Word usages and patterns in social media

A thesis presented
by

Spandana Gella

to
The Department of Intelligent Computer Systems
in partial fulfillment of the requirements
for the degree of
Msc Human Language Science and Technology (HLST)

University of Malta
Msida, Malta
Sep 2013

Thesis advisor(s)
Mike Rosner
Dr Tim Baldwin

Author
Spandana Gella

Word usages and patterns in social media

Abstract

Many words in any natural language have more than one meaning. Most of the work related to understanding word meaning in context is based on word sense disambiguation (WSD). Traditional WSD approaches relied on a lexical resource or a sense inventory where each sense is mapped to one of the best fitting senses defined in the resource or sense inventory. In this thesis we investigate and develop an approach for understanding word meanings in contexts over social media texts. In particular we deviate from traditional word sense disambiguation and we target usage similarity, an alternative to WSD for understanding meaning over social media texts. We investigate usage similarity using a topic-modeling based approach in which each usage of a word in a context is represented as a multinomial distribution of topics learned from background collection of documents. We create a gold-standard dataset to evaluate our approach. After evaluating the results over multiple background collections we conclude that it is possible to estimate usage similarity over social media texts. We execute a pilot sense tagging task and analyse sense patterns observed over social media texts. We show future directions which if followed might increase the performance of proposed usage similarity approach.

Contents

List of Figures

List of Tables

Citations to Previously Published Work

Portions of this work have appeared in the following papers:

Spandana Gella, Paul Cook and Bo Han. 2013. Unsupervised Word Usage Similarity in Social Media Texts. In Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity, pages $248-253$. Atlanta, Georgia. **Best short paper award.**

Spandana Gella, Bahar Salehi, Marco Lui, Karl Grieser, Paul Cook and Timothy Baldwin. 2013. UniMelb_NLP-CORE: Integrating predictions from multiple domains and feature sets for estimating semantic textual similarity. In Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity, pages $207-215$. Atlanta, Georgia.

Jey Han Lau, Paul Cook, Diana McCarthy, Spandana Gella and Timothy Baldwin. To appear. Learning Word Sense Distributions, Detecting Unattested Senses and Identifying Novel Senses Using Topic Models. To appear in Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014). Baltimore, Maryland.

Spandana Gella, Paul Cook and Timothy Baldwin. One Sense per Tweeter ... and Other Lexical Semantic Tales of Twitter, (to appear) In Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014), Gothenburg, Sweden.

Acknowledgments

I would like to express my gratitude to Dr. Tim Baldwin and Dr. Paul Cook for their continuous support, encouragement and advice during the thesis supervision. I greatly appreciate Paul for being always reachable, for all the patience to answer my questions and for all the valuable feedback on my reports and work. I would also like to thank Marco Lui, Bo Han and Jey Han Lau for always being available to answer my questions and helping me enhance my experiments. I am also grateful to Dr. Mike Rosner for being very supportive and helpful throughout the course. I would like to acknowledge Dr. Diana McCarthy for the valuable suggestions and feedback on my work.

I am grateful to EM LCT Consortium for providing me funding and a great opportunity of studying at two universities and for all the wonderful collaborations I had during the course. I would like to thank my past and present LCT co-ordinators and Bobbye Pernice for all the administrative assistance.

I would like to thank Dr. Matthew Naylor, Sonia, Suraj, Marina, Daan, Sandro, Kristiina, Kristina, Faustine, Long, Yahya, Deepak, Tiger, Xiwei, Min, Yousef, Bahar, Mel, Mano, Aditi, Mahtab for the great company and support throughout my stay in Malta and Melbourne.

Most importantly I would like to thank my family especially Amma, Nanna, Sriman, Ammama and Jaya aunt for all the support. I must acknowledge Siva without whose love, encouragement and continuous support, I would not have finished the thesis.

To Amma, Nanna and Siva

Chapter 1

Introduction

Recent advances in technology, the popularity of networking websites and internet forums have enabled people to create their own content and share it with others. These applications are referred to as social media applications and some of the most commonly known social media applications are Facebook[1], a social networking site and Twitter[2], a micro blogging website. In recent years, there has been a steep rise in the amount of user-generated content or posts (including status messages and comments). For example Twitter, one of the most popular micro-blogging platforms, has over 500 million posts every single day from about 250 million active users (Greenhow and Gleason 2012; Bennett 2012).

Twitter allows its users to post messages up to 140 characters in length, and the data publicly shared is immediately accessible to others. This data generated over Twitter and other social media applications is rich in information, and has been identified as having potential for many applications such as online trend analysis (Lau *et al.* 2012a), event detection (Osborne *et al.* 2012), and natural disaster response co-ordination (Earle *et al.* 2010). Despite of having been shown to be useful in many applications, processing social media texts stands far behind processing standard English texts. The unique combination of dynamism and short context with conversational nature tends to decrease the performance of many basic natural language processing tools. For example, Ritter *et al.* (2011) showed that the unique characteristics of Twitter tend to decrease the performance of a basic NLP tool part-of-speech tagging.

One reason for this is the unavailability of sophisticated tools and methodologies to understand and process social media texts. For example consider a Twitter message *"@tomA1 must b talkin bout paper but i was thinkin movieezz"* when this is translated into standard English it looks like *"TomA1 must be talking about the paper*

[1]http://www.facebook.com
[2]http://www.twitter.com

but I was thinking movies #weekend." This example shows the difference of Twitter texts and standard English, highlighting the effect of ill-formed words and the format of the sentence. If a natural language processing tool developed for standard English is used to understand the meaning and usage of the word *paper* in this Twitter message it will not be able to interpret the majority of the words and the syntax of the sentence. This shows that there is a huge need for interpreting and understanding data in social media texts.

One possible approach to interpret the social media texts is by understanding the word usages from its context. For example consider two Twitter messages:

(1.1) ne1 heading 2 @blueboys #footy *match* dis weekend? #iamcarlton

(1.2) A one careless *match* can start a forest fire!

In the first message the word *match* refers to the meaning "contest in which two or more teams participate together" whereas in the second message it refers to the meaning "small stick that produces fire". While humans can easily interpret the difference in meanings most of the time it is a difficult task for computer programs to interpret and identify the difference in meanings in the two contexts. In most of the natural languages words exhibit the phenomena of having more than one meaning leading to more than one way of using it.

1.1 Motivation

In this thesis we aim to investigate and understand the meaning of words in social media texts, specifically targeting Twitter messages for better interpretation of the texts. We believe that this would contribute to enhance many natural language processing applications like information retrieval where relevant information could be retrieved over social media texts with a better understanding of word meanings. However, Twitter text is not entirely similar to standard English and some of its characteristics such as its dynamic nature, user-generated content makes our task difficult. Unavailability of sophisticated language processing tools to process social media texts make this task much harder.

We consider this problem in terms of two approaches. The first is to estimate the *usage similarity*, a measure of similarity of two usages of a target word in a given context. Second is to study the *word usage distribution* in Twitter. There is no established or gold-standard dataset available to evaluate usage similarity of words in social media data and this motivated us to create a gold-standard dataset to evaluate the similar systems. We have therefore created a sense-tagged gold-standard dataset to study the sense distributions over social media data. In this we investigate the one predominant sense and one sense per discourse phenomena of words on Twitter. The word meanings are observed to follow a Zipfian distribution in general whereby the occurrence of one particular meaning dominates all other occurrences of the word and is referred to as one predominant sense phenomena. Multiple occurrences of a single word in a document usually tend to be used with the same meaning and this is referred to as one sense per discourse.

In this thesis the main research questions we try to answer are:

> Research Question 1: Can we automatically estimate the similarity of usage of a word in two different short social media texts independent of any lexical resource?
>
> Research Question 2: Does adding relevant context information to Twitter messages help in estimating usage similarity?
>
> Research Question 3: How are senses distributed in Twitter messages? Do they exhibit one predominant sense?
>
> Research Question 4: If all messages from a single user containing a target word within a specific time-frame are considered as a document, does it exhibit one sense per discourse phenomena?
>
> Research Question 5: Does the sense distribution across Twitter messages match the sense distribution of standard English texts?

1.2 Contributions and Outline

In this thesis we focus on estimating word usage similarity over Twitter messages without using any existing sense inventory or labeled corpora. We also study the distribution of senses across Twitter messages using a coarse-grained sense inventory. We give a brief review of the background to this research and related work on both traditional approaches to word sense disambiguation and recent unsupervised approaches which are considered as alternatives to the word sense disambiguation approaches described in Chapter 2.

In Chapter 3 we give a detailed analysis of our proposed method for estimating usage similarity and its performance over different corpora. Given a pair of Twitter messages which contain a target word, we estimate the usage similarity of the target word in the pair of messages. We evaluate the usage similarity measure on a gold standard dataset, which we created using the crowd source platform Amazon Mechanical Turk. We evaluate our proposed approach against a baseline and a benchmark approach. On average, our proposed approach out-performed both the baseline and benchmark methods. Our proposed approach is completely unsupervised and can be adopted for other similar datasets.

We have performed well in estimating usage similarity over Twitter messages compared to what was achieved by Lui *et al.* (2012) over general English. This motivated us to apply our approach to the task of estimating semantic textual similarity. Semantic textual similarity is the task of estimating the semantic equivalence of a pair of texts. A detailed analysis of the task and our submitted systems is also given in Chapter 3.

Our experiments in Chapter 3 showed that standard English texts crawled from the web can be used to estimate usage similarity of Twitter messages. This motivated us to analyse the sense distribution across Twitter messages in Chapter 4 and compare it with sense distribution over web crawled documents. We create a sense tagged corpus of Twitter messages which we later use to analyse the distributions across Twitter versus standard English text. According to our analysis a sense inventory developed for standard English does not adequately capture sense distributions across Twitter messages, as they tend to exhibit higher percentage of novel senses compared to English text. In addition we examine the sense usage of Twitter users to verify if they follow one predominant sense pattern. In Chapter 5 we conclude the thesis and describe possible future work.

Chapter 2

Background

In this chapter we review methods for understanding word meaning in context and traditional ways of dealing with similar tasks. Each distinct usage of a word can be thought of as a discrete meaning or *sense*. Navigli (2009) defined a word sense as "A commonly accepted meaning of a word". In this chapter we discuss word senses, sense representation and the resources that provide word meaning granularity with senses. Word sense disambiguation (WSD) is the computational task of identifying the meaning of a word in a context (Navigli 2009). We overview various supervised, unsupervised, and knowledge-based approaches to WSD that have been proposed to date. We discuss the inapplicability of WSD to social media texts and propose an alternative methodology to understand meaning in context. We give an overview of usage similarity, an alternative methodology to understand meaning in social media texts.

2.1 Word Sense Disambiguation

Word sense disambiguation is the task of associating a word in a context with the most appropriate meaning from a pre-defined set of meanings. For example consider the following Twitter messages

(2.1) ne1 heading to blue boys footy *match* this weekend? #iamcarlton

(2.2) A one careless *match* can start a forest fire!

The respective occurrences of *match* are used with different meaning. The first one corresponds to a *"game in which players or teams compete against each other"* whereas *match* in the second message correspond to *"a small stick that produces a flame"*. It is obvious in most cases for humans to interpret this difference whereas it is a difficult task to distinguish between the two different meanings of the word computationally.

		SENSE INVENTORY	
		Yes	No
LABELLED DATA	Yes	Supervised	-
	No	Knowledge Based	Unsupervised

Table 2.1: Different types of traditional approaches in WSD task

WSD is a very well known and explored problem in natural language processing and is known for its complexity and is described as an AI (Artificial Intelligence) complete problem (Navigli 2009). Every WSD task can be broken down into two steps, first is to determine all possible senses of a word by choosing a sense inventory [3] or learning sense clusters (in unsupervised approaches) whereas the second step is to associate each occurrence of a word with an appropriate sense label (Ide and Véronis 1998).

The Majority of the work on WSD can be categorized based on whether the approach is using a sense inventory or labelled data (shown in Table 2.1). *"A Sense inventory partitions the range of meaning of a word into its senses"* (Navigli 2009). Labelled data refers to the data in which the words are assigned with their corresponding senses. For example consider the Twitter message *loved the roast beef*, sense labelled representation for this message would look like *loved/ENJOY the roast/OVEN_COOKED beef/MEAT*.

In WSD applications if an approach uses labelled data and/or sense inventory it is categorized as below:

- **Knowledge-based approaches:** These methods are based on dictionaries or sense inventories and do not use any corpus based evidences.
- **Supervised approaches:** In supervised approaches various machine learning techniques are used to learn a model using labelled training data (In few instances words are tagged with appropriate sense labels from a sense inventory) and a model trained on labelled data is used to infer senses on the unlabelled data.
- **Unsupervised approaches:** In machine learning, unsupervised approaches try to find hidden patterns in unlabelled data. In the WSD task, unsupervised methods do not use any pre-existing tagged corpora or a sense inventory.

[3] A dictionary or a lexical resource which partitions the range of meaning of a word into senses.

	Name	Details
Thesaurus	Rogets	250,000 word entries organized in six classes and about 1000 categories
	Macquarie	200,000 synonyms
Dictionaries	Longman Dictionary of Contemporary English	Contains 55,000 entries or word definitions.
	Oxford English Dictionary	Contains 170,000 entries covering all varieties of English. This dictionary includes phrases and idioms, semantic relations and subject tags corresponding to nearly 200 major domains.
	Hector	Over 220,000 tokens were and 1,400 dictionary entries were manually analyzed and semantically annotated all of them taken from BNC corpus.
	WordNet	Contains more than 155,327 words corresponding to 117,597 lexicalized concepts, including 4 syntactic categories: nouns, verbs, adjectives and adverbs
Ontologies	Omega Ontology[4]	Constructed to conceptualize WordNet
	UMLS	The Unified Medical Language System[5] (UMLS) is composed of several knowledge sources. The Metathesaurus is a very large, multi-purpose, and multi-lingual, vocabulary database that contains information about biomedical and health-related concepts.

Table 2.2: Structured resources used for WSD

2.1.1 Resources used for WSD

In most WSD approaches, irrespective of being supervised, unsupervised or knowledge-based, knowledge resources play an important role in the task right from learning to evaluation. These resources include syntactic resources such as part-of-speech tags, collocation information (in knowledge-based approaches), structured resources like dictionaries and thesauri (a few of the commonly used structured resources are mentioned in Table 2.2) and unstructured resources like unlabelled or labelled corpora (Table 2.3). The most commonly used unlabelled corpora are the Brown Corpus, British National Corpus (Francis and Kucera 1979) or Wall Street Journal Corpus (Paul and Baker 1992). Whereas the most commonly used labelled corpora are SemCor (Mihalcea 1998) and DSO (Ng and Lee 1996). A detailed description of all knowledge resources is available in Agirre and Edmonds (2006).

The most commonly used sense inventory for WSD tasks is WordNet (Miller

	Name	Details
Unlabelled Corpora	Brown Corpus	a million word bal- anced collection of texts published in the United States in 1961
	British National Corpus (BNC)	a 100 million word collection of written and spoken samples of the English language
	Wall Street Journal (WSJ)	a collection of approximately 30 million words from WSJ
Labelled Corpora	SemCor	Contains 352 documents tagged with around 234,000 sense annotations on all words
	MultiSemCor	An English-Italian parallel corpus annotated with senses from the English and Italian versions of WordNet
	DSO	Corpus created by Defence Science Organisation (DSO) of Singapore, which includes 192,800 sense-tagged tokens of 191 words from the Brown and WSJ corpora
Collocations	Word Sketch Engine[6]	Sketch engine provides the facility of generating collocations from the raw corpora
	BNC Collocations	Collocations generated on British National Corpus
	Web1TCorpus	This corpus provides frequencies for sequences of up to five words in a one trillion word corpus derived from the Web.

Table 2.3: Unstructured resources used for WSD

1995) which is a lexical database which groups English words into sets of synonyms called synsets and provide a short description of synsets along with various semantic relations between them. Sense tagged corpora that are widely used in WSD tasks are built using the fine-grained sense definitions from WordNet are SemCor and DSO (Ng and Lee 1996). Recently many other labelled datasets are available word sense disambiguation tasks from the targeted SenseEval and later evolved SemEval tasks [7] that is these datasets provide sense labels for few targeted words.

