrapping from this seed data to build additional sense indicators which are then used to identify a sense used in a given context.

It is well established that currently best results for automation of WSD is achieved using supervised methods. However it is known that these methods require extensive sense-tagged training data to be available beforehand. This poses to be both a time consuming and expensive feature in regards to the method. Also another disadvantage with this method would also be that it lacks flexibility in changing between languages, as models can be only be trained for one specific language at a time. It is also known that, systems such as the knowledge-based, supervised and semi-supervised methods require a lot of prior knowledge base and can only be custom built for a particular language. These systems are limited in terms of number of words they can correctly distinguish in a sentence. This again leads to a problem when working with large amounts of text, new domains and new languages. A recent survey of WSD algorithms in 2014 [6] suggests that WSD work best on large volumes of the data. Hence this further restricts the use of supervised methods. Therefore, we turn towards our next best option that is Unsupervised systems, they do no need training data and can adapt to a new language. But they haven’t been well developed or their accuracy lags as compared to the other systems.

1.1.3 Application Areas of WSD

As it is previously seen already that the problem of WSD was discovered while research of Machine Translation was being conducted. This was because a word in a given language can translate to many different words in a different language according to the meanings in the given language. For example, the English word sentence in the legal context translates to one of two different words in Spanish, sentencia or condena, depending on subtle differences in the context in which the word sentence is used [2]. Hence resolving the disambiguation of the sense of the words can improve a Machine Translation(MT) models. As indicated by the ongoing workshops, SemEval and the work devoted to the evaluations of the computational semantic analysis systems [7], there is a challenge in enabling the computer to derive the meaning from the natural language input by the user. Senseval and SemEval are series of workshops created specifically for the evolution of the WSD systems. The specific area of interest that prompted this research is text analysis in educational applications, specifically modelling vocabulary acquisition and reading comprehension to facilitate automatic tutors to maximise learning [6,8,9].

Now-a-days, as education systems are expanding so is the population taking it. With the rapid increase in the students, the task of grading them by a single tutor seems to be a tedious work. Hence the education domains are adapting automation in its systems, such as auto-tutors, these require understanding the user needs and analysing them to give a appropriate feedback to the student to assist them in learning, comprehending a large text and give explanation. Another rising task in this field would be that of question-answer analysis, where a student gives answers according to them and grading has to be done on how many points they were able to cover of the question. In this task the answers would be in different wording but mean the same, so here WSD would be a very useful tool to help grading.

WSD can be key factor in other areas as well such as Information Retrieval, Text mining, message understanding, information extraction, and lexicography. In information retrieval, WSD contributes to distinguishing between relevant and non-relevant documents for a given query. Search engines have been in place for decades and search giants including Google (https://www.google.com), Bing (https://www.bing.com) and Yahoo! (https://www.yahoo.com) are playing a significant role in fulfilling the information needs of users. The syntactic search approach, which takes into account the presence of the query term(s) in the target set of documents and returns only those ones that contain one or more query terms [10], has been used from the beginning of the search engines era. This approach is too rigid and narrow. Text mining benefits from WSD in building systems that analyze text because being able to understand word meanings is essential to text understanding. The capability of extracting certain target information from a body of texts, such as news wire reports, patents, email databases, etc., requires systems that are capable of understanding the language contained in the texts, and therefore the ability to identify the correct sense of a word in a given usage. WSD is vital to lexicography, building intelligent dictionaries, thesauri, and grammar checkers. An automated WSD system would have a potentially broad impact for different communities in computational linguistics, sentiment analysis, and machine learning as well. Any automated system that needs the ability to process text and distinguish the meaning of certain words or contexts would benefit from robust WSD facilities.

1.2 Overview

This research describes the approach of unsupervised learning with regards to word embeddings for the task of words sense induction and disambiguation. The next that is Chapter 2 goes through literature reviews in the unsupervised systems, this will give a better understanding of the proceeding work done. Chapter 3 gives the detailed methodology used to build the system. Chapter 4 the final task of word sense disambiguation, identifying the sense of a word as it is used in a particular context, is covered. It also explains the two results obtained and the discuss the improvements made for the second result. Chapter 5 presents general conclusions arrived at from the experimentation and recommendations for additional follow on research in this area.

CHAPTER 2

2.1 Literature Review

In order to facilitate automated WSD there is a need to develop unsupervised algorithms that rely on no resources beyond original non-annotated monolingual corpora and yield comparable results to existing supervised systems or human understanding. There has been significant work done in developing unsupervised models. Some of these related works has been the inspiration of this research.Word sense induction approaches can be categorised into graph-based models, bayesian, and vector-space ones.

We know that the context in which the target word is found acts like a major clue for getting the sense of the word. Hence it has been observed that finding a relation between the surrounding words of the sentence(context words) with that of the target word, has been the basis of the solution for WSD. In recent times these relation are being expressed by many graph-based algorithms. In a experiment shown in Navigli and Lapata (2010) [11], they built a subgraph of the entire lexicon containing vertices useful for disambiguation and then use graph connectivity measures to determine the most appropriate senses. Mihalcea (2005)[12] and Sinha and Mihalcea (2007) [13] construct a sentence-wise graph, where, for each word every possible sense forms a vertex. Then graph-based iterative ranking and centrality algorithms are applied to find most probable sense. Agirre, Lopez de Lacalle, and Soroa (2014)[14] use personalised page rank over the graphs generated using WordNet.[32]

A common recurrent problem in graph-based WSD algorithms is exponential complexity due to pairwise comparison of senses of all the words in the sentence. The search space for a sentence becomes the product of total number of possible senses of each content word. Due to this exponential search space, many sub-optimal and approximate techniques have been used. Patwardhan, Banerjee, and Pedersen (2003)[15] use sub-optimal word-by-word greedy technique, while approximate techniques such as Simulated Annealing (Cowie, Guthrie, and Guthrie 1992) [16], Conceptual Density (Agirre and Rigau 1996)[17] and approximate solutions of equivalent problems in integer linear programming (Panagiotopoulou et al. 2012) [18] have been tried out. Chaplot and Bhattacharyya[19] algorithm reduces the search space by reducing the number of edges in the graphical model using a dependency parser, which allows us to exactly calculate the optimal solution unlike the above methods, and thereby increasing the accuracy of the system. The use of Dependency parser has made quite an impact, as it can take out the semantic dependency between words this would give an approximate relation between words. For instance, let look at the results given by parsing "I am going to the bank to take out money", when this sentence is passed through Stanford Linker parser following results were observed for relation of target word 'bank' with the context words. 'Take' was relate between 'bank' as object and 'money' as subject. But while drawing the graph there wasn't a directed link between 'bank' and 'money'. And this caused to inference to be cable to capture its relation.

Brody and Lapata (2009)[20] proposed a Bayesian approach modelling the contexts of the ambiguous word as samples from a multinomial distribution over senses which are in turn characterised as distributions over words.

Vector-space models, on the other hand, typically create context vector by using first or second order co-occurrences. Once context vector has been constructed, different clustering algorithms may be applied. However, representing the context with first or second order co-occurrences can be difficult since there are plenty of parameters to be considered such as the order of occurrence, context win-dow size, statistical significance of words in the context window and so on. Instead of dealing with these, [21] suggest representing the context with the most likely substitutes determined by a statistical language model. Statistical language models based on large corpora has been examined in [22,23,24] for unsupervised word sense disambiguation and lexical substitution. Moreover, the best results in unsupervised part-of-speech induction achieved by using substitute vectors (Yatbaz et al., 2012)[25].

In this research, I have proposed a system that represents the context of each target word by using high probability substitutes given by the statistical language model. These substitutes words and their probability are used to create word pairs which is then projected over a bag-of-words. Then each of the projection are weighted against their probabilities and the one with highest score is given as the sense of the word

CHAPTER 3

Algorithm Innovation

The main focus of this research is to develop an unsupervised WSD algorithm that is based on the following assumptions that have been observed in linguistic research:

- Words with similar meanings occur in similar context across a large body of text (Landauer 2007, Landauer 2002, Foltz 1996) [26,27,28].

- Words exhibit the same sense within a document (one sense per discourse) (Stevenson and Wilks 2005, Yarowsky 1995) [29,30].

- Words exhibit only one sense within the context of the few words immediately around them (one sense per collocation) (Stevenson and Wilks 2005, Yarowsky 1995) [29,30].

Given these characteristics, a system capable of clustering words based on contexts given in the corpora, rather than on pre-existing sense inventories, would provide an automated way to both discover and distinguish word senses in a given text corpus.

3.1 Word Embeddings

Now let us understand the term word embeddings. An embedding is a representation of a topological object, such as a manifold, graph, or field, in a certain space in such a way that it's connectivity or algebraic properties are preserved . Presented originally by Bengio et al. (2003), word embeddings aim at representing, i.e., embedding, the ideal semantic space of words in a real-valued continuous vector space. In contrast to traditional distributional techniques, such as Latent Semantic Analysis (Landauer and Dutnais, 1997, LSA) and Latent Dirichlet Allocation (Blei et al., 2003, LDA), Bengio et al. (2003) designed a feed-forward neural network capable of predicting a word given the words preceding (i.e., leading up to) that word. Collobert and Weston (2008) presented a much deeper model consisting of several layers for feature extraction, with the objective of building a general architecture for NLP tasks. A major breakthrough occurred when Mikolov et al. (2013) put forward an efficient algorithm for training embeddings, known as Word2vec. A similar model to Word2vec was presented by Pennington et al. (2014, GloVe), but instead of using latent features for representing words, it makes an explicit representation produced from statistical calculation on word countings.

3.2 Architecture

With basis of the system as the word embeddings for a given target word, we define the workflow of the research. First, I will describe the data enrichment process and then how each sense got its own cluster of recognisable context words. While trying to the find the best fit context words for the target word, we employed the skip-gram model to enrich the context words even more. And the sentence in question was processed for the context words of the target word and projected on each of the senses clustering, here also we made use of skip-gram model to give the similarity between the context words to get score for how much of the sense is applicable to the given input.

3.2.1 Skip-gram model

NLP is a field of Artificial Intelligence in which we try to process human language as text or speech to make computers similar to humans. Humans have a large amount of data written in a very unorganized format. So, it’s difficult for any machine to find meaning from raw text.

To make a machine learn from the raw text we need to transform this data into a vector format which then can easily be processed by our computers. This transformation of raw text into a vector format is known as word representation.

Word representation represents the word in vector space so that if the word vectors are close to one another means that those words are related to one other.As the vocabulary is large and cannot be labeled by the human and hence we require unsupervised learning techniques that can learn the context of any word on its own. Skip-gram is one of the unsupervised learning technique used to find the most related words given the word.

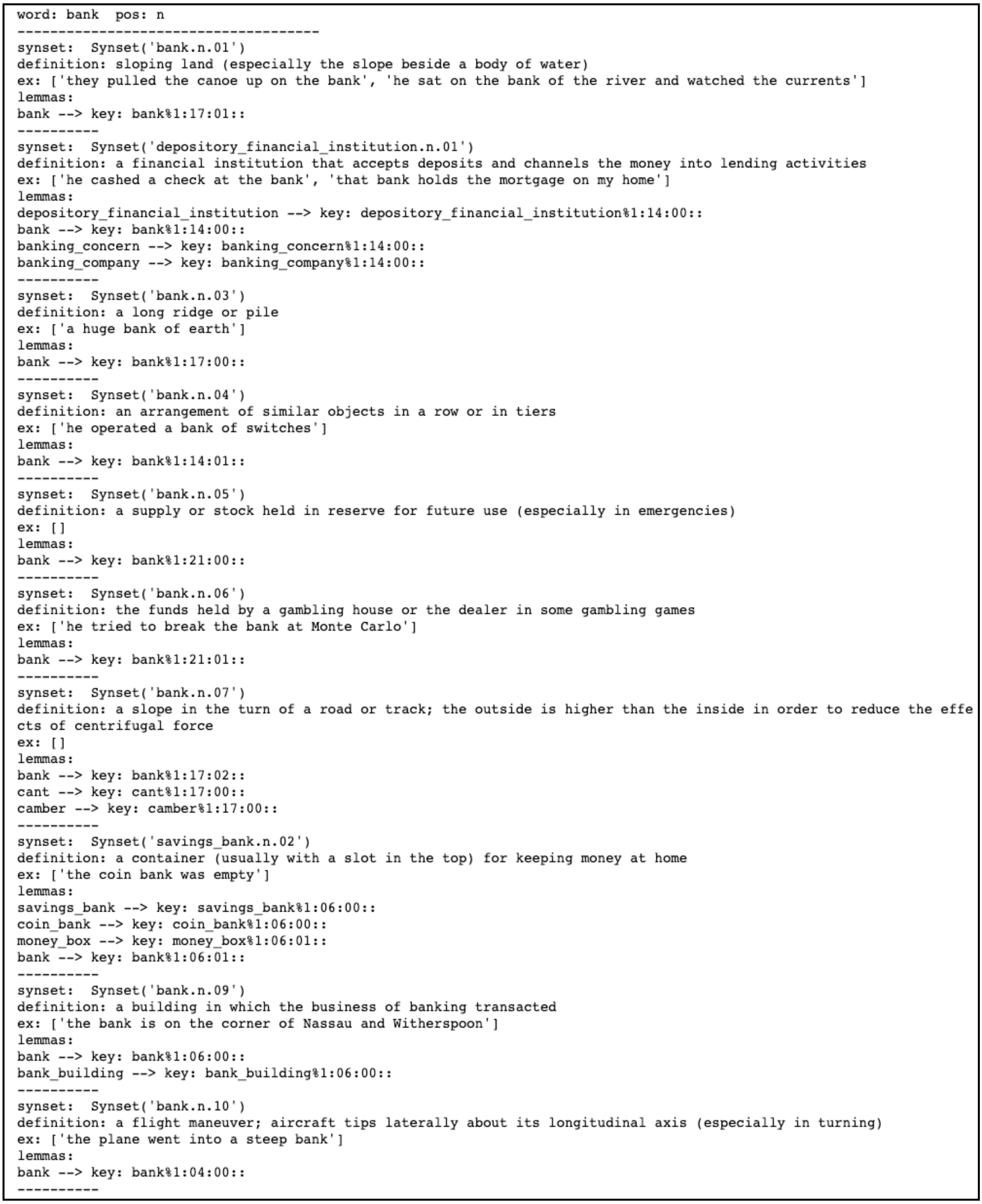
3.3.2 Sketch Engine

For the purpose of research, we had to analyse even more large dataset. But since WordNet only helped us give a base of clustering we still had to randomly collect more sentence from a larger source.

Sketch Engine is a corpus manager and text analysis software developed by Lexical Computing Limited since 2003[31]. Its purpose is to enable people studying language behaviour (lexicographers, researchers in corpus linguistics, translators or language learners) to search large text collections according to complex and linguistically motivated queries. Sketch Engine gained its name after one of the key features, word sketches: one-page, automatic, corpus-derived summaries of a word's grammatical and collocational behaviour. Currently, it supports and provides corpora in 90+ languages.

3.2.3 Data Enrichment

For the process the finding the clustering for each sense we first used made use of WordNet 3.0 to get the information regarding the senses associated with target word. Let us look at the example below to list the amount of information we can receive from word net.