2.2 Representation

In WSD word context is used to determine the sense of the word. As text is unstructured source of information a few pre-processing steps are performed to transform the word context into a structured format. These steps include tokenization, part-of-speech tagging, lemmatization, chunking and dependency parsing. An example for the text "The bar was crowded" is shown in Figure 2.1. Usually the word

[7]http://en.wikipedia.org/wiki/SemEval

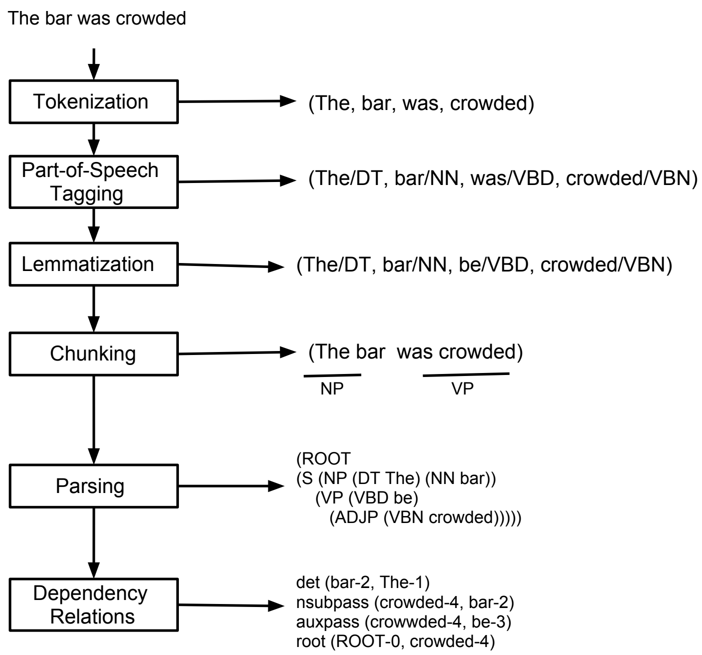


Figure 2.1: An example of transforming text into structured format.

context is represented with a set of features which are based on the structured format of the context. The definition of word context could be varied. Assume that the target word is *bar*, in the Figure 2.1 the context is considered to be the sentence in which target word occurs whereas it could be just word (unigram: *bar*), word with collocations (bigrams: *bar was*, trigrams: *bar was crowded*) or the whole paragraph in which the target word occurs. It is a common approach to consider the sentence in which the target word occurs as the context of the word.

Other features used to represent the context given by Navigli (2009) are local features (includes surrounding words of target word, part-of-speech tags etc,), topical features (include a window of words, phrases, paragraph etc.), syntactic features (include dependency relations of target word with other words in the context etc.) and semantic features (sense information of words in context etc). This form of feature vector representation is used mainly in supervised approaches where a model is learned based on feature vectors and then it is used to label test instances. In many unsupervised approaches context words are represented in a word dimensional space (referred as vector space models).

2.2.1 Distributional Approaches

Most of the supervised or knowledge-based WSD approaches are based on manually created lexical resources. A well studied alternative to distinguish word meanings are distributional approaches. Distributional approaches are built based on distributional hypothesis: "words that occur in similar contexts tend to have similar meanings" defined by Harris (1954). These approaches are built using the context of a word from its occurrences over an unlabelled or raw corpus and are independent of any pre-existing sense inventory. The feature vector used in distributional approaches are based on its context and usually uses the whole sentence of the target word as its context. This was very popular across many corpus-based unsupervised approaches as the existing sense inventories do not always show the corpus/ domain specific sense distinctions (Kilgarrif 1998; Agirre and Edmonds 2006). Distributional approaches has been popular among various other natural language processing applications and have been exploited across various well known problems like language acquisition (Redington and Chater 1997) and compositionality (to understand if the words in a multi word expression contribute to the overall meaning of the expression) by Baldwin *et al.* (2003) and McCarthy *et al.* (2007).

Distributional approaches for WSD tasks cluster the similar contexts for each word based on the distributional similarity. This allows to discriminate between each meaning or usage of the word. Compared to supervised WSD approaches distributional approaches are distinct and work completely independently of sense inventories to discriminate different senses or meanings of each word. Many of the unsupervised approaches investigated to date are based on distributional hypothesis.

2.3 Knowledge-based approaches

Knowledge-based approaches are one of the first proposed and usually rely on measuring the contextual overlap between dictionary based definitions of the target word and its context. For example consider the occurrence of *bank* in "bank balance". The word bank will be assigned the meaning bank#1, as it has the highest overlap with balance meaning definitions.

meaning definitions of word *bank*

- bank#1: a financial institution that people or businesses can keep their money in or borrow money from
- bank#2: a raised area of land along the side of a river
- bank#3: a large collection, especially of information or ideas

meaning definitions of word *balance*

- balance#1: the ability to remain steady in an upright position
- balance#2: the amount of money you have in your bank account
- balance#3: mental or emotional calm

This algorithm is proposed by Lesk and is the basis for many knowledge-based approaches. However, it has the limitation of sensitivity to exact word occurrences in definitions and short dictionary definitions (that is, semantically it might refer to same thing but does not share a common word to show the similarity according to Lesk). Various extensions to the Lesk algorithm have been proposed to overcome its limitations such as sensitivity to exact words or short dictionary definitions including Cowie *et al.* (1992) and Banerjee and Pedersen (2002). Other knowledge-based approaches are based on the semantic similarity of words in a semantic network like WordNet (Resnik 1995; Jiang and Conrath 1997; Leacock and Chodorow 1998; Lin 1998; Pedersen *et al.* 2004) or graph-based techniques based on semantic networks or other similar structured resources. In graph-based approaches disambiguation is performed by applying the page rank algorithm (Page *et al.* 1999) over the graph to find the concepts that relate to the target word. Usually these approaches are unsupervised and some of the well known graph-based approaches have been proposed by Mihalcea (2005), Sinha and Mihalcea (2007) and Agirre and Soroa (2009).

2.4 Supervised Approaches

Most supervised approaches are based on machine learning techniques and usually deal with building a classification model from labelled training data, and this model is used to classify the senses of unlabelled data. Training data contains examples where target words are associated with appropriate pre-defined senses from a sense inventory.

Some of the standard classification techniques used in WSD approaches are:

Decision Trees: A predictive model which is used to represent classification rules with a tree structure that recursively partitions the training dataset (Navigli 2009). In decision trees each internal node represent a condition on a feature value its branch/child represents the outcome of the condition and the terminal node represents the prediction. Mooney (1996) showed that decision trees perform well when specific algorithms are used to obtain decision trees.

Naive Bayes: A naive Bayes classifier is a probabilistic classifier which is based on calculation of conditional probability of each sense of a word given the features represented by its context (Navigli 2009).

Exemplar-based K Nearest Neighbor (kNN) model: In a kNN model when a new instance is given in the form of features, its sense is predicted using the k samples in training data which had similar features as the new instance (Navigli 2009). That is, instead of using all the instances in training data only the k nearest instances are used to predict. This was considered as one of the best performing supervised approaches in exemplar based learning.

Most of the supervised approaches have used WordNet, a fine-grained sense inventory as predefined sense classes, or SemCor and DSO as sense tagged corpora. Supervised approaches have shown to perform well compared to unsupervised approaches (Navigli 2009). However, they pose the same challenges of the knowledge acquisition bottleneck and sensitivity to the domain and application.

2.5 Unsupervised Approaches

Unsupervised WSD methods were also well exploited and address the issue of dependency on a dictionary or a sense inventory. Dependency on manually annotated resources with word senses have been a major overhead for many WSD tasks as they are very expensive to create and would require a different resource based on their domain and application. This problem is defined as *knowledge acquisition bottleneck* (Gale *et al.* 1992).

Most of the unsupervised approaches are based on clustering context/words or on collocation graphs. The basic intuition behind these approaches is based on the distributional hypothesis and each instance is represented as a feature vector of context words and other features like bigrams, dependency relations etc. All vector space models (described in Section 2.10.1), latent variable based models (in Section 2.10.2) fall under unsupervised approaches.

One major subtask of unsupervised WSD approaches is the sense discrimination task or dividing the occurrences of word into clusters based on its meaning or usage. Evaluation of unsupervised approaches, that is quantifying the number of clusters formed and verifying if they actually refer to each unique meanings is a problem faced in unsupervised approaches. There is another line of word sense applications called Word sense induction (WSI) which are very similar to unsupervised approaches where they target on learning the senses or meaning clusters from untagged corpora. Unsupervised approaches are targeted in this thesis as there is no availability of labelled data or any social media specific sense inventories.

2.6 Heuristics

Apart from unsupervised, supervised and knowledge-based approaches there are a few other approaches which are followed to enhance the WSD methods. These are based on heuristics and usually followed in knowledge-based approaches.

Most frequent sense: This approach was inferred after observing that word meanings exhibit Zipfian distribution i.e., one sense occurred much more frequent than others. This heuristic assigns the most frequent sense meaning to all occurrences of the word. This heuristic is often considered as a baseline for evaluating WSD systems (Gale *et al.* 1992). Although this looks promising and straightforward to execute this has its own drawback of limited sense-tagged resources and domain-specific sense distributions. For example a corpus based on the finance, domain may display the most frequent sense of word "bank" to be "a financial institution" whereas a corpus based on agriculture might refer to the sense "the slope beside a body of water". McCarthy *et al.* (2004) has proposed a solution to this using unsupervised approach to learn the most frequent sense in untagged text.

One sense per discourse: This heuristic was proposed by Gale *et al.* (1992). After testing on 9 ambiguous words using a coarse-grained sense inventory they found that the probability of having the same senses for two word occurrences in a discourse or document was 96%. However, this was later strongly opposed by Krovetz (1998) showing that this heuristic does not hold when a fine-grained sense inventory is used, showing that more than 33% of word occurrences have multiple senses per discourse whereas it was reported as 4% by (Gale *et al.* 1992) using coarse-grained senses.

One sense per collocation Collocations are words which occur within a window from the target word in a sentence. This heuristic is based on the idea that words tend to have same meaning when used with the same collocation. For example most occurrences of *match* with collocation *player* refer to the sense of a "game in which players or teams compete against each other". However, this heuristic achieved less impressive accuracy levels (around 70%) when employed with fine-grained senses or higher ambiguity data (Martinez and Agirre 2000).

When the *One sense per discourse* and *one sense per collocation* heuristics were combined with a bootstrapping algorithm by Yarowsky (1995) they were shown to give a significant increase to the performance of the WSD system. Usually, the sense heuristic approaches are combined with supervised approaches and are called semi-supervised approaches.

2.7 Problems with WSD

One of the major criticisms faced by the WSD task is the correct granularity of word senses for general applicability (Kilgarrif 1998). It is often said that WSD systems are made too hard by using fine-grained senses (Ide and Wilks 2006). The heuristics based approaches which perform well using coarse-grained senses do not show higher agreements with fine-grained senses that is, sense granularity has an impact on the performance of the system (Martinez and Agirre 2000; Agirre and Edmonds 2006). This shows that finding sense boundaries or defining sense granularity is a still an unexplained problem.

Another major issue faced by WSD is domain specific resources (Agirre and Stevenson 2006; Kilgarrif 1998). Resnik and Yarowsky (1997) stated that availability of labelled data has been a major reason for improvement in performance over many NLP tasks such as part-of-speech tagging and parsing. The labelled data is mainly used to learn models in supervised learning or can be used for efficient evaluation of the methods proposed. He also stated that many tagged corpora available are small and are tagged on few selected words and this was also considered as a major problem in learning and evaluation WSD approaches.

Another issue is updating the dictionaries or tagged corpora according to the evolution of novel senses. Currently available resources are not updated often and usually do not cover the novel senses. Unsupervised approaches which learn novel sense discriminations from the corpora are able to address this issue. However, evaluating these unsupervised sense learning approaches is a difficult task as it still involves using a labelled data and/or a sense inventory. One other unexplored problem in WSD is the application and understanding of multiple sense labels irrespective of references showing that there exist many occurrences of multiple senses as high as 23-46% of overall sentences studied (Erk *et al.* 2009; Erk *et al.* 2012).

The WSD task is often criticized for its inability to prove its usefulness over applications-oriented tasks (Reddy *et al.* 2011). It is said that WSD task has been made harder by usually testing it with fine-grained sense inventory (Ide and Wilks 2006). There has been efforts in examining the capability of WSD systems being applicable to practical NLP applications (McCarthy *et al.* 2007). Lexical substitution was one of the tasks which was proposed to examine the capability of WSD systems to find alternative words to the target word in a given context.

2.8 Alternatives to WSD

2.8.1 Lexical Substitution

The lexical substitution or *LexSub* task was proposed as an alternative to WSD and to address the meaning similarity for words occurring in a context. In the *LexSub* task the main aim is to identify the substitute of a word occurring in a sentential context i.e., to identify the similar word to the target word instead of assigning a sense to the target word from a sense inventory. For example the word *game* could be given as a substitute for the word *match* in the sentence: "After the match, replace any remaining fluid deficit to prevent problems of chronic dehydration throughout the tournament". Lexical substitution addresses the issue of assessing similarity of two different words in context. That is, to identify the target word alternatives in a given context which could be useful in applications like summarisations and question-answering. This shows that *Lexsub* essentially becomes a WSD task when the target word is polysemous. Although lexical substitution aims at comparing different meanings in context it does not aim to capture different usages. We consider that it is important to capture subtle differences in word senses.

2.8.2 Usage Similarity

One alternative to understanding the meaning of a word is to target usage similarity, which focuses on understanding the similarity of two different usages without depending on a lexical resource or sense inventory. Usage similarity (Usim) is a relatively new task, proposed by Erk *et al.* (2009) to capture the usages of a given word independent of any lexicon or sense inventory. In doing so, it avoids common issues in conventional word sense disambiguation, relating to sense underspecification, the appropriateness of a static sense inventory to a given domain, and the inability to capture similarities/overlaps between word senses. For example consider the following Twitter messages with the target word *paper*:

(2.3) Deportation of Afghan Asylum Seekers from Australia : This **paper** aims to critically evaluate a newly signed agreement.

(2.4) @USER has his number on a piece of **paper** and I walkd off!

The task aims at rating the similarity in usage between two different usages of the same word on an ordinal scale of $1-5$ where 1 indicates the usages are completely different and 5 indicates they are identical. By using this guidelines Erk *et al.* (2009) developed a usage similarity dataset *Usim lexsub* which targets 34 lemmas over the 4 major part-of-speech categories of noun, verb, adverb and adjective.

We believe that usage similarity addresses issues faced by WSD including "How to divide senses" and "Applicability of multiple senses". By not dealing with sense granularity, usage similarity task makes the task easier to relate to any application. However, it is a difficult task for both humans and systems to comprehend as we are targeting to achieve the similarity of usage of each word occurrence.

2.9 Why usage similarity over WSD for social media?

In many WSD and related techniques, context plays a great role in understanding the word usage and the performance of the system. There are many difficulties associated with text in social media as they are short and dynamic in nature. In Section 2.2 a brief overview of context features which are used for WSD tasks are given and these include part-of-speech tag information, dependency relations etc. Social media texts do not have sophisticated tools which could perform well on basic NLP tasks such as part-of-speech or dependency relations. This shows that many WSD approaches cannot be applied on social media texts as they are based on these features. Another obstacle to the applicability of traditional WSD technique is lack of resources in this domain. These considerations suggest that the application of traditional WSD techniques are not feasible over social media data. Instead of investigating sense and its applicability over social media we try to understand the usages of words with usage similarity being used as word meaning annotation.

2.10 Potential Approaches to compute usage similarity

Text in social media is dynamic in nature and is difficult to process even for standard natural language tasks like part-of-speech tagging (Ritter *et al.* 2011). Given the lack of lexical and knowledge-based resources, the only scope to explore the usages is to try unsupervised approaches. It is difficult to process social media texts and generate features like dependency relations on Twitter text and its accuracy level is shown to be poor (Foster *et al.* 2011). The only way left to study meaning in social media applications is to consider various unsupervised approaches using context words as features and learn possible usages from the corpus.

The vector-space models of distributional semantics was proven to be successful to model the meaning of a word based on its context. It was also proven to be useful in many unsupervised WSD approaches. We intend to target vector space models

	problem	**drug**	**case**	**approach**	**report**	**paper**
investigator	35	14	7	6	22	42
researcher	20	40	13	12	19	30
farmer	10	0	2	5	1	3

Table 2.4: Context word count for words *investigator*, *researcher* and *farmer*

and probabilistic bag-of-words approaches as they are simple to implement in an unsupervised fashion. Vector space models have been used in many different ways and as a standard framework to represent a word meaning. Usually the context words are a bag-of-words (Schütze 1998) with or without syntactic dependencies (Thater *et al.* 2011).

In the following sections we represent a vector space model, a latent probabilistic bag-of-words approach and a weight matrix factorization approach that counts weights from missing words to address sparseness issues in latent variable models.

2.10.1 Second Order co-occurrence

In the vector space models, the meaning of a word is represented by a feature vector, with each of its context words as the dimensions (or features). These vectors are also referred to as first order co-occurrence vectors (Schütze 1998). For example, consider the context words that occur with the words *investigator* and *researcher* shown in Table 2.4.

This shows that first-order co-occurrences work well when measuring similarity between words across the whole corpus as context with no regards to a specific usage or sense. In Table 2.4, *investigator* and *researcher* occur with similar words as they share a common meaning whereas the vector for *farmer* has different context words as it doesn't share any common definition with *investigator* and *researcher*. Now consider the words *paper* and *report* in the the following sentences

(2.5) John is an investigator working in this field and the author of this *paper*.

(2.6) Mark is a researcher and important contributor for the *report*.

The meaning of *paper* and *report* described by first order co-occurrence vectors with *investigator* and *researcher* as dimensions is given in Table 2.5.

The similarity between *report* and *paper* in this context is counted as 0 since none of the dimensions are shared. Though we know that the usage of *report* and *paper*

	investigator	researcher	contributor	field	author	work
report	1	0	0	0	1	0
paper	0	1	1	1	0	1

Table 2.5: Context word count for words *report* and *paper*

are similar in examples (2.5) and (2.6), the first order co-occurrence vector does not take this information into consideration when computing similarity between *report* and *paper*. This is a limitation of first order co-occurrence vectors.