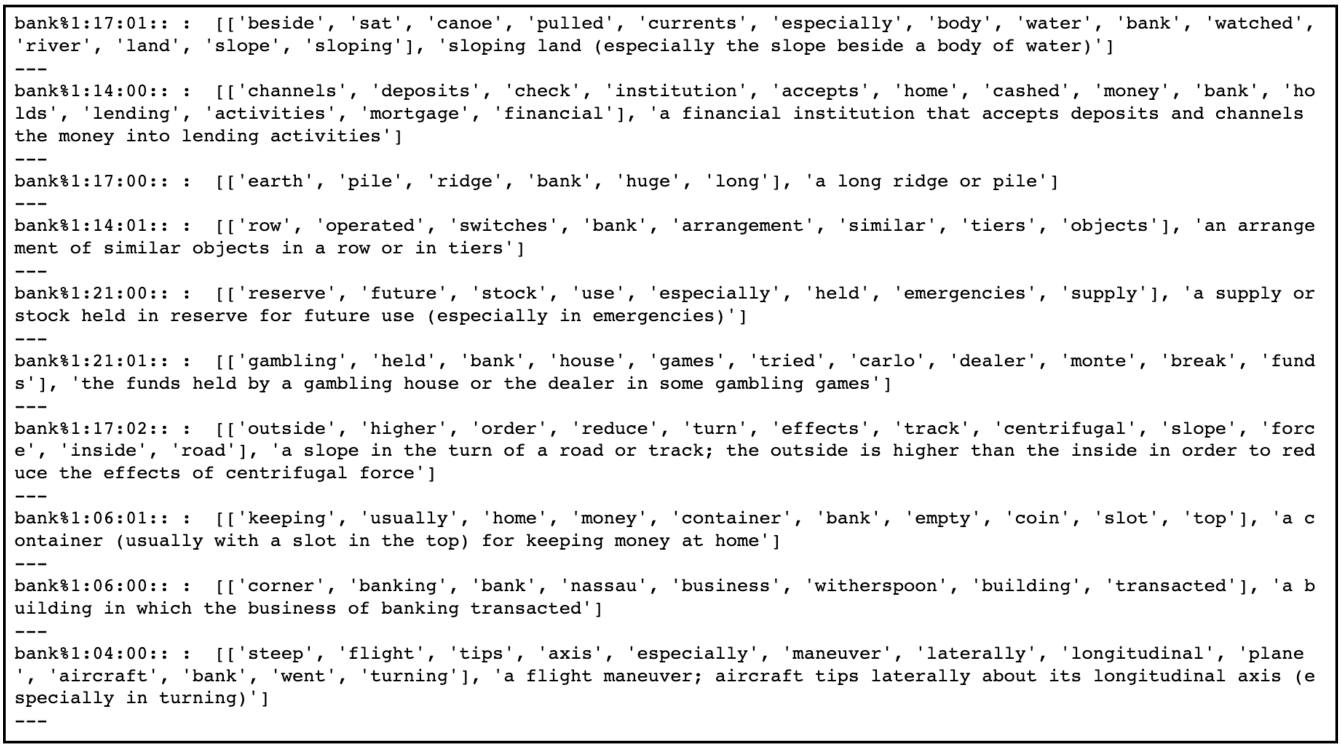
```
word: bank  pos: n
-------------------------------------
synset:  Synset('bank.n.01')
definition: sloping land (especially the slope beside a body of water)
ex: ['they pulled the canoe up on the bank', 'he sat on the bank of the river and watched the currents']
lemmas:
bank --> key: bank%1:17:01::
----------
synset:  Synset('depository_financial_institution.n.01')
definition: a financial institution that accepts deposits and channels the money into lending activities
ex: ['he cashed a check at the bank', 'that bank holds the mortgage on my home']
lemmas:
depository_financial_institution --> key: depository_financial_institution%1:14:00::
bank --> key: bank%1:14:00::
banking_concern --> key: banking_concern%1:14:00::
banking_company --> key: banking_company%1:14:00::
----------
synset:  Synset('bank.n.03')
definition: a long ridge or pile
ex: ['a huge bank of earth']
lemmas:
bank --> key: bank%1:17:00::
----------
synset:  Synset('bank.n.04')
definition: an arrangement of similar objects in a row or in tiers
ex: ['he operated a bank of switches']
lemmas:
bank --> key: bank%1:14:01::
----------
synset:  Synset('bank.n.05')
definition: a supply or stock held in reserve for future use (especially in emergencies)
ex: []
lemmas:
bank --> key: bank%1:21:00::
----------
synset:  Synset('bank.n.06')
definition: the funds held by a gambling house or the dealer in some gambling games
ex: ['he tried to break the bank at Monte Carlo']
lemmas:
bank --> key: bank%1:21:01::
----------
synset:  Synset('bank.n.07')
definition: a slope in the turn of a road or track; the outside is higher than the inside in order to reduce the effe
cts of centrifugal force
ex: []
lemmas:
bank --> key: bank%1:17:02::
cant --> key: cant%1:17:00::
camber --> key: camber%1:17:00::
----------
synset:  Synset('savings_bank.n.02')
definition: a container (usually with a slot in the top) for keeping money at home
ex: ['the coin bank was empty']
lemmas:
savings_bank --> key: savings_bank%1:06:00::
coin_bank --> key: coin_bank%1:06:00::
money_box --> key: money_box%1:06:01::
bank --> key: bank%1:06:01::
----------
synset:  Synset('bank.n.09')
definition: a building in which the business of banking transacted
ex: ['the bank is on the corner of Nassau and Witherspoon']
lemmas:
bank --> key: bank%1:06:00::
bank_building --> key: bank_building%1:06:00::
----------
synset:  Synset('bank.n.10')
definition: a flight maneuver; aircraft tips laterally about its longitudinal axis (especially in turning)
ex: ['the plane went into a steep bank']
lemmas:
bank --> key: bank%1:04:00::
----------
```

*Figure 1. The information given by WordNet for the word '**bank**' having POS tag as 'noun'*

From the information regarding the target word ‘bank’ we make use of examples, lemmas and the definition of each synset to get cluster or bag-of-words for the particular sense. The sentences in examples and definitions are passed through text cleaning process to get the most relevant possible for cluster. The text cleaning process is called normalize which has the following functions associated with it.

- replace_contractions(sentence): Replace contractions in string of text
- tokenize(sentence) : NLTK has the function word_tokenize, which takes a sentence and gives a list of words in them.
- remove_non_ascii(words) : Remove non-ASCII characters from list of tokenised words
- to_lowercase(words) : Convert all characters to lowercase from list of tokenised words
- remove_punctuation(words) : Remove punctuation from list of tokenised words
- replace_numbers(words) : Replace all integer occurrences in list of tokenized words with textual representation.
- remove_stopwords(words) : Remove stop words from list of tokenized words.

Hence we form a basic clustering for each sense in the target word by the above process.



```
bank%1:17:01:: :  [['beside', 'sat', 'canoe', 'pulled', 'currents', 'especially', 'body', 'water', 'bank', 'watched',
'river', 'land', 'slope', 'sloping'], 'sloping land (especially the slope beside a body of water)']
---
bank%1:14:00:: :  [['channels', 'deposits', 'check', 'institution', 'accepts', 'home', 'cashed', 'money', 'bank', 'ho
lds', 'lending', 'activities', 'mortgage', 'financial'], 'a financial institution that accepts deposits and channels
the money into lending activities']
---
bank%1:17:00:: :  [['earth', 'pile', 'ridge', 'bank', 'huge', 'long'], 'a long ridge or pile']
---
bank%1:14:01:: :  [['row', 'operated', 'switches', 'bank', 'arrangement', 'similar', 'tiers', 'objects'], 'an arrange
ment of similar objects in a row or in tiers']
---
bank%1:21:00:: :  [['reserve', 'future', 'stock', 'use', 'especially', 'held', 'emergencies', 'supply'], 'a supply or
stock held in reserve for future use (especially in emergencies)']
---
bank%1:21:01:: :  [['gambling', 'held', 'bank', 'house', 'games', 'tried', 'carlo', 'dealer', 'monte', 'break', 'fund
s'], 'the funds held by a gambling house or the dealer in some gambling games']
---
bank%1:17:02:: :  [['outside', 'higher', 'order', 'reduce', 'turn', 'effects', 'track', 'centrifugal', 'slope', 'forc
e', 'inside', 'road'], 'a slope in the turn of a road or track; the outside is higher than the inside in order to red
uce the effects of centrifugal force']
---
bank%1:06:01:: :  [['keeping', 'usually', 'home', 'money', 'container', 'bank', 'empty', 'coin', 'slot', 'top'], 'a c
ontainer (usually with a slot in the top) for keeping money at home']
---
bank%1:06:00:: :  [['corner', 'banking', 'bank', 'nassau', 'business', 'witherspoon', 'building', 'transacted'], 'a b
uilding in which the business of banking transacted']
---
bank%1:04:00:: :  [['steep', 'flight', 'tips', 'axis', 'especially', 'maneuver', 'laterally', 'longitudinal', 'plane
', 'aircraft', 'bank', 'went', 'turning'], 'a flight maneuver; aircraft tips laterally about its longitudinal axis (e
specially in turning)']
---
```

Figure 2. Basic Clusters formed from the information of WordNet of the given target word

But the amount of context words to help define a sense isn’t enough. So we bring in additional data to expand the database for each sense. Therefore we have named this step as data enrichment technique. For the purpose of expanding the data we need to analyse more raw sentence. We make use of sketch engine to bring more sentences. Assuming that our target word is ‘bank’ in noun form. We randomly fetch 5,000 additional contexts from English Web 2015 (enTenTen15) where our target word occurs with the same part-of-speech tag. This implies that we skip those sentences in which the word ‘bank’ functions as a verb. This is followed for each target word. The request is returned with a dictionary containing left context, target word and right context. Again each of these sentences are cleaned with the help of the normalize function. The list of words returned are then projected on each basic cluster of the senses with the help of skip-gram model. Each word in the list of words is scored to the maximum possible similarity score as given by the skip-gram model. Thereby expanding the basic clustering of the senses.

3.2.4 Methodology

The sentence with the target word is given to the system. If we are not already given with the part-of-speech of the target word, we run the sentence through spacy and get part-of-speech of the target word. Given the target word and part-of-speech(pos), it is processed by the word net to get the basic clustering of the context words for a sense. Then the pair of (target word, pos) is again passed to function to randomly collect sentences by requesting them to sketch engine [31]. The request sends gives back sentences labelling the left context and right context to the target word. Using the contexts we clean them and project it to basic clusters of the senses. With help of Skip-gram similarity function we score the context words against the cluster words and d the cluster with highest score is expanded with these context words. With the help of the expanded context we now repeat the above process for our input sentence and give probability of similarity to each cluster of the sense. The sense having the highest probability is chosen as the sense of the target word.

The window size regarding the context for a word was not taken into consideration. This was because we had observed that the window size for the target context could not be defined after the text was cleaned. But we observed that if we include a distance variable that acts both as penalty and reward to the context words, it increases the accuracy drastically.

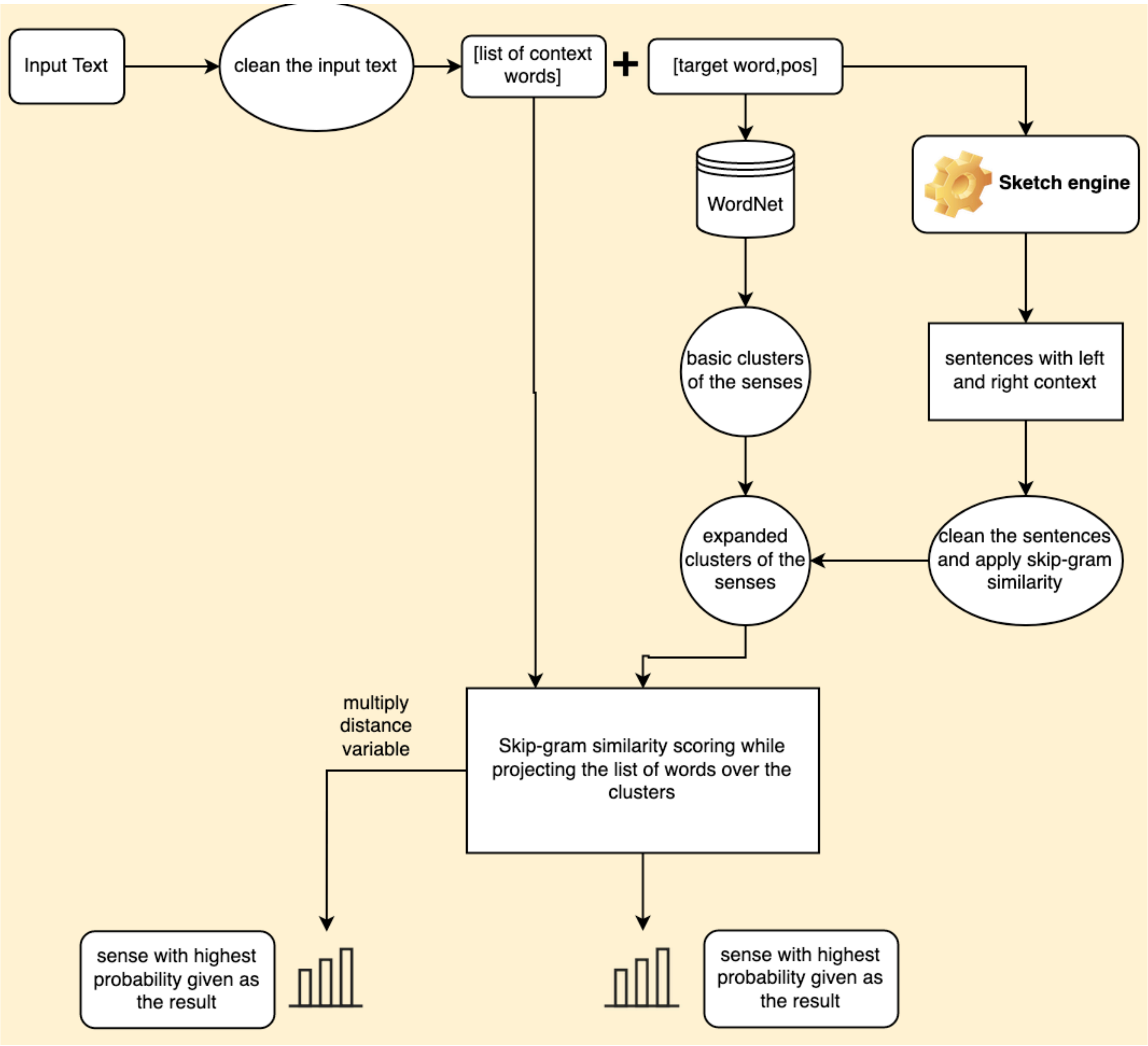

Input Text
clean the input text
[list of context words]
+
[target word,pos]
WordNet
Sketch engine
basic clusters of the senses
sentences with left and right context
expanded clusters of the senses
clean the sentences and apply skip-gram similarity
multiply distance variable
Skip-gram similarity scoring while projecting the list of words over the clusters
sense with highest probability given as the result
sense with highest probability given as the result

Figure 3. The flowchart of the working of the system is represented

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Dataset

The test data for the graded word sense induction task in SemEval-2013 includes 50 terms containing 20 verbs, 20 nouns and 10 adjectives. There are a total of 4664 test instances provided. All evaluation was performed on test instances only. In addition, the organizers provided sense labeled trial data which can be used for tuning. This trial data is a redistribution of the Graded Sense and Usage data set provided by Katrin Erk, Diana McCarthy, and Nicholas Gaylord (Erk et al., 2009). It consists of 8 terms; 3 verbs, 3 nouns, and 2 adjectives all with moderate polysemy (4-7 senses). Each term in trial data has 50 contexts, in total 400 instances provided. Lastly, participants can use ukWaC[1], a 2- billion word web-gathered corpus, for sense induction. Furthermore, unlike in previous WSI tasks, organizers allow participants to use additional contexts not found in the ukWaC under the condition that they submit systems for both using only the ukWaC and with their augmented corpora.