To alleviate this problem, Schütze (1998) proposed second order co-occurrence vectors to compute similarity between two words in a smaller context. In the second co-occurrence vector, the first order co-occurrence vector of each context word is summed up together to form a new vector which describes the meaning of the target word in context with the meaning of context words rather than the surface forms of the context words. In the above example, the meaning of *report* and *paper* are given by the equations Equation 2.7 and Equation 2.8. First order co-occurrence vectors for each of the context word w are built from all of the sentences which contain w in the corpus.

$$\begin{aligned}\text{second-order-vector}_{paper} &= 1 \text{ x first-order-vector}_{investigator} \\ &+ 0 \text{ x first-order-vector}_{researcher} + \ldots\end{aligned} \tag{2.7}$$

$$\begin{aligned}\text{second-order-vector}_{report} &= 0 \text{ x first-order-vector}_{investigator} \\ &+ 1 \text{ x first-order-vector}_{researcher} + \ldots\end{aligned} \tag{2.8}$$

This shows that the second order co-occurrence vector transforms the meaning of words from a dimensional space formed by the context words to the dimensional space where context words itself are defined.

Schütze (1998) showed that second-order co-occurrences performed well at clustering word usages. Banerjee and Pedersen (2002) and Patwardhan *et al.* (2003) demonstrated that a variant of Lesk algorithm (Lesk 1986) using second order co-occurrence features performed better than first order co-occurrence features at word sense disambiguation. Purandare and Pedersen (2004) explored second-order co-occurrence vectors for word sense induction and concluded that first order co-occurrence vectors work well with highly frequent words whereas second-order co-occurrence work well with mid and low frequent words.

Since we work with tweets which are known to have fewer context words, we use second order co-occurrence vectors to contrast with our other models. We intend to use this as baseline approach to compare the performance of our proposed approach.

2.10.2 Topic Modeling -LDA

Topic models are generative models for document collections and are built on the idea that each document can be viewed as a finite mixture of topics whereas each topic is a distribution of words that frequently occur with each other (Blei *et al.* 2003; Steyvers and Griffiths 2007).

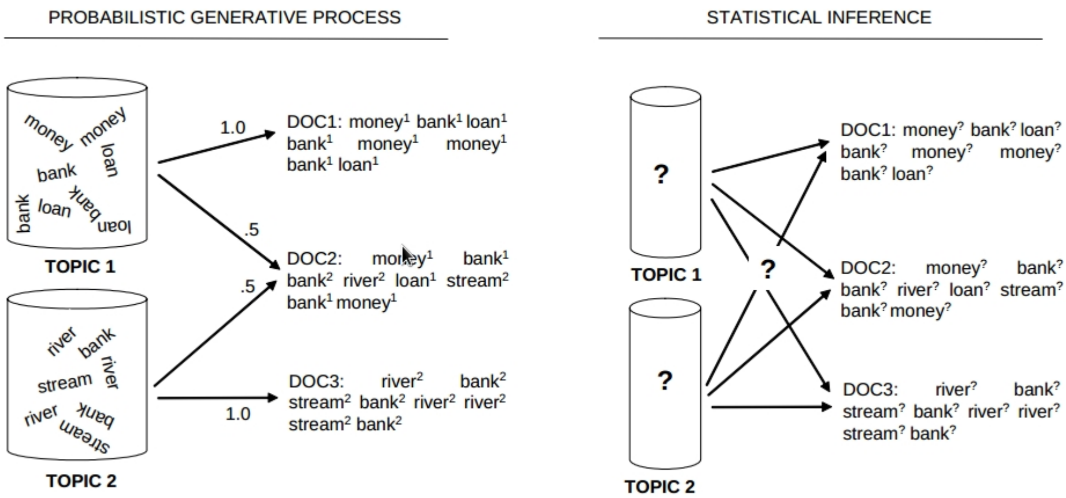


Figure 2.2: An example showing the generative process and the problem of statistical inference underlying topic models from Steyvers and Grifths, 2007.

Although topic models are generative and specify a probabilistic procedure by which documents can be generated usually it is studied in the inverse way. That is finding the best set of latent variables (topics) that can explain the observed words in the documents. This is clearly explained by Steyvers and Griffiths (2007) by analysing topic modeling as two distinct problems of a generative model and a problem of statistical inference. Figure 2.2 illustrates the difference between these two problems. Topic models have been earlier used to represent the word meaning and have been shown to capture the polysemy of words (Steyvers and Griffiths 2007). Topic models were earlier used to induce senses from unlabelled corpus in unsupervised WSD approaches.

Topic modeling is based on a background document collection with D documents where each document d in D contains N_d tokens or words i.e., $(d = (w_1, w_2....w_{N_d}))$ and N denotes the overall number of words i.e., $(N = \sum N_d)$. In a generative model each word w_i in a document is generated by sampling a topic from the topic distribution (where $\theta = P(z)$ denotes the distribution over topics z in a document) followed by choosing a word from the topic-word distribution $P(w_i|z)$. In this notation $P(z_i = j)$ denotes the probability of the j^{th} topic that is sampled for i^{th} word, and $\phi(j) = P(w_i|z_i = j)$ is the probability of word w_i given topic j. Overall the model for the distribution of each word w_i in a document is specified by:

$$P(w_i) = \sum_{i=1}^{T} P(w_i|z_i = j)P(z_i = j) \tag{2.9}$$

In standard LDA , T refers to number of topics and it is set manually whereas other non-parametric variants of LDA are available which learn the number of topics by building the best fitting model (Teh *et al.* 2006). In the LDA process two dirichlet priors are considered the first one being α on the topic distribution of a document θ and the second one β on word-topic distribution ϕ. The parameter α could be interpreted as observation count of the total number of times a topic j is sampled in a document and this determines the smoothing value of topic distribution in every document. The hyper-parameter β is the observation count on the number of times words are sampled from a topic before a word from the corpus is observed and this determines the word distribution in every topic. Both the hyper-parameter values have an impact on topic distributions and word-topic distributions and thus on the topics generated. They are usually set based on number of topics that are being learned and the vocabulary size of the background collection.

Usually the Gibbs sampling algorithm, a specific form of Markov chain Monte Carlo (Gilks *et al.* 1996; Steyvers and Griffiths 2007), is used to extract topics. Gibbs sampling considers each word in the document collection and estimates the probability of assigning the word to a topic with conditional probability on the topic assignments to all other words. This conditional distribution is used to sample a topic and is stored as the new topic assignment for a word. A detailed analysis of Gibbs sampling algorithm is available in Steyvers and Griffiths (2007).

Topic modeling for Usage similarity (Lui): Topic models have been earlier studied by Lui *et al.* (2012) to estimate usage similarity over standard English. They have used LDA based topic model to generate the topic distribution vector of a sentence and used this vector to compute the word usage similarity over a pair of messages which contain the target word. They have experimented with different sizes of context starting from sentence, ± 3, ± 5 sentences i.e., 3 or 5 sentences that occur before and after the target sentence, whole page/document as the context and the whole corpus as single document. They worked on general models (not word-based) with different types of background collections with varying document/context length. Each usage is represented as a multinomial probability distribution of the topics learned and and the cosine similarity of the probability distribution of two different usages is measured as usage similarity.

Lui *et al.* (2012) have used Gibbs algorithm to extract the topics from background collection. However, their methodology was sensitive to the parameters specified as they worked with constant α and β (despite varying the number of documents and vocabulary size), the hyper parameters and the number of topics T were also a parameter to their approach. Similar methodologies could be applied to Twitter data. However, there are issues involved with the context-size, as each document/message in Twitter is no more than 140 characters. Another question is whether to use global or word-based topic models.

2.10.3 Weighted Textual Matrix Factorization

Weighted Textual Matrix Factorization (WTMF) was proposed by Guo and Diab (2012a) to predict the semantic similarity between two texts using weighted matrix factorization (Srebro and Jaakkola 2003). It is proposed to address the sparsity issues faced by latent semantic approaches by modeling the information from missing words in the context. Topic models are often criticized for predicting one dominating topic over short documents producing exactly the same semantic profile for documents sharing common words as long as they share a common topic. This approach addresses the inability of topic models to exploit missing words to create a semantic profile which are usually much more than observed words in short documents.

The intuition behind WTMF is explained with the example below originally presented in Guo and Diab (2012a). Consider the latent semantic profile of the word *bank* which is modeled across the three dimensions sport, finance, institution shown in Table 2.6. In this table R_{obs} and R_{miss} denote the sum of relatedness between latent vector and observed words and missing words respectively. The vector v_0 represented in standard LDA models is chosen by maximizing R_{obs}, and shows that the sentence is only related to finance domain. The second chooses the vector found by the latent semantic approaches which treats both the observed and missing words equally. This is not related to the exact meaning of *bank* in the current context. The third representation shows the ideal one which assigns good weights to related domains and shows substantial decrease in R_{miss}.

WTMF chooses the most appropriate latent semantic vector for each target word which maximises the relatedness value ($R_{obs} - w_m * R_{miss}$) by assigning small weights to the missing words. This missing weight is given as a parameter along with the dimensions in which the latent semantic profile should be created. Thus it captures the information from related missing words.

	finance	sport	institution	R_{obs}	R_{miss}	$R_{obs} - R_{miss}$	$R_{obs} - w_m * R_{miss}$
v_0	1	0	0	20	500	-480	15
v_1	0.6	0.3	0.1	5	100	-95	4
v_2	0.8	0	0.2	18	200	192	16

Table 2.6: The two possible latent vectors for the meaning of bank according to given context. Here missing weight w_m is assumed to be equal to 0.01

2.11 Related Approaches to Distributional Similarity

In this section we describe the recent works that have explored the meaning in context or senses using topic model or similar distributional approaches.

2.11.1 Models for Word meaning Representation

Multiple-Prototype based Approach: The prototype-based context-dependent vector representation of meaning is proposed by Reisinger and Mooney (2010) has shown to accommodate both polysemy and homonymy nature of words. In this approach each word is represented by a set (K) of distinct *sense-specific* vectors. Each word vector is generated by clustering word feature vectors derived from all instances in which a word occurs in a large corpus (they worked with Wikipedia and Gigaword corpus). This approach is very similar to the unsupervised word sense disambiguation approaches whereas they cluster n senses but here the generated clusters are not referred to senses but are intended to capture the meaningful variation of word usage.

Similarity of two words w_1 and w_2 is defined using the minimum distance between one of the w_1 vectors and and one of the w_2 vectors. This way of capturing word usages has shown good results over word meaning similarity and out performed exemplar based approaches. It was also successful in predicting near synonyms whereas they found that when individual senses captured by each prototype is compared with human intuition of a given word they showed negative correlation. They have also tested this multiple prototype approach over Usage similarity dataset (*Usim lexsub*) collected by (Erk *et al.* 2009) dataset and found the correlation to be very low ($\rho = 0.04$). Which shows that using multiple prototype vectors actually do not correspond to human senses.

Probabilistic distribution over latent senses: Dinu and Lapata (2010) have used a probability distribution over latent topics (global senses) to represent the meaning of a word in a context. Unlike Reisinger and Mooney (2010) they represent a word with a single probability distribution vector of senses learned using LDA based topic modeling. They use LDA to induce senses of a words based on global topic models (not word-based) learned from the Gigaword corpus.

In this approach, all the words occur in the document as context words, and the LDA model is trained on that data to obtain sense distributions and context-word distributions for each sense learned. They experimented with high numbers of topics as they are global and not word-based models. They showed that global topics are capable of capturing global senses (not word-based) which could be interpreted as topics that were covered in total over the corpus. This shows that latent topics learned from a corpus is capable of representing a word meaning in context.

Learning latent senses using topic modeling: Word-based topic modeling has been used to automatically induce word senses of a given word by Lau *et al.* (2012b). However, they showed that non-parametric variants of LDA perform better compared to LDA on WSI tasks as they automatically learn the topics (or senses) from the document collection. They also showed that the topics learnt using non-parametric variants of LDA, i.e., HDP (Teh *et al.* 2006) tend to represent word senses better than LDA for WSI tasks.

One another approach which they showed to improve the WSI task or topics learned is to use word positional and dependency relation features. An example of a word positional feature would be crime_#-1, indicating that the word *crime* occurs immediately to the left of the target word. These word positional features increased the performance of both LDA and HDP approaches for WSI. Although this approach looks promising, it is not easy to apply to our data as the documents are shorter in length. And adding word positional features (e.g. crime_#-1, crime_#1 etc.) will increase the sparsity in the dataset (Go *et al.* 2009). Although dependency features have been shown to improve WSD , topic modeling for WSI shows that dependency relations do not improve the performance of the models.

Given the size of our dataset HDP was computationally too expensive, so we have chose a simple LDA based approach by manually choosing number of topics T.

2.12 Summary

In this chapter we reviewed the the background and related work of understanding meaning in context for standard English. We also examine the inapplicability of traditional WSD approaches over Twitter messages. We choose an alternative other than traditional WSD for understanding meaning in Twitter messages and give a brief introduction of the usage similarity method.

In addition we also explain in the approaches of second order co-occurrence, LDA based topic modeling and weighted textual matrix factorization approach we intend to investigate for computationally modeling usage similarity. Additionally we give a brief overview of approaches which used topic modeling or similar distributional approaches to represent word meaning in a context.

Chapter 3

Usage similarity over social media

This chapter we focus on various approaches to estimate usage similarity over social media texts and on creating the gold-standard dataset to evluate proposed approaches. We discuss the necessary measures taken to create a worthy gold-standard dataset. We discuss the results of our proposed approaches over different background collections we experimented on. We also discuss the application of the proposed approach on the tasks similar to usage similarity task.

3.1 Gold-standard Dataset Creation

Usage similarity over social media data is a new task and there is no gold-standard dataset available to evaluate the systems which propose solutions. In order to evaluate how well we can automatically estimate the usage similarity of nouns over Twitter data, we collected gold-standard ratings from human annotators by asking them to quantify how similar a word is being used in a message pair.

In this section, we describe the experimental setup for collecting word usage similarity judgments on English nouns. As discussed in Chapter 2 word usage study is closely related to word senses and usually involves a sense inventory which defines and distinguishes different senses for a given word. Most of the sense inventories which exist today are targeted for general English text and there is no such inventory available for social media data (Section 2.9). We create a dataset which focuses on graded user rating similarities without the use of a sense inventory. This dataset creation is identical to the original Usim dataset (Erk *et al.* 2009) where each annotator makes graded judgments on how similarly a word is being used in a message pair.

Messages used for gold standard dataset creation are sampled from TREC 2011 microblog track dataset[8]. This dataset contains approximately 16 million tweets

[8]http://trec.nist.gov/data/tweets/

(Twitter messages) sampled between January 23rd and February 8th, 2011. For our study we have selected all the nouns from *Usim Lexsub* task (Erk *et al.* 2009) which are *bar*, *charge*, *execution*, *figure*, *field*, *function*, *investigator*, *match*, *paper*, *post*.

3.1.1 Crowd Sourcing

To collect the dataset we relied on crowd-sourcing techniques to create our annotations. We used the Amazon Mechanical Turk (AMT) service which is a crowd-sourcing internet marketplace that enables to perform short paid tasks called as "Human Intelligent Tasks" (HITs) by a pool of non-expert workers. The AMT service has been widely used in linguistic experimentation tasks as it makes it quicker easier and inexpensive. A recent study by Schnoebelen and Kuperman (2010) shows that AMT is a reliable source for linguistic tasks and is heavily used in psycholinguistic tasks. They have analyzed Amazon Mechanical Turk against traditional methods of collecting data and showed that its highly reliable. However we have taken additional precautions in avoiding spam data and to maintain a diverse set of annotations for annotation pair. AMT has also been recently criticized for violating some ethical aspects like lower pay when compared to standard pay rate where the annotators originate from (Fort *et al.* 2011). However, the reason we chose AMT is it has been approved by the University of Melbourne as it satisfies its guidelines of ethics code.