The gold-standard of test data was prepared using WordNet 3.1 by 10 annotators. Since WSI systems report their annotations in a different sense inventory than WordNet 3.1, a mapping procedure should be used first. The organizers use the sense mapping procedure explained in (Jurgens, 2012). This procedure has adopted the supervised evaluation setting of past SemEval WSI Tasks, but the main difference is that the former takes into account applicability weights for each sense which is a necessary for graded word sense.

Evaluation can be divided into two categories: (1) a traditional WSD task for Unsupervised WSD and WSI systems, (2) a clustering comparison setting that evaluates the similarity of the sense inventories for WSI systems. WSD evaluation is made according to three objectives:

- Their ability to detect which senses are applicable (Jaccard Index is used)
- Their ability to rank the applicable senses ac- cording to the level of applicability (Weighted Kendall's is used)
- Their ability to quantify the level of applicability for each sense (Weighted Normalized Discounted Cumulative Gain is used)

Clustering comparison is made by using:

- Fuzzy Normalized Mutual Information: It captures the alignment of the two clusterings independent of the cluster sizes and therefore serves as an effective measure of the ability of an approach to accurately model rare senses.
- Fuzzy B-Cubed: It provides an item-based evaluation that is sensitive to the cluster size skew and effectively captures the expected performance of the system on a dataset where the cluster (i.e., sense) distribution would be equivalent.

4.2 Experiment

4.2.1 Results without Distance Factor

We tested our data with the trail data given in SemEval-2013 Task 13: Word Sense Induction for Graded and Non-Graded Senses. The words contained in the trail data with their pos tags are as follows : paper(noun), interest(noun), win(noun), add(verb), ask(verb), important(adjective), different(adjective), argument(noun). Using Fuzzy B-cubed as the evaluation parameter, we give the following results of our output key with *all-senses.avg-rating.key(Baseline)* - Each instance is assigned all senses with each sense being rated with its average applicability rating in the gold standard and *trial.gold-standard.key(Gold Standard)* - this is standard for getting the correct outputs

word.pos	Precision	Recall	F-score
paper.n	0.871413390788688	0.82	0.9312890407623295
interest.n	0.854916878434403	1.0	0.9217845698357916
win.n	0.87152112971254	1.0	0.931350563855406
add.v	0.8810811395974489	1.0	0.9367816422698281
ask.v	0.8728197712625257	1.0	0.9320915815344377
important.a	0.9012586824214999	1.0	0.9480652903829266
different.a	0.8978547159015929	1.0	0.94617855453183
argument.n	0.8817054981266191	1.0	0.9371344230055383
ALL	0.8779078387772321	0.8525	0.9349850090075421

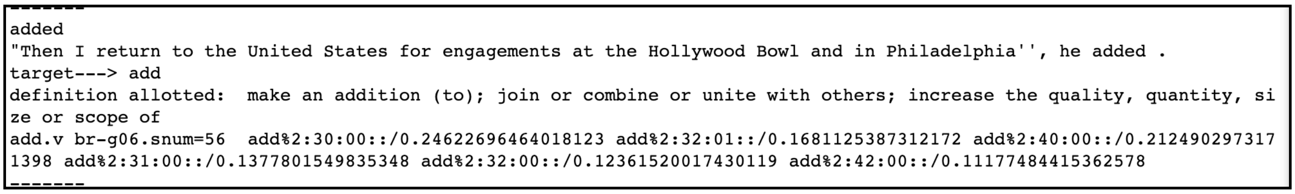
Table 2. Result given by ratings comparison between the baseline and our output

The baseline average score comparison gives us good results. With Recall, that refers to the percentage of total relevant results correctly classified by our algorithm being mostly a perfect 85%. And the precision which means the percentage of your results which are relevant also being close to a 90% . The F-score which is a harmonic mean of precision and recall comes to a satisfactory 93% . Hence we compared it with the gold standard of the dataset.

word.pos	Precision	Recall	F-score
paper.n	0.9955934067688383	0.547870315979729	0.706794810029537
interest.n	0.999973741913878	0.507448410120447	0.673248810642179
win.n	0.972299306618207	0.918546271350238	0.944658742243921
add.v	1.0	0.488718399020478	0.656562583416765
ask.v	0.997643316631851	0.648558501710586	0.786088373207883
important.a	0.990210058868402	0.843322625010546	0.910882640379352
different.a	0.990591745308899	0.892212379671751	0.938831827101854
argument.n	0.999932761061754	0.563617129656992	0.720897025397182
ALL	0.993280542146479	0.676286754065096	0.804690503040537

Table 3. Results obtained when our output is compared to the gold standard

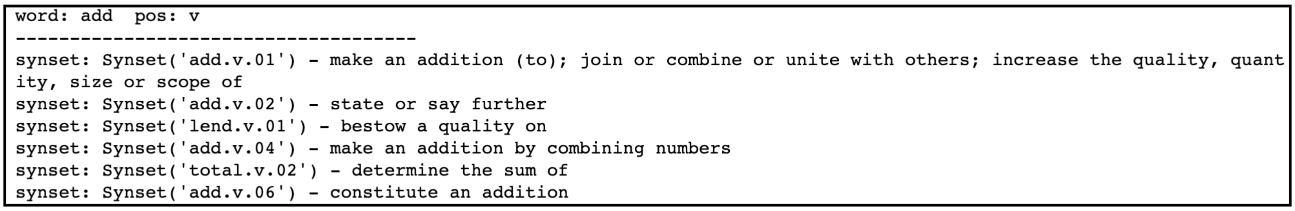
When compared to the standard gold key, there was observed a drastic drop in the Recall but a increase in the precision by a slight percentage. The cumulative F-score was affected by this variation and a drastic drop to 80% was noted. We then tried to pick out the sentences in which our model was not able to perform well. Below are some of the sentences our model performed poorly.



```
-------
added
"Then I return to the United States for engagements at the Hollywood Bowl and in Philadelphia'', he added .
target---> add
definition allotted:  make an addition (to); join or combine or unite with others; increase the quality, quantity, si
ze or scope of
add.v br-g06.snum=56  add%2:30:00::/0.24622696464018123 add%2:32:01::/0.1681125387312172 add%2:40:00::/0.212490297317
1398 add%2:31:00::/0.1377801549835348 add%2:32:00::/0.12361520017430119 add%2:42:00::/0.11177484415362578
-------
```

Figure 4. Result obtained for the sentence "Then I return to the United States for engagements at the Hollywood Bowl and in Philadelphia, he added" without distance component involved

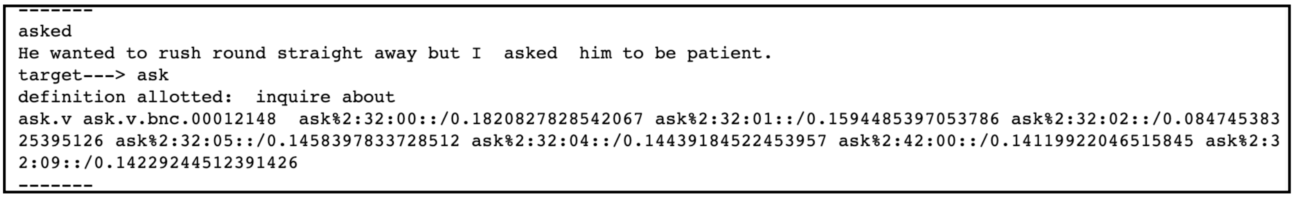
The definition allotted by our model for add isn't correct. The senses for add are as follows :



```
word: add  pos: v
------------------------------------
synset: Synset('add.v.01') - make an addition (to); join or combine or unite with others; increase the quality, quant
ity, size or scope of
synset: Synset('add.v.02') - state or say further
synset: Synset('lend.v.01') - bestow a quality on
synset: Synset('add.v.04') - make an addition by combining numbers
synset: Synset('total.v.02') - determine the sum of
synset: Synset('add.v.06') - constitute an addition
```

Figure 5. WordNet sense for the word 'add' having POS tag as 'verb'

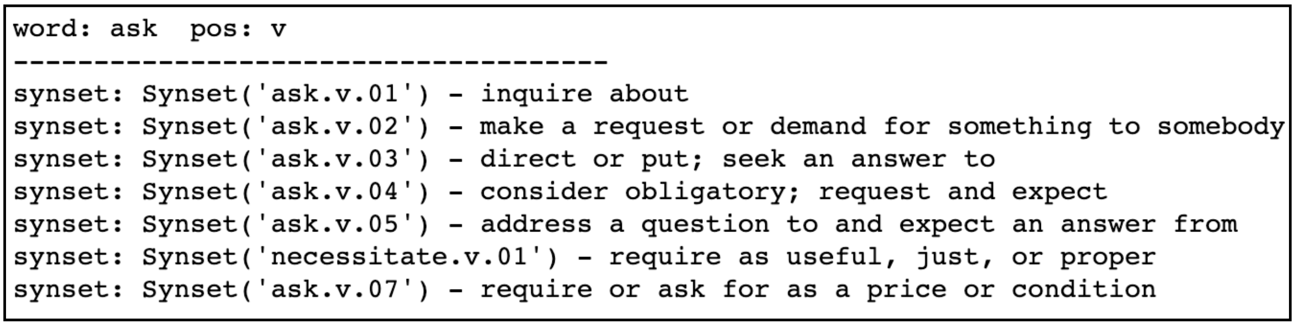
According to the sentence the correct definition for ‘added’ here should be ‘state or say further’. Let us see another example to make some conclusions.



```
-------
asked
He wanted to rush round straight away but I  asked  him to be patient.
target---> ask
definition allotted:  inquire about
ask.v ask.v.bnc.00012148  ask%2:32:00::/0.1820827828542067 ask%2:32:01::/0.1594485397053786 ask%2:32:02::/0.084745383
25395126 ask%2:32:05::/0.1458397833728512 ask%2:32:04::/0.14439184522453957 ask%2:42:00::/0.14119922046515845 ask%2:3
2:09::/0.14229244512391426
-------
```

Figure 6. Results for the sentence ‘He wanted to rush round straight away but I asked him to be patient’ without the distance component

In this example we can see that the target word is ‘asked’. The senses for ‘ask’ as verb given by the WordNet are as follows:



```
word: ask  pos: v
----------------------------------------
synset: Synset('ask.v.01') - inquire about
synset: Synset('ask.v.02') - make a request or demand for something to somebody
synset: Synset('ask.v.03') - direct or put; seek an answer to
synset: Synset('ask.v.04') - consider obligatory; request and expect
synset: Synset('ask.v.05') - address a question to and expect an answer from
synset: Synset('necessitate.v.01') - require as useful, just, or proper
synset: Synset('ask.v.07') - require or ask for as a price or condition
```

Figure 7. WordNet Information regarding the word ‘ask’ with POS tag ‘verb’

From the above senses the more likely definition for ‘asked’ in the above sentence appears to be ‘make a request or demand for something to somebody’.

The major drawback while working with this model was that, if more than one word in the input sentence was semantically similar to the words in the cluster then disambiguating it would become a difficult task, as each will weigh equally. For much clear example lets us observe the sentence

“There is a financial institute on the river bank”

And from Ch. 1 we have seen that ‘bank’ has two senses : 1) sloping land (especially the slope beside a body of water) and 2) a financial institution that accepts deposits and channels the money into lending activities. Now the basic clusters of the two senses are roughly :

```
bank%1:17:01:: :  [['pulled', 'canoe', 'he', 'currents', 'bank',
'they', 'slope', 'watched', 'sat', 'river', 'water', 'beside',
'body', 'land', 'sloping', 'especially'], 'sloping land
(especially the slope beside a body of water)']
---
bank%1:14:00:: :  [['check', 'he', 'home', 'bank', 'cashed',
'activities', 'my', 'holds', 'mortgage', 'accepts', 'lending',
'deposits', 'financial', 'channels', 'money', 'institution'], 'a
financial institution that accepts deposits and channels the money
into lending activities']
```

If we try to disambiguate using the projecting for context words of the input sentence then since '*financial*' and '*institute*' are present they will point to sense 2 that is "a financial institution that accepts deposits and channels the money into lending activities." But the actual sense is that the following sense is "sloping land (especially the slope beside a body of water)". Another observation that can be made is 'river' is in close context than 'financial' and 'institute'. So keeping this in mind we introduced a distance variable to be multiple with score after projection to calculate the final score. And the results obtained due to this can be observed as a slight increase in the F-score.

4.2.2 Results with Distance Factor

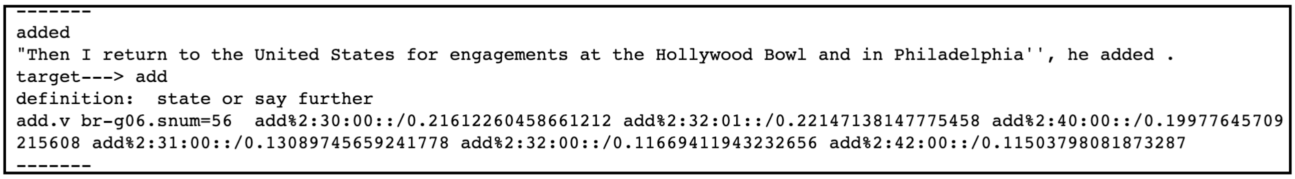
After cleaning the sentence the meaning of having a context window was lost. But we still needed to incorporate some sort of reward-penalty system for words that lie close and far respectively to the target word. lets us take the above example to explain the distance component, after removing stop words from the sentence we are left with ['financial', 'institution', 'river', 'bank'] as the list of context words. Therefor river should be allotted a higher distance score as it is near the target word. Taking target word as the point of start for calculating the distance scores, we will get [1,2,3,0] respectively. We have allotted higher score instead of the traditional [3,2,1,0] because we want to give the near words as a reward system which will increase its over all score.

Following the above convention we were able to improve the scores of the data. There was observed a significant improvement in all F-scores of the words. A detailed information about the scores is given below.

word.pos	Precision	Recall	F-score
paper.n	0.988793147761575	0.575938806256743	0.727900192354584
interest.n	0.9995782801103	0.528052065861187	0.691043323660275
win.n	0.921281802228141	0.952537680100955	0.936649062398314
add.v	0.999224844396822	0.525447800147722	0.688725541472677
ask.v	0.994332653858875	0.688801797120599	0.813836492402757
important.a	0.952063800854314	0.887154640596661	0.918463843159017
different.a	0.941949339956771	0.935052386989831	0.93848819221159
argument.n	0.996716944064969	0.627895846046745	0.770441341680888
ALL	0.974242601653971	0.715110127890056	0.824801995794869

Table 4. Results by rating comparison between gold -standard and output with distance component

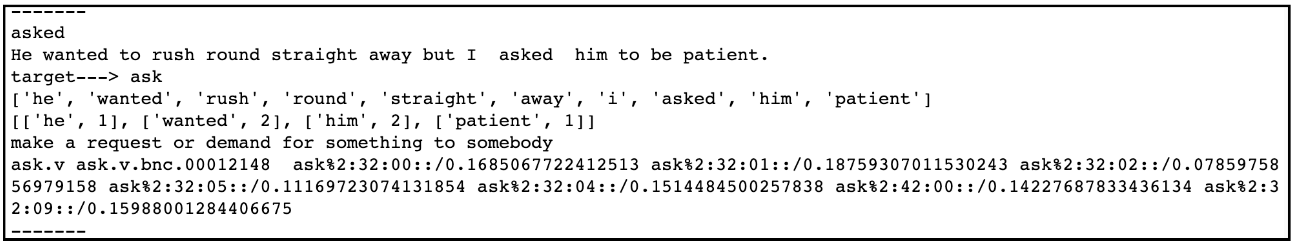
We were able to increase the F-score form 80% to 82%. The recall values also improved in comparison of the above table. The sentences also picked up the correct sense, let us see for the sentences in the above examples.



```
-------
added
"Then I return to the United States for engagements at the Hollywood Bowl and in Philadelphia'', he added .
target---> add
definition:  state or say further
add.v br-g06.snum=56  add%2:30:00::/0.21612260458661212 add%2:32:01::/0.22147138147775458 add%2:40:00::/0.19977645709
215608 add%2:31:00::/0.13089745659241778 add%2:32:00::/0.11669411943232656 add%2:42:00::/0.11503798081873287
-------
```