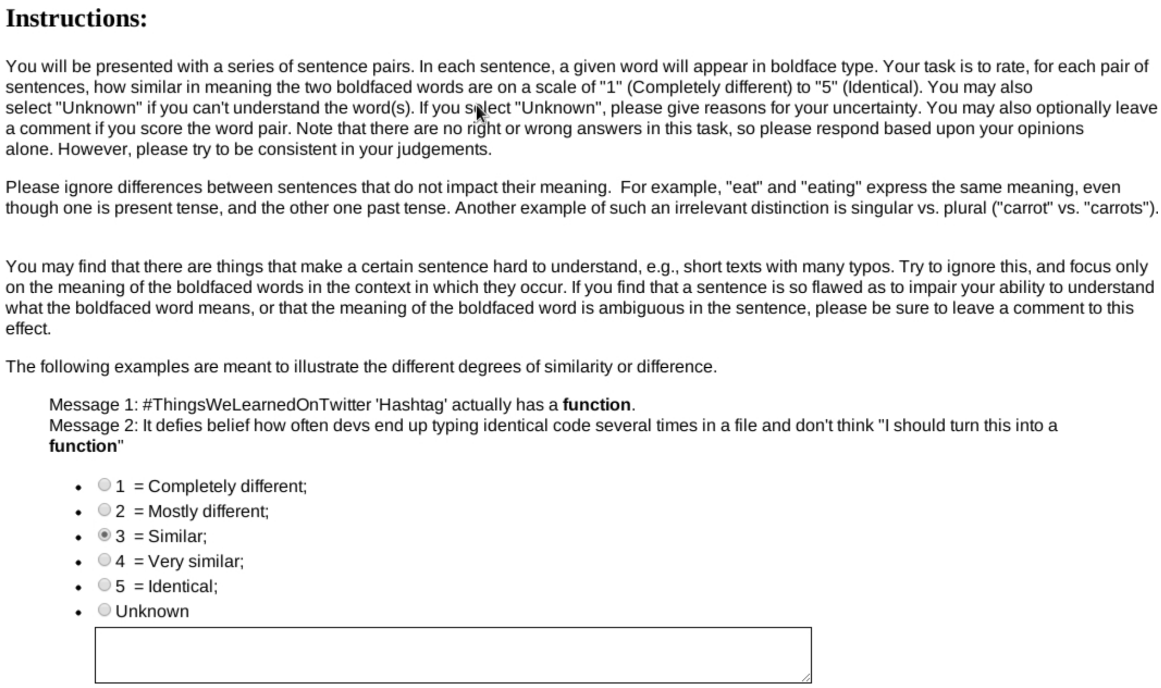
3.1.2 Annotator Settings

In our task all the message pairs were annotated by AMT annotators (Turkers). We have only allowed the turkers who are from United States and have a track record of having 95% of their previous work accepted. This is a preliminary check to ensure that we are considering people who are familiar with English and have good records in doing similar annotation tasks to avoid spammers. Similar to the original Usim Lexsub task (Erk *et al.* 2009) experiment settings we do not require our annotators to have prior knowledge of word senses or usage similarity or similar methodologies.

Annotators are requested to rate a pair of Twitter messages which have one of the target nouns based on their understanding of "how similar is the usage of target noun" in both the messages. In AMT all the annotations are executed at HIT level. Each HIT in our annotation setting comprised 5 message pairs from any of the target nouns and each Turker is allowed to annotate a single hit only once.

3.1.3 Annotators and Annotations

Our annotations instructions are very similar to the original Usim task where we allow annotators to rate the word usage similarity on a graded scale of 1 to 5: 1 being completely different, 2-mostly different, 3-similar, 4-very similar and 5-identical. We



Instructions:

You will be presented with a series of sentence pairs. In each sentence, a given word will appear in boldface type. Your task is to rate, for each pair of sentences, how similar in meaning the two boldfaced words are on a scale of "1" (Completely different) to "5" (Identical). You may also select "Unknown" if you can't understand the word(s). If you select "Unknown", please give reasons for your uncertainty. You may also optionally leave a comment if you score the word pair. Note that there are no right or wrong answers in this task, so please respond based upon your opinions alone. However, please try to be consistent in your judgements.

Please ignore differences between sentences that do not impact their meaning. For example, "eat" and "eating" express the same meaning, even though one is present tense, and the other one past tense. Another example of such an irrelevant distinction is singular vs. plural ("carrot" vs. "carrots").

You may find that there are things that make a certain sentence hard to understand, e.g., short texts with many typos. Try to ignore this, and focus only on the meaning of the boldfaced words in the context in which they occur. If you find that a sentence is so flawed as to impair your ability to understand what the boldfaced word means, or that the meaning of the boldfaced word is ambiguous in the sentence, please be sure to leave a comment to this effect.

The following examples are meant to illustrate the different degrees of similarity or difference.

Message 1: #ThingsWeLearnedOnTwitter 'Hashtag' actually has a **function**.
Message 2: It defies belief how often devs end up typing identical code several times in a file and don't think "I should turn this into a **function**"

- 1 = Completely different;
- 2 = Mostly different;
- 3 = Similar;
- 4 = Very similar;
- 5 = Identical;
- Unknown

Figure 3.1: Screenshot of annotation task for the word *function*

also provide an Unknown option and a comment box along with it to collect optional comments from annotators which can be chosen when the message pair is difficult to interpret. A sample annotation for the word *function* is displayed in Figure 3.1.

In total we have created 110 Mechanical Turk jobs or HITs each containing 5 message pairs. Each hit was annotated by 10 turkers resulting in a total of 5500 annotations. We had 68 turkers participated in our annotation task, each completing between 1 and 100 HITs.

3.1.4 Inter Annotator Agreement

A gold standard dataset is considered worthy only if there are multiple annotators to each task and the Inter-Tagger Agreement (ITA) or Inter-Annotator agreement scores are high enough when compared to similar existing datasets (Kilgarrif 1998). Thus ITA is calculated to examine the reliability of the annotations. ITA is usually known to define the upper bound for a automated system to perform on a particular task and how well a human is able to interpret the task (Navigli 2009). Lower ITA scores show that either the dataset could not be considered as gold-standard or the task itself is uninterpretable and would be very difficult to model via a supervised or unsupervised machine learning approach. ITA is also considered as an indicator for the difficulty of the task of manually assigning senses or meanings (Erk *et al.* 2012; Krishnamurthy and Nicholls 2000). One other important factor that affects the ITA of sense annotation tasks is the sense granularity of the lexical resource used (Hovy *et al.* 2006; Palmer *et al.* 2007; Erk *et al.* 2012). We have overcome this issue by collecting annotations of word usages without using any sense inventory or dictionary senses, just by asking annotators to rate the similarity of meaning across two different usages of the target word. This way of capturing word usage meaning via graded judgments has proven to capture word usage meaning (Erk *et al.* 2012). However this has a huge impact on annotations based on annotator's intuition or understanding of sense granularity. We tried to normalize this by giving proper guidelines and examples. We also study the ITA of usage similarity versus sense distribution (coarse-grained) comparison in Chapter 4.

Difficulty in interpreting the data can be a potential reason for lower ITA scores. We have addressed this issue by choosing the tweets which have enough lexical tokens and are interpretable (see Section 3.1.5). Our main aim here is not to improve ITA, however we made sure that necessary steps have been taken to create a worthy gold standard dataset.

Unlike *Usim lexsub* task we had large number of annotators and each annotator have performed a different number of annotations. So reporting ITA over each annotator pair is difficult. So we report weighted mean average ITA score over all annotators. We measure ITA using Spearman rank correlation which is a non-parametric evaluation of ranks as they are graded ratings. This measure was chosen following Mitchell and Lapata (2008), Erk *et al.* (2009), Erk *et al.* (2012) who have used the same measure in evaluating ITA for similar graded tasks.

3.1.5 Data Sampling

For creating the gold standard datase,t messages were sampled from the TREC 2011 dataset messages. One major issue faced during sampling the messages was tweets containing a large percentage of noisy words and messages which are very short in length. The noisy text and ungrammatical format makes it difficult to comprehend the word usages from the context. This might affect the annotations of the turkers. To have a harmonization in the dataset and to overcome the issue of interpretability we sampled the tweets based on heuristics below. Detailed steps of pre-processing are mentioned in Section 3.4.

1. Classified as English tweet based on `langid.py` language identification tool (Lui 2012).

2. Has at-least 4 content words (categorized as nouns, verbs, adjectives or adverbs) after POS tagging. We have used CMU Twitter POS tagger by Owoputi *et al.* (2012).

3. Contains the target word categorized as noun.

   Our study in this thesis is based on word usage in social media which are nouns. So we have filtered only the tweets which has our target word categorized as nouns by POS tagger.

4. Contains at-least 70% of its post-normalization tokens in an English dictionary. We have used Aspell[9] dictionary to verify the post-normalized tokens.

   One of the major difficulties faced by Natural Language Processing applications on social media is its noisy text and conversational nature. In this study we are limiting to study the tweets which have a good proportion of English tokens post-normalization (excluding URLs, usernames and hashtags). Lexical normalization is the task of mapping the ill-formed or noisy form of the word to its original form. Few examples of ill formed words before and post normalization is 'makin' → 'making', 'coooooool' → 'cool', '2mwrw' → 'tomorrow'. We have tried to identify the noisy text of lexical variants by mapping the words using the normalisation dictionary of Han *et al.* (2012b). The normalisation dictionary provided by Han *et al.* (2012b) contained about 40k pairs of normalised pairs which were mined from 80 million English tweets from September 2010 to January 2011.

In total we sampled 55 pairs of messages for each noun which satisfied the above heuristics, comprising a total of 550 message pairs as our study is based on 10 target nouns. Note that we have not altered the Twitter messages with post-normalization tokens, we have used these heuristics to select the messages that have sufficient linguistic content to include in our gold standard dataset.

3.1.6 Spam detection

Annotations collected using a crowd source are prone to spam (Kazai and Milic-Frayling 2009; Yuen *et al.* 2011; Vuurens *et al.* 2011). We used the average Spearman correlation score (ρ) over each annotation to detect outliers or spam. We calculated the average Spearman correlation score of every annotator by correlating their annotation values with every other annotator who rated the same message pairs. We accepted the work of all annotators whose average correlation ρ is greater than 0.6,

[9] http://aspell.net/

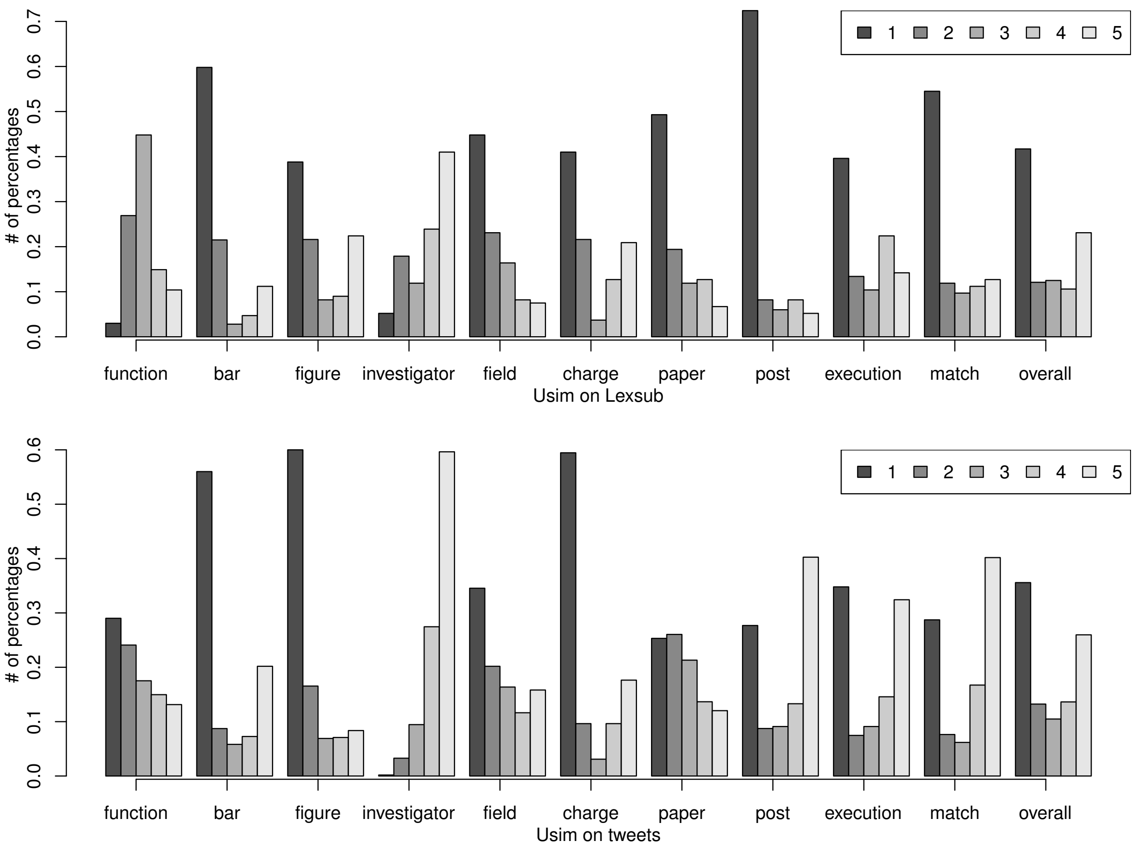


Figure 3.2: Rating distribution of Usim per word and overall in two datasets

95% of our annotators had correlation score greater than 0.6 . We have chosen 0.6 as our threshold correlation score as most of our annotators had greater than 0.6. Similarly negative correlation score shows that they are completely disagreeing with all the other annotators which shows that the annotation was executed randomly or indicates spam. So, we rejected the work of annotators whose average correlation score was negative. Only two annotators and a total of 4 HITs were rejected using this heuristic. For the rest of the annotators (i.e., whose $\rho \geq 0$ and $\rho \leq 0.6$) we accepted each of their HITs only if at least 2 out of 5 of the annotations for that HIT were within ± 2.0 of the mean for that annotation based on judgments of the other turkers. In total 21 HITs were rejected based on this heuristic. We also eliminated 7 annotations which had incomplete judgments. In total only 32 HITs were rejected, which is around 3% of the whole dataset. This shows that our filtering measures were not biased and originality in the annotation was maintained.

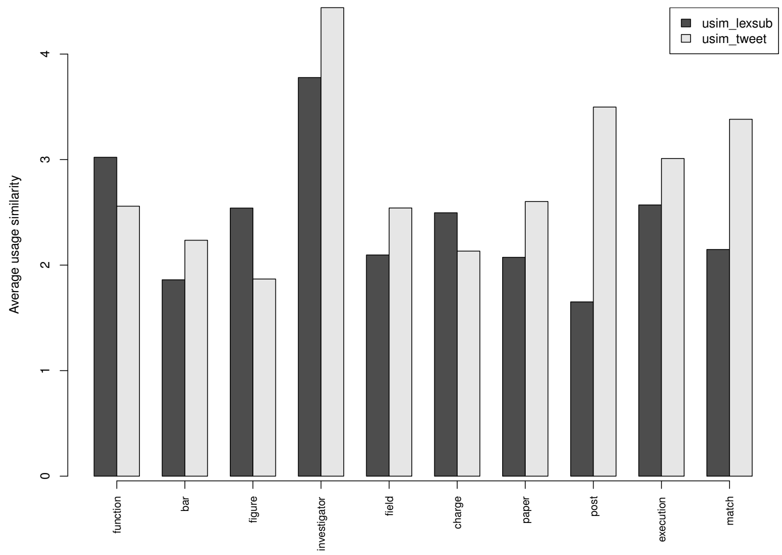


Figure 3.3: Domain difference in overall usage similarity rating average mean per word

3.1.7 Agreement Scores

The weighted average Spearman correlation or ITA over all annotators who had at least two common annotations after the filtering is 0.681. It is interesting to observe that weighted ITA of *Usim tweet* is almost similar to the inter annotator agreement of 0.687 on nouns reported by (Erk *et al.* 2009) on *Usim Lexsub* despite having a large number of untrained annotators and sentence pairs.

3.2 Usim Data Difference Analysis

In this section we present the overall rating distribution of usage similarity across our *Usim tweet* and *Usim Lexsub* collected by Erk *et al.* (2009). This study is an extension to the work by Han *et al.* (2012a) which was on a sample of the *Usim tweet* dataset. They worked on the same 10 target lemmas but they performed analysis on annotations of only 5 sentences for each lemma. We extend their study onto our larger dataset and compare the distribution of ratings across general English *Usim Lexsub* dataset and social media messages *Usim tweets*.

We compare the ratings of 135 (45 message pairs x 3 annotators) *Usim Lexsub* annotations against 550 (55 message pairs x 10 annotators) social media annotations over each lemma. Figure 3.2 shows the difference between the overall and per lemma rating distribution across the two datasets. The overall distribution in both the datasets look similar despite having different distributions for each lemma. The overall mean ratings of each lemma across the two datasets is shown in Figure 3.3. Both distributions show that although for individual lemmas distribution differs they have a similar overall distribution and are comparable. In Section 3.6 we analyze our results against the results of Lui *et al.* (2012) whose experiments are based on *Usim Lexsub*.

We calculate the Jensen-Shannon divergence that is the distance between the rating distributions with Equation 3.2 and entropy fluctuations across two datasets for each lemma. Table 3.1 shows the Jensen-Shannon divergence across two datasets and Table 3.2 shows the entropy difference for each lemma and overall across two datasets .

$$KL(v||m) = \sum_{i=1}^{5} v_i log(v_i/m_i) \tag{3.1}$$

$$JS(v,w) = KL(v||m) + KL(w||m) \qquad \text{where m= } 1/2x(v+w) \tag{3.2}$$

Lower divergence scores show that all the lemmas have similar usage distribution in both the domains fofor lemmas *field*, *charge*, *bar*, *match*. Higher divergence and entropy difference scores show that lemmas *post* and *function* show higher difference in distribution in social media and general English. In Figure 3.2 *post* has one of the extremes dominated in *Usim Lexsub* where as it is skewed in social media showing that messages sampled in *Usim tweet* were more related than the ones sampled in *Usim Lexsub*.

We have observed an interesting distribution for lemmas in social media that in most cases one or both of the extremes are dominating the distribution except for the lemmas *paper* and *function*. This shows that the usages in the sampled Twitter messages are mostly similar to each other or different from each other whereas this was not the case over *Usim Lexsub*.