Figure 8. *Result obtained for the sentence "Then I return to the United States for engagements at the Hollywood Bowl and in Philadelphia, he added" with distance component involved*

As observed the model was improved to give the correct sense to the word 'added' here, which means that the person wanted to further state a point to his previous statement.



```
-------
asked
He wanted to rush round straight away but I  asked  him to be patient.
target---> ask
['he', 'wanted', 'rush', 'round', 'straight', 'away', 'i', 'asked', 'him', 'patient']
[['he', 1], ['wanted', 2], ['him', 2], ['patient', 1]]
make a request or demand for something to somebody
ask.v ask.v.bnc.00012148  ask%2:32:00::/0.1685067722412513 ask%2:32:01::/0.18759307011530243 ask%2:32:02::/0.07859758
56979158 ask%2:32:05::/0.11169723074131854 ask%2:32:04::/0.1514484500257838 ask%2:42:00::/0.14227687833436134 ask%2:3
2:09::/0.15988001284406675
-------
```

Figure 6. Results for the sentence 'He wanted to rush round straight away but I asked him to be patient' with the distance component

Similarly, the correct sense was allotted to the word 'asked', where the person was politely requesting to be patient.

WSD is a difficult problem, but Interestingly, human readers seem to be able to do this innately, differentiating between multiple senses of an ambiguous word given sufficient context. Of course, individual humans also have differences in their ability to perform this disambiguation task given their cognitive background, exposure to spoken and written language, and domain specific vocabulary. Even for human experts, lexicographers, determining the number of senses of a word and giving those senses a definition is a challenging and subjective task.

CHAPTER 5

Summary and conclusions

5.1 Summary

In computational linguistics, **word-sense disambiguation** (**WSD**) is an open problem involved with characteristic that sense of a word is employed in a very sentence. The answer to the current issue impacts alternative computer-related writing, like discourse, rising relevancy of search engines, anaphora resolution, coherence, and abstract thought.

The human brain is kind of quite skilled at guessing the sense a particular word is trying to make in the sentence, word-sense disambiguation. That natural language is formed in a way that requires so much of it as that of reflection that of a neurologic reality. In other words, human language developed an approach that reflects (and additionally has helped to shape) the innate ability provided by the brain's neural networks activities. In computer science and the information technology, it has been a long-run challenge to develop the following ability in computers to do natural language processing and machine learning.

A rich variety of techniques have been researched, from dictionary-based strategies that use the knowledge/information encoded in lexical resources, to supervised machine learning techniques in which a classifier is trained for every distinct word on a corpus of manually sense-annotated examples, to completely unsupervised methods that cluster occurrences of words, thereby inducing word senses.

The difficulties associated with WSD are as follows:

- A wide variety of dictionaries: One drawback with word sense disambiguation is deciding what appropriate senses to be assigned to the target word. In cases like the word *bank above*, at least some senses are obviously different. In other cases, however, the various senses may be closely connected (one meaning being a metaphorical or metonymic extension of another), and in such cases division of words into senses becomes rather more troublesome. Different dictionaries and thesauruses can offer different divisions of words into senses. One solution some researchers have used is to settle a particular lexicon and simply use its set of senses. Generally, however, research analysis giving results using broad distinctions in senses have been much better than those using narrow ones. However, given the shortage of a full-fledged coarse-grained sense inventory, most researchers still continue to work on fine-grained WSD. Most analysis in the field of WSD is performed by using WordNet[32] as a reference sense inventory for English.

WordNet is a large lexical database of English that encodes concepts as synonym sets (e.g. the concept of car is encoded as { car, auto, automobile, machine, motorcar }). For the aim of our research we have made use of WordNet.

- Part-of-speech tagging : In any analysis regarding part-of-speech tagging and sense tagging may be considered very closely connected with each potentially creating constraints to the other. And therefore a question arises, whether these tasks ought to be kept together or decoupled, remains unanimously resolved, but recently scientists incline to check these things separately (e.g. in the Senseval/SemEval competitions parts of speech are provided as input for the text to disambiguate).
- Inter-judge variance : Another drawback that arises is inter-judge variance. WSD systems are usually tested by having their results on a task compared against those of a human. However, while it is comparatively straightforward to assign parts of speech to text, asking people to train to tag senses is far more difficult. While users can memorize all of the potential parts of speech a word can take, it is often seems almost impossible task for a human to memorize all of the senses a word can take. Moreover, It is possible that different humans do not agree on the task at hand – provide them with a list of senses and sentences, and not all humans will not always agree on which sense belongs to the word. As human performance serves as the standard, it is an upper bound for computer performance. Human performance, however, is far higher on coarse-grained than fine-grained distinctions, thus this again proves why research on coarse-grained distinctions has been put to test in recent WSD evaluation exercises.
- Sensibility/Practicality : Some AI researchers like Douglas Lenat argue that one cannot analyze meanings from words without forming some kind of common sense ontology. This linguistic issue is called pragmatics. For example, comparing these two sentences:
  - "Jill and Mary are mothers." – (each is independently a mother).
  - "Jill and Mary are sisters." – (they are sisters of each other).

  To be able to determine the sense for a word we must have some facts about the word, these facts are usually given by the surrounding words(also called context words) to the target word. Moreover we require common sense about the POS tag for the word to eliminate the unwanted list of sense which might be applied for a different POS.
- Sense inventory and algorithms': A task-independent sense inventory is not a coherent concept every task needs its own division of word meaning into senses relevant to the task. For instance, the paradox of 'mouse' (animal or device) is not relevant in English-French machine translation, but has relevancy in information retrieval. The alternative is true for 'river', which needs a

selection choice in French (fleuve 'flows into the sea', or rivière 'flows into a river'). so, completely different algorithms might be required by different applications. In machine translation, the complication takes the shape of target word selection. Here, the "senses" are words in the target language, which frequently correspond to vital meaning distinctions in the source language ("bank" could translate to the French "banque"—that is, 'financial bank' or "rive"—that is, 'edge of river'). In information retrieval, a sense inventory is not essential needed, because it is enough to know that a word is used in the same sense as that of the query and a retrieved document; what sense that is, is unimportant.

- Discreteness of senses : Lastly, the very notion of "word sense" is slippery and debatable. Majority will agree in distinctions at the coarse-grained homograph level (e.g., pen as writing instrument or enclosure), however go down one level to fine-grained ambiguity, and disagreements arise. For instance, in Senseval-2, which used fine-grained sense distinctions, human annotators agreed in only 85% of word occurrences. Word sense acceptance is in principle infinitely variable and context sensitive. It does not divide up simply into distinct or separate sub-meanings. Lexicographers frequently discover in corpora loose and overlapping word meanings, and normal or typical meanings extended, modulated, and exploited in a exceedingly unclear form of ways. The art of lexicography is to generalise from the corpus to definitions that evoke and make a case of the variety in range of meaning of a word, making it appear like words are well-behaved semantically. However, it is not in the slightest degree clear if these same meaning distinctions are applicable in computational applications, as the decisions of lexicographers are usually driven by alternative concerns. In 2009, a task – named lexical substitution – was proposed as a possible solution to the sense discreteness problem. The task consists of providing a substitute for a word in context that preserves the meaning of the original word (potentially, substitutes can be chosen from the full lexicon of the target language, thus overcoming discreteness).

The research method makes use of unsupervised system to get do word sense disambiguation. The main technological stack used for implementation were WordNet 3.0, Sketch Engine and Skip-gram model with pre-trained vectors trained on part of Google News dataset (about 100 billion words). The model contains 300-dimensional vectors for 3 million words and phrases. The input sentence was cleaned and the context for the target word separated. With the target word and its POS tag we formulated cluster for the senses. After that the context words list of the input sentence was projected over the clusters and scored as probability. Two variations of the results were formed, one just the projection and another projection plus the distance between the target and the context words.