3.3 Background Corpora

In this work, we present three different corpora derived from the Twitter public streaming API[10] from February 1^{st} 2012 to February 29^{th} 2012 and a standard English

[10] https://dev.twitter.com/docs/streaming-apis

Word	Number	Word	Number
function	0.055	investigator	0.04
figure	0.02	charge	0.015
post	0.081	execution	0.017
bar	0.016	paper	0.021
match	0.017	field	0.008
overall	0.002		

Table 3.1: Jensen-Shannon divergence of rating distributions for each word in two datasets

Word	Entropy diff.	Word	Entropy diff.
function	0.221	investigator	−0.411
figure	−0.271	charge	−0.236
post	0.454	execution	−0.037
bar	0.089	paper	0.191
match	0.074	field	0.134
overall	0.036		

Table 3.2: Entropy difference of rating distributions (*Usim tweets* subtracts *Usim Lexsub*) for each word in two datasets.

corpus derived from Web. All the Twitter messages based corpora are entirely new and collected specifically for this work. All the tweets in the background corpora pass the pre-processing steps. We have also stemmed the words so that they could be mapped to their base word forms and excluded URLs, emoticons, hash tags and user mentions when stemming. A brief overview of three Twitter messages based corpora and I-EN corpora is described in Figure 3.4

Twitter contains significant number of tweets which are non English (Hong *et al.* 2011). As our study is specified to English tweets, we filter them using a language identification tool. When compared to standard English, Twitter messages are short, conversational in nature and contain many typos and ill-formed words (Han *et al.* 2012b). Language identification tools developed for general English might not be suitable for language identification on Twitter messages. There is no available language identification tool which is made specifically for social media data. However, `langid.py` (Lui *et al.* 2012) has been shown to have high performance over Twitter compared to other existing state-of-the art systems and is easy to use off-the-shelf tool and faster as well. We perform the language identification using the `langid.py`

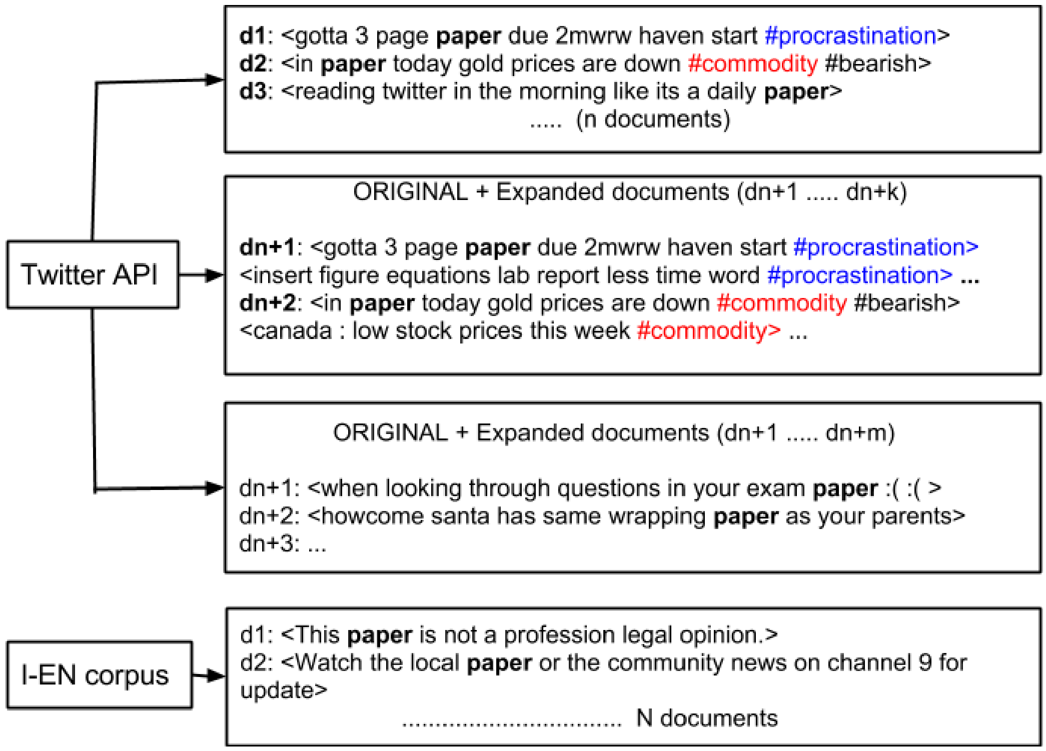


Figure 3.4: An example of each background corpora for the word **paper**.

tool.

3.3.1 Original

The ORIGINAL corpus is based on tweets from the Twitter streaming API. It consists of 10 word-based sub corpora based on the target words. All the tweets which are available via the streaming API were filtered to contain one of our target words. We have applied all the pre-processing steps mentioned in Section 3.4 and also stemmed the words using the Porter stemmer (Porter 1980). Each word corpora had tweets varying from 17k tweets to 300k tweets depending on the frequency of the target word. The average number of tokens per message after pre-processing 9.04. Statistics of each target word are reported in Table 3.3 under ORIGINAL category.

Word	*Original*	*Expanded*
bar	180k	186k
charge	41k	43k
execution	28k	30k
field	72k	75k
figure	28k	29k
function	26k	27k
investigator	17k	19k
match	126k	133k
paper	210k	218k
post	299k	310k

Table 3.3: Number of tweets for each word in each background corpus

3.3.2 Expanded

After executing an initial set of experiments we wanted to investigate if adding more relevant context to tweets, that is, *expanding the documents*, would increase the performance of estimating the usage similarity. To answer this we created the corpus EXPANDED which is an expanded version of ORIGINAL. We chose the hashtag based context to expand the tweets. We selected medium frequency hashtags with an occurrence of $(10-35)$ in an hour and at least one of those occurrences had our target word in the noun category. We observed that around $3-7\%$ of tweets in ORIGINAL had a medium frequency hashtag per lemma basis and overall 3.7% in ORIGINAL corpus had tweets with medium frequency hashtags. Based on medium frequency hashtag heuristic we expanded the dataset and created 10 new versions of the corpus similar to ORIGINAL. EXPANDED had all the tweets from ORIGINAL + 40k *expanded tweets* in overall (Tweets which are formed after merging based on hashtag, these are longer than traditional tweet length). Each *expanded tweet* had words from at least 10 to 35 tweets which shared same hashtag irrespective of having the target word. The main reason why we targeted on medium frequency hashtags is because low frequency hashtags tend to be ad hoc and non thematic in nature whereas high-frequency tags are potentially too general to capture usage similarity. The average number of tokens per message in this corpus after pre-processing is 12.7 whereas the average number of tokens just over expanded messages is 105.

3.3.3 RandExpanded

We wanted to investigate whether expanding the documents by adding relevant context will improve the performance or *expanding the background collection* will also improve the performance. In order to check if an expanded background collection will improve the performance we expanded the ORIGINAL data by randomly adding tweets which satisfy all the pre-processing criteria. We have added the same number of tweets that were added in EXPANDED to efficiently compare both the results. Average number of tokens per message in this corpus after pre-processing over all documents is 9.05.

3.3.4 I-EN

Considering one of the the heuristics that we employed to sample the *Usim tweet* dataset being having enough valid English lexical tokens, we wanted to test if a model trained on standard English would be suitable to estimate the usage similarity over tweets. To test this we have used the English internet corpus (I-EN) (Sharoff 2006). *Usim Lexsub* dataset and the dataset for English Lexical Substitution task (McCarthy *et al.* 2007) were sampled from the I-EN corpus. Similar to the above models, we test this using word-based subcorpus. We consider all sentences from the I-EN corpus which contain our target words and then lemmatized using TreeTagger (Schmid 1994) those sentences in each sub corpora. Each sentence is considered as a document similar to above 3 corpora. None of the pre-processing steps mentioned in Section 3.4 is applied on this data apart from lemmatisation using TreeTagger. Average number of tokens per sentence in this corpus after pre-processing over all documents is 28.6.

3.4 Data Pre-Processing

In this section we describe the pre processing steps which we executed while creating all the background corpora except I-EN corpus.

Tokenization: Twitter messages contain many non-standard English tokens. For example URLs, hash tags i.e., '#' tag followed by a word (for example #football, #music etc.), user mentions '@' followed by a user-Id (@username) which is allowed to contain a few special characters and numbers, email id's, abbreviations and emoticons. It is difficult to retain the original content and structure of the Twitter messages if a general English tokenizer is used. We used a tokenizer built especially for Twitter messages using regular expressions. Using this tokenizer each hashtag, user mention, URL, emoticon, punctuation mark, email address are considered to be single tokens, thus retaining the original content.

$$goalllssss \xrightarrow{\text{regular expression}} goallss \xrightarrow{\text{dictionary lookup}} goals \xrightarrow{stemming} goal$$

Figure 3.5: An Example of lexical normalisation

Lexical Normalization: Social media texts are prone to noise due to their informal and conversational nature. The main reason why there are many noisy or ill-formed words in social media texts is due to activity factors (Gouws *et al.* 2011).Some standard formats of lexical transformations observed earlier are transliteration candidates (e.g, 'and' → '&', '2' → 'to','two' etc.), suffix transformations ('why' → 'y'), suffix transformations ('tomorrow' → 'tom'), character repetitions often with vowels ('good' → 'goooooood') etc (Gouws *et al.* 2011; Cook and Stevenson 2009; Han *et al.* 2012b). As our study is not targeted at lexical normalization we used the lexical normalization dictionary available from Han *et al.* (2012b) who have taken care of many of the above mentioned categories. Before using the dictionary we pre-processed the text based on regular expressions to eliminate obvious categories like replacing a repeated character sequence of length greater than 3 with a length 2 (e.g, 'hellooooooooooooo' $->$ 'helloo'). We have also stemmed the word using porter stemmer and replaced it if the stemmed version is present in dictionary. An example of complete transformation of word *goalllssss* is given in Figure 3.5

Part-of-speech tagging:In this study we are targeting the usage similarity of noun words. To filter the Twitter messages with target words as nouns we have used a Twitter specific POS tagger. Given the noisy text and conversational and informal nature of Twitter Part-of-Speech tagging itself is a difficult task on Twitter compared to general English text. As we are just interested in basic categorization of POS tags we wanted to use a tagger built on coarse-grained tagset as it attains high accuracy scores. We have used the CMU Twitter POS tagger (Owoputi *et al.* 2012) to tag the tweets which uses a coarse tagset and has shown higher tagging accuracy over existing systems. They have reported an accuracy score of 90% for tagging nouns. The percentage of tweets that got eliminated after noun filtering varied per each target word. The target word *figure* had most of its tweets eliminated showing only 11% of them tagged as noun followed by 22% for *charge* and 24% for *post.*

3.5 Experimental Methodology

Usage similarity is a new task and its applicability to social media hasn't been demonstrated as far as we are aware. In order to compare our results, we will present

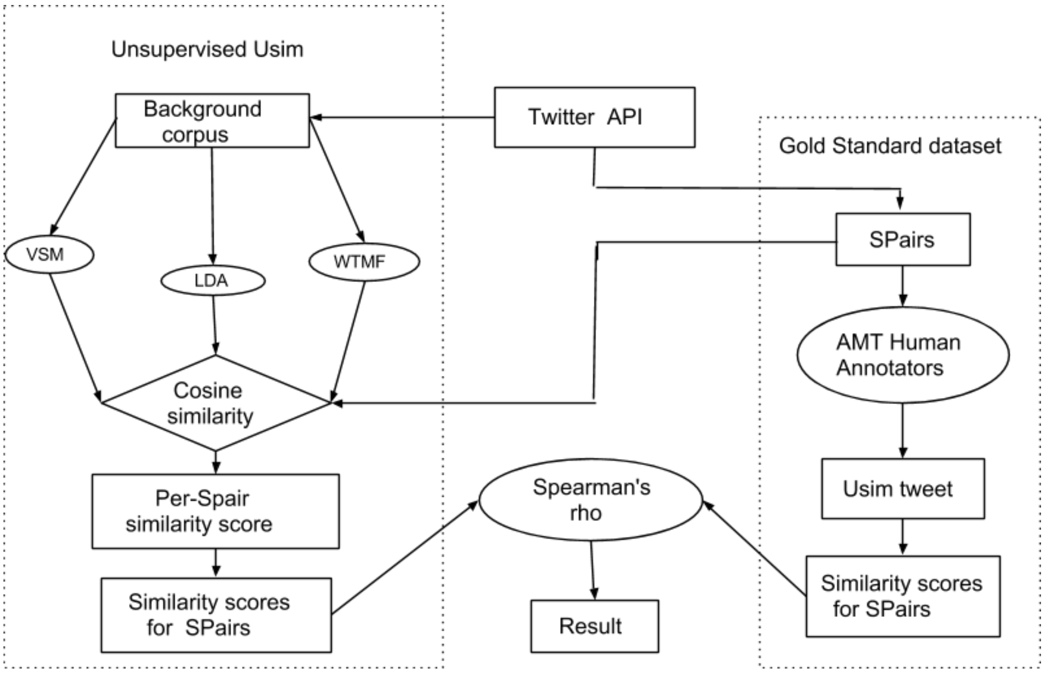


Figure 3.6: Overview of experiment methodology for automating usage similarity

baseline and benchmark approach. The baseline result is the output of second-order co-occurrence system. The benchmark system is WTMF system which has outperformed LDA and Latent Semantic Analysis (LSA) based systems in similar sentence similarity tasks. An overview of our methodology and evaluation is given in Figure 3.6. All our results are discussed in detail in Section 3.6. Our proposed system results exceeded both benchmark and baseline results over all four different corpora.

3.5.1 Baseline - Second order co-occurrence model

We use the cosine similarity measure of second-order co-occurrence vectors (Schütze 1998) of both the messages as our baseline method. A second-order co-occurrence vector of each Twitter message is built from the centroid summation of all the first-order co-occurrence vectors of the context words. For building the first-order vector of a word we consider all tweets which contain our target word categorized as noun in the background corpus. This customizes the first-order vector to the target word in noun category. We have used the most frequent 10000 words obtained after excluding stop words in the background corpus as our vector dimensions. Each dimension or context word in the first order vector is weighted based on mutual information (Resnik 1992). Second-order co-occurrence is used as the context representation to

Corpus	*Word*	*MAE*	*RMSE*	τ	*Pearson*	*Spearman*
ORIGINAL	bar	2.117	2.52	0.084	0.292	0.128
	charge	2.11	2.58	−0.066	0.083	−0.076
	execution	2.91	3.213	−0.117	-0.017	−0.178
	field	2.442	2.651	0.141	−0.017	0.206
	figure	1.801	2.042	0.061	0.109	0.094
	function	2.419	2.596	−0.039	0.29	−0.045
	investigator	4.379	4.39	0.047	0.232	0.065
	match	3.207	3.512	0.132	0.149	0.196
	paper	2.494	2.625	0.21	0.241	**0.304**
	post	3.164	3.436	0.215	0.255	**0.307**
	total	2.704	3.026	0.062	0.149	**0.093**
EXPANDED	bar	1.865	2.314	0.029	0.273	0.037
	charge	1.868	2.383	−0.027	0.093	−0.04
	execution	2.6	2.979	−0.132	−0.17	−0.208
	field	2.22	2.433	0.174	0.249	0.243
	figure	1.587	1.879	−0.076	−0.094	−0.121
	function	2.252	2.449	0.035	0.058	0.05
	investigator	4.161	4.177	0.052	0.193	0.075
	match	2.892	3.217	0.191	0.232	**0.29**
	paper	2.303	2.447	0.113	0.192	0.167
	post	2.713	3.023	0.106	0.262	0.157
	total	2.446	2.798	0.065	0.155	**0.096**
RANDEXPANDED	bar	2.116	2.52	0.095	0.295	0.14
	charge	2.108	2.577	−0.041	0.102	−0.051
	execution	2.914	3.214	−0.067	0.019	−0.106
	field	2.44	2.648	0.201	0.018	0.297
	figure	1.805	2.047	−0.009	0.04	0.002
	function	2.463	2.65	−0.005	0.104	−0.003
	investigator	4.389	4.4	0.054	0.232	0.062
	match	3.204	3.51	0.077	0.146	0.111
	paper	2.497	2.626	0.219	0.309	**0.32**
	post	3.169	3.446	0.178	0.171	0.241
	total	2.711	3.033	0.057	0.131	**0.085**
I-EN	bar	1.921	2.398	-0.022	0.005	-0.029
	charge	1.922	2.427	0.001	0.088	0.02
	execution	2.633	2.974	-0.012	-0.004	-0.015
	field	2.33	2.543	0.135	0.117	0.188
	figure	1.638	1.906	-0.002	0.025	0.008
	function	2.381	2.572	0.09	0.143	0.122
	investigator	4.224	4.238	0.057	0.033	0.083
	match	3.084	3.415	-0.081	-0.071	-0.09
	paper	2.341	2.5	-0.031	-0.073	-0.046
	post	3.148	3.421	0.24	0.337	**0.34**
	total	2.562	2.912	0.001	0.015	0.003

Table 3.4: Evaluation measures for each word, baseline method based on each background corpus. Spearman's ρ values that are significant at the 0.05 level are shown in **bold**.