5.2 Conclusion

We were able to compare two different results in our research. The first result was obtained when we project the context words of the input on the clusters formed by the senses. We made of expanded clusters since cluster formed with help of WordNet were small. With help of Skip-gram we found out the similarity score between words, this helped us expand our clusters. After we got our first results, the difference were observed. And the need for reward-penalty system was established to give the advantage of a context window size. This system gave a slight boost in getting the correct sense of the word.

5.3 Future Scope

Word sense disambiguation is a large and complicated undertaking. This research has shown promising results for unsupervised sense discovery and sense identification. The following research was conducted only on the sentences. A slight improvement would be to consider it on a paragraph and also the whole document. We can try to apply dependency parsing over long distances in a paragraph to get more context for the target word. It is seen that only with the help of context can we able to determine the sense of the word.

REFERENCES:

[1] Berry M., Mohamed A., Yap B. (eds), Supervised and Unsupervised Learning for Data Science, 1st Ed., Ch. 6, Springer International Publishing, Switzerland AG,(2020).

[2] R. Dale, H. Somers, & H. Moisl (Eds.), Handbook of Natural Language Processing, 2nd Ed., Ch. 14, CRC Press, 6000 Broken Sound Parkway NW, Suite 300 , (2000).

[3] Ide, Nancy & Jean Véronis, Word sense disambiguation: The state of the art, *Computational Linguistics*, 24(1),1–40,(1998).

[4] Eneko Agirre & Philip Edmonds(Eds.), Word Sense Disambiguation : Algorithms and Applications, 1st Ed., Ch. 1, Springer International Publishing, Switzerland AG, (2007).

[5] Fellbaum, C, WordNet, In The Encyclopaedia of Applied Linguistics, John Wiley & Sons, Inc, (2012).

[6] Bhala, R. V., & Abirami, Trends in word sense disambiguation, Artificial Intelligence Review, 42(2), 159–171. http://doi.org/10.1007/s10462- 012-9331-5,(2014).

[7]Alexander Clark, Chris Fox & Shalom Lappin (Eds.), The Handbook of Computational Linguistics and Natural Language Processing, 1st ED., Ch. 11, (pp. 271–295), Wiley- Blackwell, (2010).

[8] Landauer, T. K., Kireyev, K., & Panaccione, C, Word Maturity: A New Metric for Word Knowledge, Scientific Studies of Reading, 15(1), 92– 108. http://doi.org/ 10.1080/10888438.2011.536130 , (2011).

[9] Foltz, P. W., Kintsch, W., & Landauer, T. K, The measurement of textual coherence with latent semantic analysis, Discourse Processes, 28(2-3), 285–307. http://doi.org/ 10.1080/01638539809545029 , (1998).

[10] Y. Lei, V. Uren, E. Motta. Semsearch: A search engine for the semantic web. In EKAW, pp. 238–245, (2006).

[11] Navigli, R., and Lapata, M. An experimental study of graph connectivity for unsupervised word sense disambiguation, *IEEE Trans. Pattern Anal. Mach. Intell.* 32(4):678– 692. (2010).

[12] Mihalcea, R. Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labelling. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, 411–418. Association for Computational Linguistics. (2005).

[13] Sinha, R., and Mihalcea, R. Unsupervised graph- based word sense disambiguation using measures of word semantic similarity. In *Proceedings of the International Conference on Semantic Computing*, 363–369. Washington, DC, USA: IEEE Computer Society. (2007).

[14] Agirre, E.; Lopez de Lacalle, O.; and Soroa, A. Random walks for knowledge-based word sense disambiguation. *Computer Linguist.* 40(1):57–84. (2014).

[15] Patwardhan, S.; Banerjee, S.; and Pedersen, T. 2003. Us- ing measures of semantic relatedness for word sense disambiguation. In Proceedings of the 4th International Conference on Computational Linguistics and Intelligent Text Processing, 241–257. Berlin, Heidelberg: Springer-Verlag.

[16] Cowie, J.; Guthrie, J.; and Guthrie, L. 1992. Lexical disambiguation using simulated annealing. In *Proceedings of the 14th Conference on Computational Linguistics - Volume 1*, 359–365. Stroudsburg, PA, USA: Association for Computational Linguistics.

[17] Agirre, E., and Rigau, G. 1996. Word sense disambiguation using conceptual density. In *Proceedings of the 16th conference on Computational linguistics-Volume 1*, 16–22. Association for Computational Linguistics.

[18] Panagiotopoulou, V.; Varlamis, I.; Androutsopoulos, I.; and Tsatsaronis, G. 2012. Word sense disambiguation as an integer linear programming problem. In *Proceedings of the 7th Hellenic Conference on Artificial Intelligence: Theories and Applications*, 33–40. Berlin, Heidelberg: Springer-Verlag.

[19] Chaplot, D. S.; Bhattacharyya, P.; and Paranjape, A. 2015. Un-supervised word sense disambiguation using markov random field and dependency parser. InTwenty-Ninth AAAI Conference Artificial Intelligence.

[20] Samuel Brody and Mirella Lapata. 2009. Bayesian Word Sense Induction. In Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009), pages 103-111, Athens, Greece.

[21] Baskaya, Osman, Enis Sert, Volkan Cirik and Deniz Yuret. "AI-KU: Using Substitute Vectors and Co-Occurrence Modelling For Word Sense Induction and Disambiguation." SemEval@NAACL-HLT (2013).

[22] Deniz Yuret. 2007. KU: Word sense disambiguation by substitution. In Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval- 2007), pages 207214, Prague, Czech Republic, June. Association for Computational Linguistics.

[23] Tobias Hawker. 2007. USYD: WSD and lexical substitution using the Web 1T corpus In Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007), pages 207214, Prague, Czech Republic, June. Association for Computational Linguistics.

[24] Deniz Yuret and Mehmet Ali Yatbaz. 2010. The noisy channel model for unsupervised word sense disambiguation. *Computational Linguistics,* Volume 36 Is- sue 1, March 2010, pages 111-127.

[25] Deniz Yuret. 2012. FASTSUBS: An Efficient Admissible Algorithm for Finding the Most Likely Lexical Substitutes Using a Statistical Language Model. *Computing Research Repository (CoRR).*

[26] Landauer, T. K. (2007). LSA as a Theory of Meaning. In T. K. Landauer, D. S. McNamara, S. Dennis, & W. Kintsch (Eds.), Handbook of Latent Semantic Analysis (pp. 3–34). Mahwah, New Jersey: Lawrence Erlbaum Associates.

[27] Landauer, T. K. (2002). On the computational basis of learning and cognition: Arguments from LSA. In Psychology of Learning and Motivation (Vol. 41, pp. 43–84). Elsevier Inc.

[28] Foltz, P. W. (1996). Latent Semantic Analysis for Text-based Research. Behaviour Research Methods, Instruments, & Computers, 28(2), 197–202.

[29] Stevenson, M., & Wilks, Y. (2005). Word-Sense Disambiguation. In R. Mitkov (Ed.), The Oxford Handbook of Computational Linguistics (pp. 249– 265). OUP Oxford.

[30] Yarowsky, D. (1995). Unsupervised Word Sense Disambiguation Rivalling Supervised Methods. In Proceedings of the 33rd Annual Meeting on Association for Computational Linguistics (pp. 189–196). Stroudsburg, PA, USA: Association for Computational Linguistics. http://doi.org/10.3115/981658.981684

[31] Adam Kilgarriff, Vít Baisa, Jan Bušta, Miloš Jakubíček, Vojtěch Kovář, Jan Michelfeit, Pavel Rychlý, Vít Suchomel. The Sketch Engine: ten years on. *Lexicography*, 1: 7-36, 2014.

[32] George A. Miller. WordNet: A Lexical Database for English.Communications of the ACM Vol. 38, No. 11: 39-41. (1995).