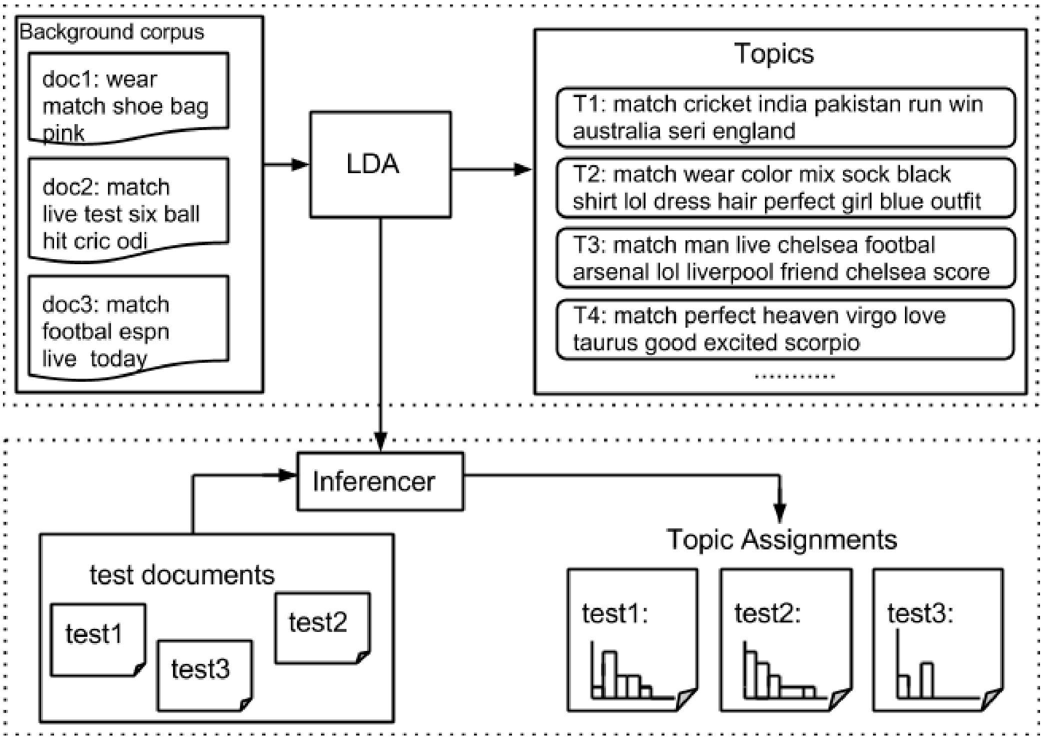


Figure 3.7: Overview of LDA based topic modeling approach

reduce the effects of data sparseness in the tweets which cannot be more than 140 codepoints in length.

3.5.2 Our approach - LDA

In this approach we tried to model the usage of the word in a short text by generating the topics from all words in the documents and then modeling each document as a distribution of these topics. An example illustrating the our approach for the target word *match* using topic modeling can be viewed in Figure 3.7. We try to represent the usage or meaning of word over latent topics that are learned using the word-based topic modeling approach. This is very much related to meaning representation and similarity estimation using latent senses learned from LDA proposed by Dinu and Lapata (2010) and inducing word senses from corpus based on LDA and its non-parametric variants Lau *et al.* (2012b). However, Dinu and Lapata (2010) try to estimate the meaning similarity of two different target words and we use a similar methodology for estimating usage similarity of the same word in two different texts.

execution $_{T1}$	**execution** $_{T2}$	**field** $_{T1}$	**field** $_{T2}$	**match** $_{T1}$	**match** $_{T2}$	**function** $_{T1}$	**function** $_{T2}$
marketing	job	play	love	watch	love	work	day
director	sale	today	day	play	perfect	applic	sleep
busi	hire	trip	good	good	shit	search	time
market	account	track	life	day	peopl	order	work
chief	manage	footbal	work	win	wear	websit	good
media	assist	good	god	time	girl	design	brain
social	develop	time	peopl	game	hell	add	love
news	market	team	time	team	follow	creat	feel
business	busi	basebal	happi	footbal	heaven	book	school
great	servic	year	feel	great	chamber	site	tire

Table 3.5: Top 10 topic words for 2 topics for lemmas *execution*, *field*, *match* and *function*.

Our experiments were performed using `Mallet` (McCallum 2002)[11], an open-source framework for LDA based topic modeling. Defining the optimal number of topics is a difficult task in LDA (Lau *et al.* 2012b). A non-parametric alternative of LDA, HDP (Teh *et al.* 2006) can be used to dynamically learn the number of topics as part of modeling. It is difficult to use HDP on our training dataset as we have a huge number of documents and HDP is very computationally intensive. Instead of this we used Mallet's hyper-parameter optimization functionality as part of training which allows the building of robust models with the flexibility of experimenting with a higher number of topics. Another advantage of using optimized parameters is it decreases the sensitivity of the model to the number of topics. It also generated more data-driven models with substantially less model complexity and computational cost than non-parametric models (Wallach *et al.* 2009).

Lui *et al.* (2012) have used a low number of topics ranging from $2-100$ whereas Dinu and Lapata (2010) used higher number of topics along with dynamic hyper-parameters to model the meaning of a word from context. We experimented on topics ranging from $2-500$ for each word based model. Lau *et al.* (2012b) used both HDP and LDA topic modeling for WSI tasks to learn latent senses from the corpus and experimented with lower number of topics.

Some example topics generated for few lemmas *execution*, *field*, *function* and *match* over EXPANDED corpus are shown in Table 3.5. Topics are selected from their individual best performing number of topics.

[11] http://mallet.cs.umass.edu/index.php

Corpus	*Word*	*d*	*MAE*	*RMSE*	τ	*Pearson*	*Spearman*
ORIGINAL	bar	8	2.791	3.133	0.118	0.109	0.158
	charge	8	2.43	2.909	0.071	0.219	0.117
	execution	8	1.995	2.409	0.1	0.17	0.14
	field	8	2.499	2.701	0.056	0.085	0.087
	figure	8	3.16	3.305	0.002	0.029	-0.003
	function	8	2.431	2.624	0.026	-0.064	0.04
	investigator	8	0.567	0.647	-0.02	-0.078	-0.026
	match	8	1.696	2.23	0.233	0.192	0.326
	paper	8	2.439	2.579	0.061	-0.0	0.085
	post	8	1.732	2.209	-0.084	-0.198	-0.123
	total	8	2.174	2.571	0.025	0.126	**0.036**
EXPANDED	bar	20	2.791	3.133	0.14	0.153	0.209
	charge	20	2.842	3.21	-0.02	-0.086	-0.033
	execution	20	2.028	2.44	-0.047	-0.096	-0.068
	field	20	2.504	2.706	0.157	0.183	0.229
	figure	20	3.16	3.305	0.082	0.219	0.125
	function	20	2.411	2.602	0.08	0.125	0.129
	investigator	20	0.569	0.649	-0.013	-0.103	-0.01
	match	20	1.696	2.23	0.133	0.14	0.201
	paper	20	2.439	2.579	0.18	0.112	**0.295**
	post	20	1.732	2.209	-0.166	-0.169	-0.246
	total	20	2.217	2.608	0.07	0.049	**0.105**
RANDEXPANDED	bar	5	2.791	3.133	-0.112	-0.188	-0.142
	charge	5	2.609	3.034	0.016	0.099	0.03
	execution	5	1.874	2.309	-0.111	-0.101	-0.173
	field	5	2.498	2.7	-0.004	0.036	-0.009
	figure	5	3.16	3.305	0.047	0.072	0.064
	function	5	2.375	2.573	0.049	-0.101	0.064
	investigator	5	0.569	0.649	0.081	0.095	0.114
	match	5	1.696	2.23	-0.093	-0.07	-0.125
	paper	5	2.439	2.579	-0.262	-0.215	-0.371
	post	5	1.732	2.209	-0.091	-0.1	-0.137
	total	5	2.174	2.571	0.063	0.085	**0.093**
I-EN	bar	10	2.791	3.133	-0.065	-0.155	-0.097
	charge	10	2.853	3.217	-0.135	0.011	-0.191
	execution	10	2.028	2.439	-0.104	-0.19	-0.153
	field	10	2.504	2.706	0.098	-0.003	0.146
	figure	10	3.16	3.305	-0.109	-0.019	-0.145
	function	10	2.448	2.638	0.08	0.178	0.131
	investigator	10	0.569	0.649	-0.174	-0.21	-0.235
	match	10	1.696	2.23	0.063	0.108	0.089
	paper	10	2.439	2.579	-0.057	-0.091	-0.084
	post	10	1.732	2.209	0.087	-0.128	0.124
	total	10	2.222	2.612	0.152	0.006	**0.216**

Table 3.6: Evaluation measures for each word, benchmark method based on each background corpus for optimal dimensions. Spearman's ρ values that are significant at the 0.05 level are shown in **bold**.

Lemma	IAA	Original		Expanded		RandExpanded		I-EN	
		Lemma	Global	Lemma	Global	Lemma	Global	Lemma	Global
		ρ (T)	ρ (T=8)	ρ (T)	ρ (T=5)	ρ (T)	ρ (T=20)	ρ (T)	ρ (T=5)
bar	0.75	0.39 (10)	**0.28**	**0.35 (50)**	0.1	0.34 (350)	0.18	**0.34 (100)**	0.13
charge	0.83	**0.27 (30)**	0.04	**0.33 (20)**	−0.08	**0.26 (10)**	0.19	**0.35 (10)**	0.04
execution	0.63	**0.43 (8)**	**0.43**	**0.58 (5)**	**0.58**	0.26 (20)	0.26	**0.32 (5)**	**0.32**
field	0.56	**0.46 (5)**	**0.33**	**0.53 (10)**	0.32	**0.41 (10)**	**0.39**	**0.48 (5)**	**0.48**
figure	0.63	0.24 (150)	0.06	0.24 (250)	0.14	0.23 (5)	−.022	**0.3 (2)**	0.16
function	0.5	**0.44 (8)**	**0.44**	**0.40 (10)**	**0.27**	**0.39 (10)**	**0.28**	**0.26 (10)**	0.20
investigator	0.62	**0.3 (30)**	0.05	**0.50 (5)**	**0.50**	**0.34 (8)**	0.18	**0.38 (8)**	0.23
match	0.74	**0.28 (5)**	0.26	**0.45 (5)**	**0.45**	**0.36 (50)**	0.16	**0.6 (20)**	0.47
paper	0.46	**0.29 (30)**	0.20	0.32 (30)	0.22	0.32 (100)	0.14	0.16 (350)	-0.01
post	0.63	0.1 (3)	−0.13	0.2 (30)	−0.01	0.15 (30)	0.01	**0.27 (450)**	0.11

Table 3.7: Spearman's ρ using LDA for all the background corpora

3.5.3 Benchmark - WTMF

We have used Weighted Textual Matrix Factorization (`WTMF`) as our benchmark model. WTMF addresses the data sparsity problem suffered by many latent variable models by predicting missing words based on the document context and adding it to the vector representation. This approach was shown to outperform LDA on the SemEval-2012 semantic textual similarity (STS) task (Agirre *et al.* 2012) by Guo and Diab (2012b). Similar to LDA and the baseline model the semantic space required for this model was built from word-based background tweets. We consider all the words that occur in the message or sentence as context. Similar to LDA, WTMF has various parameters including the number of dimensions (which could be related to our topics) and the missing weight parameter w_m which we set in the range {0.01, 0.05, 0.001, 0.005, 0.0005}. We have experimented with the dimensions in the range {2, 3, 5, 8, 10, 20, 30, 50, 80, 100} over tweets. Dimensions higher than 100 are computationally very expensive. Although we experimented using a different w_m parameter on all datasets we have observed that WTMF performs better when the missing weight is set as 0.0005. We also execute similar experiments on the *Usim Lexsub* dataset to check the performance of WTMF against LDA in general English text.

3.6 Experiments over *Usim Tweets*

We calibrate our method relative to a baseline and benchmark results. Baseline results are obtained from the second order co-occurrence model and benchmark results are obtained from the WTMF model. All the methods are evaluated using five

Model	ORIGINAL	EXPANDED	RANDEXPANDED	I-EN
Baseline	0.09	0.08	0.09	0.003
WTMF	0.03	0.10	0.09	0.22
LDA	**0.20**	**0.29**	**0.18**	**0.26**

Table 3.8: Spearman rank correlation (ρ) for each method based on each background corpus. The best result over each corpus is shown in **bold**.

criteria: the mean absolute error (MAE), the root mean squared error (RMSE), the Kendall rank correlation coefficient (τ), the Pearson correlation coefficient (Pearson) and the Spearman rank correlation coefficient (Spearman). All these measure were presented for baseline and benchmark approaches and for brevity we chose to present only Spearman's ρ for our approach and we consider Spearman is our main evaluation metric.

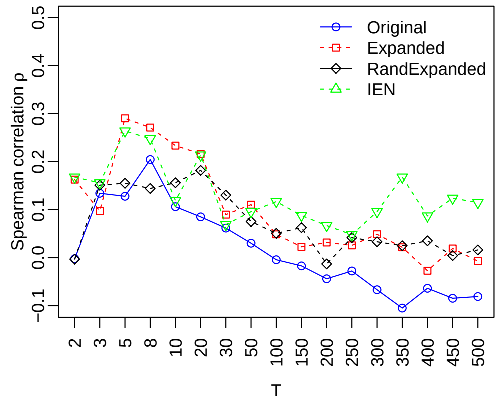
We present the results for each of the three models on all the four background corpora. Thus for each baseline, benchmark and proposed model we have 4 sets of results using ORIGINAL, EXPANDED, RANDEXPANDED and I-EN as background corpus used to build the model. Table 3.4 shows the baseline results for each word over *Usim-tweet* dataset and Table 3.6 shows results for the benchmark model. Results for our proposed approach and the inter-annotator agreement scores for each word in *Usim tweet* dataset are shown in Table 3.7. We also report the best scores for each background corpus using each approach in Table 3.8. This table shows that overall LDA out-performed both baseline and benchmark results.

In Figure 3.8a and Figure 3.8b we show the performance of our approach and benchmark over models learned with different number of topics (T) in LDA versus different number of dimensions (d) over each background corpus. In both approaches the EXPANDED corpus achieved better results than the ORIGINAL corpus. The results over the EXPANDED corpus are better and more consistent using both the approaches.

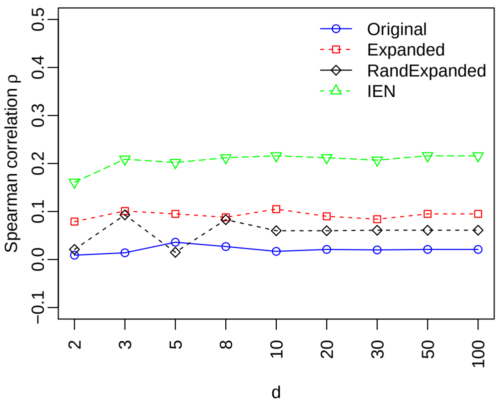
It is evident from the tables that our proposed approach out-performs both the baseline and benchmark methods This answers our RESEARCH QUESTION 1 that usage similarity can be estimated using an unsupervised approach. Results over the EXPANDED corpus are better when compared to ORIGINAL and RANDEXPANDED. This answers our RESEARCH QUESTION 2 that adding relevant text as context to the document does improve the performance in estimating the usage similarity.

		Lui-8	Lui-T		10-topic	T-topic		Baseline	2-WTMF	d-WTMF	
Lemma/POS	IAA	ρ	T	ρ	ρ	T	ρ	ρ	ρ	d	ρ
bar(n)	0.410	0.244	30	0.306	**0.467**	2	**0.467**	0.172	0.226	3	**0.358**
charge(n)	0.836	**0.394**	10	**0.667**	**0.52**	10	**0.52**	-0.074	0.165	2	0.165
charge(v)	0.658	**0.342**	30	**0.429**	0.232	50	**0.311**	0.222	-0.04	2	-0.04
check(v)	0.448	0.233	8	0.233	0.056	5	0.218	**0.396**	0.245	100	0.257
clear(v)	0.715	0.224	8	0.224	**0.339**	20	**0.473**	0.008	-0.025	20	0.149
draw(v)	0.570	0.192	10	**0.606**	**0.331**	3	**0.583**	-0.045	-0.233	2	0.233
dry(a)	0.563	**0.608**	5	**0.756**	**0.372**	10	**0.372**	-0.203	-0.077	5	0.267
execution(n)	0.813	0.174	30	0.277	0.266	8	**0.406**	-0.001	-0.066	100	-0.054
field(n)	0.267	0.118	3	**0.375**	**0.063**	2	**0.442**	0.219	0.179	50	**0.407**
figure(n)	0.554	0.158	3	**0.356**	**0.28**	200	**0.447**	-0.082	-0.068	100	-0.064
flat(a)	0.871	**0.444**	50	**0.684**	**0.718**	10	**0.718**	0.255	0.241	20	0.308
fresh(a)	0.260	-0.002	20	**0.408**	0.067	3	**0.352**	0.208	0.159	2	0.159
function(n)	0.121	0.234	30	0.292	0.049	8	0.087	**0.399**	-0.128	10	**0.357**
hard(r)	0.432	0.138	5	**0.309**	0.282	500	**0.454**	-0.217	-0.048	2	-0.048
heavy(a)	0.652	-0.014	5	0.261	0.291	30	**0.363**	0.235	0.223	10	0.245
investigator(n)	0.299	**0.364**	10	**0.583**	0.27	3	**0.5**	-0.105	0.076	100	0.115
light(a)	0.549	-0.078	20	0.180	0.133	8	0.232	0.135	-0.067	3	0.077
match(n)	0.694	-0.228	80	0.227	-0.238	500	**0.346**	0.77	0.152	5	0.155
order(v)	0.740	0.153	10	0.287	0.061	8	**0.234**	0.022	-0.098	3	0.077
paper	0.701	-0.026	3	**0.330**	**0.362**	150	**0.465**	0.316	0.094	2	0.094
poor(a)	0.537	0.210	10	**0.353**	0.148	2	0.211	0.025	-0.023	5	0.022
post(n)	0.719	**0.482**	8	**0.482**	0.183	2	**0.452**	**0**.248	-0.159	5	-0.121
put(v)	0.414	**0.544**	8	**0.544**	0.225	2	**0.526**	-0.298	-0.098	8	0.158
raw(a)	0.386	**0.387**	2	**0.392**	0.177	10	0.177	0.237	0.094	3	0.095
right(r)	0.707	**0.436**	8	**0.436**	**0.304**	8	**0.313**	0.023	-0.023	3	-0.044
rude(a)	0.669	**0.449**	8	**0.449**	**0.445**	10	**0.445**	0.22	-0.159	5	0.054
softly(r)	0.610	**0.604**	8	**0.604**	-0.238	30	0.244	0.106	0.055	3	0.162
solid(a)	0.603	**0.364**	3	**0.417**	0.211	100	0.296	-0.99	**0.484**	3	**0.52**
special(a)	0.438	0.140	30	**0.393**	0.236	5	**0.501**	-0.031	0.139	100	0.233
stiff(a)	0.386	0.289	8	0.289	0.005	2	**0.26**	-0.045	0.272	8	**0.433**
strong(a)	0.439	0.163	2	0.292	**0.45**	5	**0.527**	0.265	0.127	2	0.127
tap(v)	0.773	0.233	30	0.272	**0.376**	10	**0.376**	0.167	-0.062	2	0374
throw(v)	0.401	**0.334**	8	**0.334**	-0.073	50	**0.404**	0.083	0.202	3	0.222
work(v)	0.322	-0.063	80	0.132	0.06	5	0.262	0.223	0.201	100	0.235
adverb	0.585	**0.418**	8	**0.418**	**0.169**	450	**0.235**	0.092	-0.026	5	-0.019
verb	0.634	**0.268**	8	**0.268**	**0.271**	10	**0.271**	0.02	**0.313**	2	0.313
adjective	0.601	**0.171**	50	**0.219**	**0.23**	10	**0.23**	0.125	0.029	2	0.076
noun	0.687	**0.109**	3	**0.261**	0.091	2	**0.269**	**0.137**	**0.214**	100	**0.237**
overall	0.630	**0.202**	8	**0.202**	**0.205**	10	**0.205**	**0.098**	**0.152**	2	**0.152**

Table 3.9: Comparison of mean Spearman's ρ of inter-annotator agreement (IAA), Spearman's ρ for overall parameter combination of Lui using PAGE as background collection (Lui-8), and Spearman's ρ for the optimal number of topics for each lemma, using Lui PAGE as the background collection (Lui-T). Spearman's ρ for global optimum T using our method and best scores for each lemma with optimal setting of T for each topic. Spearman's ρ for global optimum d using WTMF and best scores for each lemma. ρ values significant at the 0.05 level are presented in bold.



(a) LDA versus topics (T)

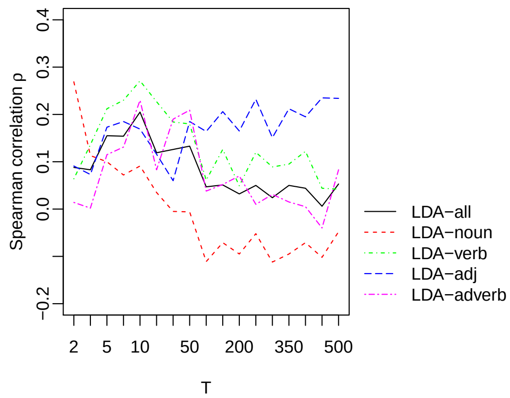


(b) WTMF versus dimensions (d)

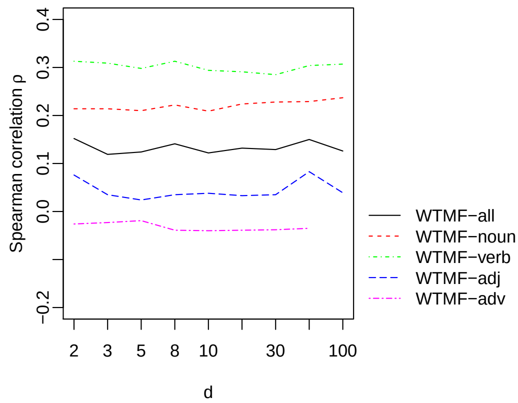
Figure 3.8: Spearman rank correlation (ρ) for LDA and WTMF for varying numbers of topics (T) or dimensions (d) using three different background corpora over *Usim tweets*

3.7 Experiments over *Usim Lexsub*

After observing some promising results on *Usim tweet* and also the lower performance of WTMF when compared to LDA on social media data we wanted to investigate if word-based topic models would perform well on general English text as well. In order to understand this we applied all our methodologies to the *Usim Lexsub* dataset using I-EN word-based background corpus. Earlier Lui *et al.* (2012) applied topic models to the *Usim Lexsub* dataset using same background corpus to study usage similarity. Their models were global topic models and used more context than our word-based topic models. They also used fixed hyper-parameters in all their



(a) LDA versus topics (T)



(b) WTMF versus dimensions (d)

Figure 3.9: Spearman rank correlation (ρ) for WTMF and LDA for varying numbers of dimensions (d) or topics (T) for overall and different POS categories over i-en corpus

experiments and hyper-parameters play a major role when the size of the document is small. We address this issue by using optimized paramaters and also show that our topics generated had low perplexity values. Table 3.9 reports and compares our word based topic models over I-EN corpus with best performing general topic models from the PAGE corpus (background corpus build using considering the whole page as document) experimented by Lui (2012). In this table we also report the optimal number of topics (in LDA) and optimal number of dimensions (in WTMF) for each lemma and the global optimal count. Although our analysis on *Usim tweet* was only limited to nouns, in this section we analyse its applicability over other POS categories on standard English text.

T_0: ⟨engineer, describe, water, energy, consultant, material, animal, expert, fire, service, identify, analysis, space, science, design, technology, failure demonstrate⟩
T_1: ⟨research, principal, study, project, include, university, health, report, subject, task, date, grant, information, programme, fund agency datum, human⟩
T_2: ⟨private, investigation, find, case, detective, evidence, work, state, federal, law, year, time, criminal, government, police, security, department⟩

Figure 3.10: Characteristic terms per topic for lemma investigator over *Usim LEX-SUB* dataset

T_0: ⟨obama, komen, ows, music, fund, state, peace, prize, penn, women, santorum⟩
T_2: ⟨jobs, law, work, health, media, announce, socialmedia, leveson, report, read, social, post, service, facebook, twitter⟩
T_1: ⟨private, crime, watch, scene, sex, video, fire, case, feder, murder, find, death, criminal, porn, csi, man, time⟩
T_3: ⟨cdnpolice, classicjokemonday, wear, vest, allig, fraud, elect, follow, tax, love, canada, demand, public, inquiry, lol, teamfollowback⟩
T_4: ⟨news, houston, whitney, egypt, uk, bahrain, syria, usa, death, anonymous, fbi, feb, call, bbc, iran, hack, world, apple⟩

Figure 3.11: Characteristic terms per topic for lemma investigator over *Usim tweets dataset*

In Figure 3.9a and Figure 3.9b we compare LDA versus WTMF over all POS categories (nouns, verbs, adverbs and adjectives). This analysis shows that when the LDA approach is used, nouns perform well with smaller numbers of topics whereas adjectives tend show higher performance at large numbers of topics. The performance of WTMF doesn't vary much with number of dimensions over all categories. However, the performance of WTMF varied for each POS category. Verbs performed the best followed by nouns, adjectives and adverbs. Verbs tend to perform well with WTMF whereasother categories showed better performance with LDA.

3.8 Discussion

For most of the lemmas we experimented on, the topics generated over standard English as background corpus were closely related to the sense definitions of the target word. On the other hand topics generated over the Twitter dataset represented the general topics that were covered in background corpus. This is evident from the example topics of the word *investigator* on *Usim Lexsub* and *Usim tweet* presented in Figure 3.10 and Figure 3.11, respectively. Topics T_1 and T_2 for the word *investigator* in *Usim Lexsub* represent the words related to senses *a scientist who devotes himself to doing research* and *a officer who investigates crimes*, respectively. For *Usim tweet* topics T_1, T_3, T_4 shows the words related to crime investigator and the news articles whereasT_0 shows words which are unrelated to any of the senses.

Comparing the performance of LDA and WTMF across *Usim Lexsub* and *Usim tweet* LDA performs similarly on both the datasets i.e., nouns show higher performance at topics ≤ 50 and as the number of topics increase performance decreased.

In Table 3.9 we show the results of all the three approaches on *Usim Lexsub* dataset and compare against inter annotator agreement score and the best performing results of Lui (2012).

Considering the fact that the Twitter messages sampled had enough lexical tokens we also tested the *Usim tweet* dataset on the models learned using I-EN word-based corpus. The best score observed over LDA was 0.264 which is little lower than our best score using EXPANDED corpus. This shows that expanding sentences or using PAGE level corpus could be a potential alternative background collection to experiment or investigate. However when PAGE level background collection was used to model over standard English and evaluated against *Usim Lexsub* this has not shown any better results according to Lui (2012). This might hold for Twitter messages as well as they have less common tokens compared to Standard English.

3.9 Topic Modeling on Semantic Textual Similarity Evaluation

Semantic Textual Similarity (STS) is the task of measuring the degree of semantic similarity between two texts (Agirre *et al.* 2012). STS is applicable to many NLP applications directly or indirectly. In text summarization to check if a sentence should be included in the summary (Aliguliyev 2009), in machine translation to evaluate systems (Kauchak and Barzilay 2006; Castillo and Estrella 2012), in question answering to check if two questions are similar enough so that answers from one question can be suggested to other (De Boni and Manandhar 2003; Jeon *et al.* 2005).

In usage similarity the task is to estimate the similarity of two different usages of a particular word where as in semantic textual similarity the task is to estimate the semantic relevance between two texts. An example illustrating usage similarity and semantic textual similarity is given in Figure 3.12.

As the semantic textual similarity task is very similar to the usage similarity

Usage similarity example: target word **paper**

s1: Someone is cutting a circle out of a pink sheet of **paper**.

s2: How often do you get to publish your work as a **paper**?

Semantic Textual Similarity: Estimate similarity of two short texts.

s1: Two U.S. soldiers shot, killed by Afghan soldier.

s2: Two British soldiers were attacked by Afghan militants.

Figure 3.12: Example describing the difference between Usim and semantic textual similarity

System	OnWN	FNWN	Headlines	SMT	Overall
SystemA	**0.648**	**0.358**	0.516	0.209	0.433
SystemB	**0.675**	**0.394**	0.593	0.256	**0.484**
Median	0.528	0.327	0.640	0.318	0.480
Baseline	0.283	0.215	0.540	0.286	0.364
Best-Score	0.843	0.581	0.783	0.403	0.618

Table 3.10: Pearsons ρ of topic modeling based systems, best run, the baseline set by the organizers, and median of all the systems submitted to the task, on each test dataset, and the micro-average over all test datasets. Best run represent system with best avg mean score. Our scores which are above median system are highlighted in **bold**

task, we applied the usage similarity topic model approach to calculate the semantic similarity of two sentences. Instead of using word based multiple sub-corpora we had a single corpus and modeled the general topics instead of word-based topics and inferred them on target sentences. We have created the background corpus from WordNet sense definitions, Wiktionary[12] sense definitions and all the sentences in the Brown Corpus. Each of them are considered as single document in the background corpus, in total we had 393k documents. For the evaluation, four different datasets OnWN, FNWN, SMT and Headlines were provided which were sampled from different domains. Each dataset has a pair of sentences for which semantic similarity must be quantified. OnWN has sense definitions sampled from OntoNotes versus WordNet, FNWN has sense definitions sampled from FrameNet against WordNet, Headlines has

[12]http://en.wiktionary.org/wiki/Wiktionary:Main Page

headlines mapped from different news sources and SMT is system translation mapped against human corrected translation.

Determining the optimal number of topics T is a difficult task as the test dataset is sampled from multiple domains and T may vary for each test dataset. This is clearly evident from the different number of optimal topics for each lemma in usage similarity over Twitter dataset and *Usim Lexsub* dataset (Section 3.8). Instead of choosing a single topic and quantifying the similarity score we have chosen 33 topics in the range $2, 3, 5, 8, 10, 50, 80, 100, 150, 200, ...1350$ and used all the similarity scores computed as features in building a regression model. We have chosen topics in a broad range as smaller number of topics have been show to perform poorly on sentence similarity tasks (Guo and Diab 2012b).

We conducted two different experiments based on two different samples of training datasets available. Our initial experiments were just based on a model learned from 2234 sentence pairs given as training data for STS 2012 task (Agirre *et al.* 2012) (TrainingA). Our next set of experiments were based on much larger dataset which used both test and training data available from STS 2012 task to learn the model. This dataset is referred as TrainingB and had 5342 sentence pairs.

We learned a ridge regression model based on the topic model features learned for two different training datasets. For each pair of sentences we had computed three similarity measures: cosine similarity, KL divergence and Jensen-Shannon divergence. Thus for each sentence pair we extract 99 features corresponding to the 3 similarity measures for each of the 33 topics chosen.

3.9.1 Results for STS 2013 task

In Table 3.10 we have reported the performance scores of our systems on the test data of STS 2013 task (Agirre *et al.* to appear). In this table we have also reported the baseline score set by task organizers, Median system score and best-score achieved on each of the datasets. Table 3.10 shows that ridge regression model built using topic model based features is useful in estimating semantic similarity of texts. Adding more training data increased the performance of the system. The system built using TrainingB dataset outperformed the median of 89 systems submitted to STS 2013 task. (Gella *et al.* 2013) have shown that topic modeling based features are useful in estimating similarity of sentences not sharing many common lexical tokens.

We have also executed experiments using extended feature set which include string similarity based features and information retrieval rank based features. When these additional features were added we saw an improvement in the scores and our best performing systems using TrainingA ranked 17 whereasour systems trained using

TrainingB dataset ranked 4th out of 89 systems submitted.

3.10 Summary

In this chapter we have presented the gold-standard dataset created for evaluating our unsupervised approach for estimating usage similarity. We have explained our experimental methodology to automatically estimate usage similarity and evaluate it against gold-standard datasets. We have carried our various experiments using different background collections and show that our proposed topic modeling based approach outperforms investigated baseline and benchmark approaches over all experiment settings investigated.

Apart from working on social media texts we also execute a similar set of experiments over English and show that our proposed approach not only works for Twitter messages it also out-performs both baseline and benchmark results over standard English text. We also give an overview of the difference between topics generated over Twitter messages and standard English. We also show that the benchmark model performs well on certain part of speech categories over standard English.

We also show that expanding Twitter messages using hash tags show significant improvement over the results. We have discussed our results in detail. We have also showed that the proposed topic modeling based approach could be used to address other tasks similar to usage similarity by evaluating our approach against the semantic textual similarity task 2013 dataset.

Chapter 4

Sense distribution in Social Media

In this chapter we explore the overall sense distributions on Twitter and compare them with sense distributions in standard English text. We execute two major analysis over sense distributions. Firstly, we verify if sense distributions on Twitter exhibit one predominant sense which is a strong tendency observed in standard English. We verify whether one predominant sense is observed on Twitter messages correlates with the same predominant senses over standard English. Second, we verify one-sense-per-discourse which is a strong tendency observed in standard English documents. Along similar lines we determine whether Twitter messages from a specific user over time confirm to such a tendency.

4.1 Sense distribution

The traditional way of investigating the sense distribution in a text is to manually assign word senses to the words in a context. For example consider the Twitter messages

(4.1) #knifeart carving the watermelon with a *knife*

(4.2) kidnapper used a *knife* to threaten her!

The word *knife* in the above examples are used in two different senses. A gold-standard sense annotation for message (4.1) with reference to word *knife* would be "#kniefeart carving the watermelon with *knife/TOOL*" whereas for the message (4.2) it would be "kidnapper used *knife/WEAPON* to threaten her". This would be the case if senses from a fine-grained sense inventory like WordNet is used for annotation. However, if the coarse-grained inventory like Macmillan dictionary is used for annotation in both the messages word *knife* would refer to "an object with a sharp blade" representing a single sense for both tool and weapon together. Ideally in sense tagging tasks multiple annotators are asked to tag each occurrence of a word with the most appropriate sense and the sense tagged by the majority of the annotators is given as the label in case of disagreements between annotators. According to (Krishnamurthy and Nicholls 2000) manual assignment of sense labels or the sense tagging task is considered a difficult task for human annotators.

4.1.1 Social Media

Sense distribution in social media was not explored earlier as far as we are aware. Considering the characteristics like informal representation and dynamic user-generated text in social media, it is difficult to study sense distribution in social media. Short text and informal representation increase the difficulty in interpreting the social media texts. Using coarse-grained senses to study sense distribution was known to alleviate difficulty in sense tagging tasks on English (Hovy *et al.* 2006; Erk *et al.* 2009), as well as being shown to raise in ITA scores to 90%. In this study we follow the rule of coarse-grained senses and study the sense distribution in social media text.

We execute the sense tagging task on Social media data using a coarse-grained sense inventory. We also compare it with the sense distribution of similar senses on standard English sentences sampled from ukWac corpus (Ferraresi *et al.* 2008). We have opted to sample sentences from ukWac corpus than other available corpora as its relatively new and is built by crawling the web where most of the documents are user generated and have certain similar features similar to Twitter. Interestingly ukWac sentences are compared to Twitter messages on a stereotypical gender-actions-based study by Herdağdelen and Baroni (2011). This shows the evidence that ukWac has earlier been used as standard English benchmark to execute Twitter messages versus standard English comparison.

4.1.2 Sense Inventory

A sense inventory partitions the range of meanings of a word into its senses (Navigli 2009). According to Navigli (2009) senses can be listed by splitting (fine-grained) or lumping sense distinctions (coarse-grained). We have chosen a coarse-grained sense inventory to execute our experiments. Sense tagging task with fine-grained senses will increase the possibility of having multiple senses applied in the cases where there are many relevant senses. One example of difference between coarse-grained and fine-grained sense inventory can be observed over the word *post*. For example the meaning of *post* referring to letters, postal service and the process of collecting/delivering letters are combined and given as single sense in coarse-grained sense inventory Macmillan dictionary. Whereas in a fine-grained sense inventory like WordNet they are given as 3 different senses. Thus using a coarse-grained sense inventory makes the task easier to execute. We have manually verified senses in OntoNotes (Hovy *et al.* 2006) and Macmillan, two coarse-grained sense inventories. We observed that senses in Macmillan are the latest (frequently updated) and reflected Twitter

Word	Word	Word	Word	Word
band	bar	case	charge	deal
degree	field	form	function	issue
job	light	match	panel	paper
position	post	rule	sign	track

Table 4.1: 20 Target words studied in sense distribution task

specific usages better compared to OntoNotes. One example is the for the word *post* which has a new meaning of " a message sent over the Internet to a newsgroup" which is mentioned in Macmillan dictionary whereas OntoNotes do not mention that newly evolved usage of word *post*. We also observed that for a few of our selected target words mentioned in Table 4.1 do not have noun sense definition in OntoNotes sense inventory. For these two reasons we chose Macmillan dictionary as sense inventory.

4.1.3 Target Words

In this study we study 20 target words in the noun category mentioned in Table 4.1. We have extended our target word from words in *Usim-tweets* Section 3.1. We have eliminated the words *investigator*, *execution* which had far less sense definitions in our chosen dictionary that is not exhibiting multiple meanings. We have also eliminated *figure* as we could not get enough users to pass all our heuristics mentioned in Section 4.2.1 for this word. The newly added target words are chosen primarily on three characteristics first: occurrence count in Twitter corpus; second: number of senses described in WordNet; third: number of senses described in Macmillan. Based on the above three characteristics we have selected our final target words which fall in the range of frequent to mid frequent over Twitter and which had at least 3 Macmillan senses and $7 - 30$ WordNet senses. Thus making sure that selected target words had good diversity among coarse-grained versus fine-grained senses and occurrence of frequent versus less frequent over Twitter.

4.1.4 Multiple Senses

Most of the well known sense annotation tasks have always allowed annotators to assign multiple senses to a single occurrence of a word (Mihalcea 1998; Mihalcea *et al.* 2004). However, the percentage of annotations that received multiple senses over the corpus has always varied. For example annotations executed using fine-grained corpus on SemCor has only 0.3% annotated as possessing multiple senses (Mihalcea 1998; Erk *et al.* 2009). Whereas in the SensEval-3 English lexical task corpus 8% of the corpus was tagged as possessing multiple senses (Mihalcea *et al.* 2004). One major difference between these two tagged corpora is the number of words covered is different. In SemCor they had 254 target words in noun, verb, adjective and adverb whereas SenseEval-3 had 57 target words in noun, verb and adjective categories. In both the sense tagging tasks sense definitions from WordNet 1.7.1 (Miller 1995) are used for noun category. In this study we analyse multiple senses observed with a coarse-grained dictionary on Twitter and compare it with ukWac sentences.

4.2 Annotation Settings

The sense tagging task is executed using the Amazon Mechanical Turk application which was earlier used in collecting our *Usim-tweets* task in Chapter 3. For each target word we extract a set of sentences/messages from the ukWac corpus and the Twitter steaming API. The sentences/messages were extracted in two samples:

- To study the overall sense distribution across Twitter messages and verify if it exhibits the tendency of one predominant sense. This analysis would address our Research Question 3 and Research Question 5 (Section 1.1)

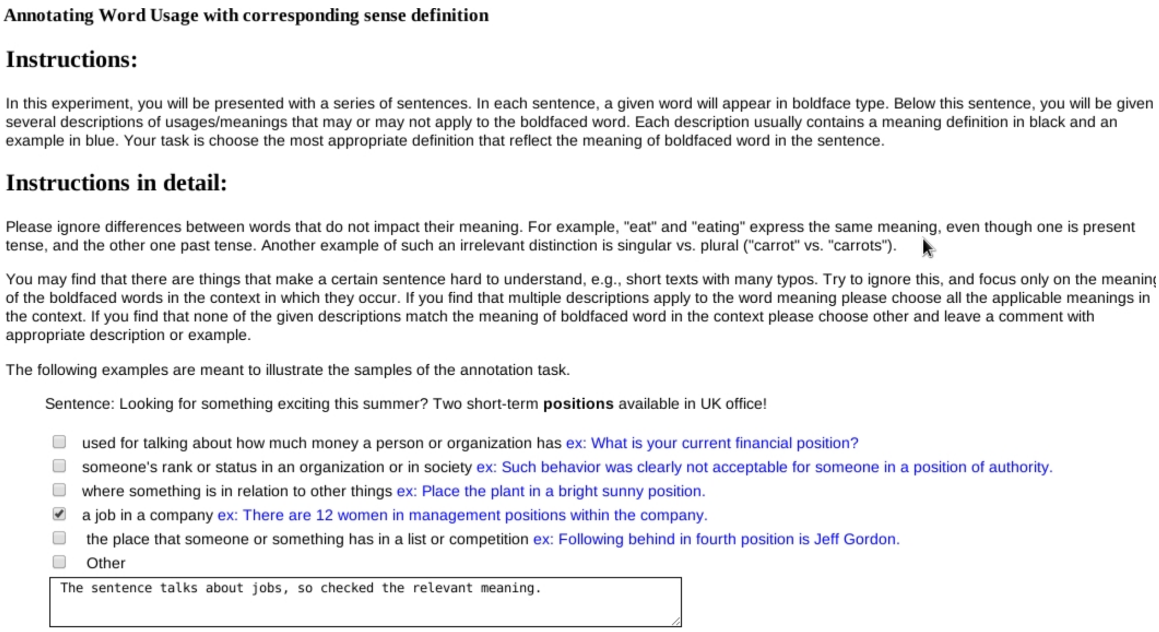
- To study the one-sense-per-discourse heuristic over Twitter messages from the same user over a time frame which would address our Research Question 4 (Section 1.1).

Each HIT had 5 sentences or message pairs having one of our target words along with respective sense definitions and was annotated by 5 turkers. One of these 5 sentences was taken from examples mentioned in dictionary sense definition. This sentence is served as gold-set in the annotation task which we later used for quality control checks. More details about this would be found in Section 4.2.3.

4.2.1 Data sampling

We have filtered all the English Tweets that were crawled through the steaming API[13] over the time period January 03^{rd} 2012 to February 29^{th} 2012 using `langid.py` off-the-shelf language identification tool (Lui 2012). We filtered all the tweets which contain one of the target words in noun category using a Twitter specific POS tagger (Owoputi *et al.* 2012). These two steps were carried out according to the data processing steps mentioned in Section 3.4. Then we sampled tweets in two categories

[13]https://dev.twitter.com/docs/streaming-apis



Annotating Word Usage with corresponding sense definition

Instructions:

In this experiment, you will be presented with a series of sentences. In each sentence, a given word will appear in boldface type. Below this sentence, you will be given several descriptions of usages/meanings that may or may not apply to the boldfaced word. Each description usually contains a meaning definition in black and an example in blue. Your task is choose the most appropriate definition that reflect the meaning of boldfaced word in the sentence.

Instructions in detail:

Please ignore differences between words that do not impact their meaning. For example, "eat" and "eating" express the same meaning, even though one is present tense, and the other one past tense. Another example of such an irrelevant distinction is singular vs. plural ("carrot" vs. "carrots").

You may find that there are things that make a certain sentence hard to understand, e.g., short texts with many typos. Try to ignore this, and focus only on the meaning of the boldfaced words in the context in which they occur. If you find that multiple descriptions apply to the word meaning please choose all the applicable meanings in the context. If you find that none of the given descriptions match the meaning of boldfaced word in the context please choose other and leave a comment with appropriate description or example.

The following examples are meant to illustrate the samples of the annotation task.

Sentence: Looking for something exciting this summer? Two short-term **positions** available in UK office!

- [ ] used for talking about how much money a person or organization has ex: What is your current financial position?
- [ ] someone's rank or status in an organization or in society ex: Such behavior was clearly not acceptable for someone in a position of authority.
- [ ] where something is in relation to other things ex: Place the plant in a bright sunny position.
- [x] a job in a company ex: There are 12 women in management positions within the company.
- [ ] the place that someone or something has in a list or competition ex: Following behind in fourth position is Jeff Gordon.
- [ ] Other

The sentence talks about jobs, so checked the relevant meaning.

Figure 4.1: Screenshot of label the word meaning annotation task for the word *position*

1. We grouped all the tweets based on the user who tweeted them. And selected the users who have tweeted at least 5 times over two week period with our target word as noun. We then randomly sampled 20 users who satisfied all these categories.

   We wanted our targeted users to be real people rather than automatic accounts or agencies posting about advertisements. In many social media applications spam accounts are a real big concern and this is not a trivial task (Wang 2010). Instead of going into details of spam detection we observed a few patterns that were observed in spam / advertisement accounts. We filtered the users based on our heuristics of using target word at least 5 times over two week period and not containing the patterns of spam accounts.

   Some patterns observed in spam accounts are:

   (a) High word similarity across tweets in a two week time frame. We filtered the tweets based on similarity with every other tweet from that user and added it to a set only if the similarity is below 0.7 using word-based cosine similarity. If the number of tweets in the filtered set was less than one-third of the original number of tweets from the user then we discarded the user.

(b) Repetition of same text with change in username mentions and URLs and retweets. (Anonymisation of URL and usernames and would easily detect these kind of users).

(c) Very high number of tweets. We have only targeted the user accounts who have mentioned target word in the noun category in less than 50 tweets over a two week period.

2. We randomly sampled 100 tweets for each target word from 100 different user accounts.

We filtered all the documents in ukWac corpus containing our target word as noun at least 5 times. For each target word we have randomly sampled 20 of those documents. We have also created another sample by randomly selecting 100 sentences for each target word from different documents to study the overall sense distribution across ukWac corpus.

4.2.2 Sample Annotation

The randomly selected examples were presented to the turkers, who were asked to select the most appropriate sense for each target word in each sentence/message. Along with these randomly sampled messages we also collected some gold-set example sentences and annotations based out of the Macmillan dictionary sense definitions and examples. These were included along with the sentences from ukWac or messages from Twitter. We later used this Macmillan gold-set examples to detect spam annotations from our Turkers.

In each HIT we had 5 sentences/messages including one gold-set from the example sentence. Each message is accompanied by a target word in bold case letters and senses definitions from the Macmillan dictionary. Turkers were free to select multiple sense labels where applicable. We have mentioned the applicability of multiple senses and have assigned check-boxes for annotation showing no bias towards single-single assignment. Along with sense definitions from the Macmillan dictionary we have also provided an "other" option which authors were free to choose if they judge that none of the senses mentioned are applicable. We have also provided an optional text-box where users can add their views on why a particular option is chosen especially in the case of "other". A sample annotation example is given in Figure 4.1 .

4.2.3 Spam detection and Quality Control

In this section we present the heuristics used to filter the spam annotations while collecting gold-standard datasets. Datasets collected using crowd-sourcing are prone to have spam annotations (Kazai and Milic-Frayling 2009; Yuen *et al.* 2011; Vuurens *et al.* 2011). We have used the gold-sets in the annotated data to filter out spam detections. A gold-set is a sample of data for which the gold-standard annotations are already known and used to filter spam annotations in crowd-sourcing methodologies. This methodology was earlier experimented and proven to be successful in filtering spam annotations (Bentivogli *et al.* 2011; Vuurens *et al.* 2011). We have filtered out the spam annotations by using the following heuristics

1. Accepting all HITs from turkers whose accuracy on gold-sets was above 80%.
2. Rejecting all HITs from turkers whose accuracy on gold-sets was below 20%
3. Accepting the HITs which had correct gold-set mappings or at least 2 out of other 4 (non gold-set) annotations had common answers with other turkers who annotated the gold-set correctly. Rejecting the HITs which do not qualify for this heuristic. This filtering technique was applied for the turkers whose accuracy was in the range of 20-80%.

4.3 Analysis

We divided our datastet into 4 different samples and executed the sense tagging on each sample.

- Sense tagging on the random sample of Twitter messages.
- Sense tagging on the random sample of ukWac sentences.
- Sense tagging on the user sample of Twitter messages.
- Sense tagging on the document based sample of ukWac sentences.

For each of the above mentioned tasks we had a different number of turkers. We had turkers who rated from $1-500$ annotations in each task. Detailed analysis of sense agreements and distributions found are given in subsections below. In all the tasks we have only considered the annotations which satisfied our quality control steps and we did not consider the rejected annotations to do the analysis below.

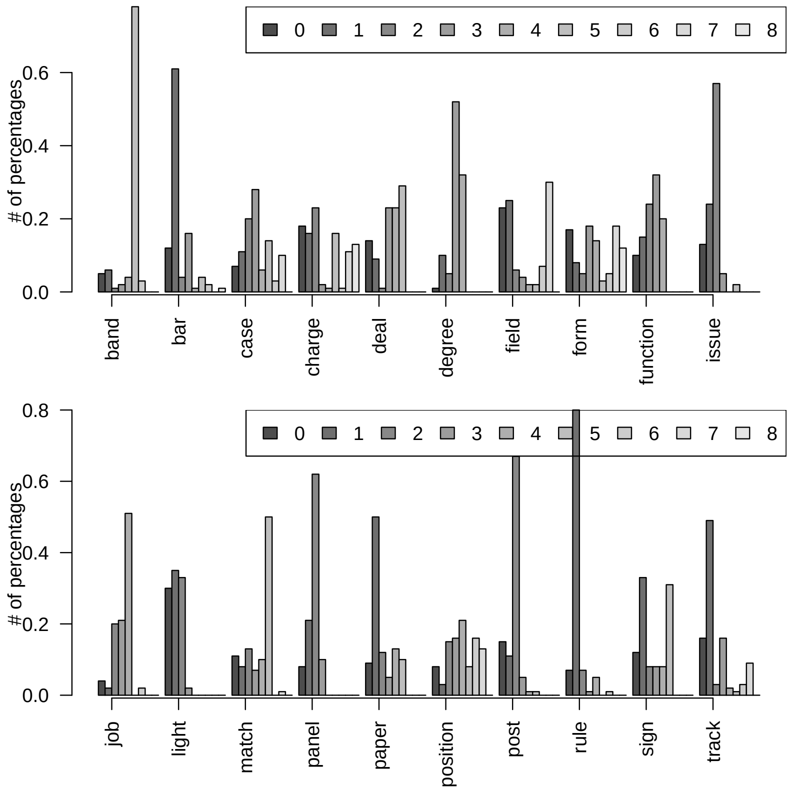


Figure 4.2: Sense distribution per word over randomly sampled data from Twitter

4.3.1 Sense distribution over Twitter random sample

In this section we present the overall sense distribution observed over randomly sampled tweets from Twitter. In total we had 83 turkers who participated in this annotation task out of which 46 had accuracy percentages of 80 and above, 33 had accuracy of 50 - 80%, only 1 had 20 - 50% and 3 of them who had less than 20% according to our spam detection heuristics. Overall sense distribution for each lemma is shown in Figure 4.2. For this dataset the overall inter-annotator agreement with weighted Fliess kappa is 0.466.

4.3.2 Sense distribution over ukWac random sample

In this section we present the overall sense distribution observed over randomly sampled tweets from the ukWac corpus. In total we had 86 who participated in this annotation task out of which 57 had accuracy percentages of 80 and above, 21 had accuracy of 50 - 80%, only 3 had 20 - 50% and 5 who had less than 20% according

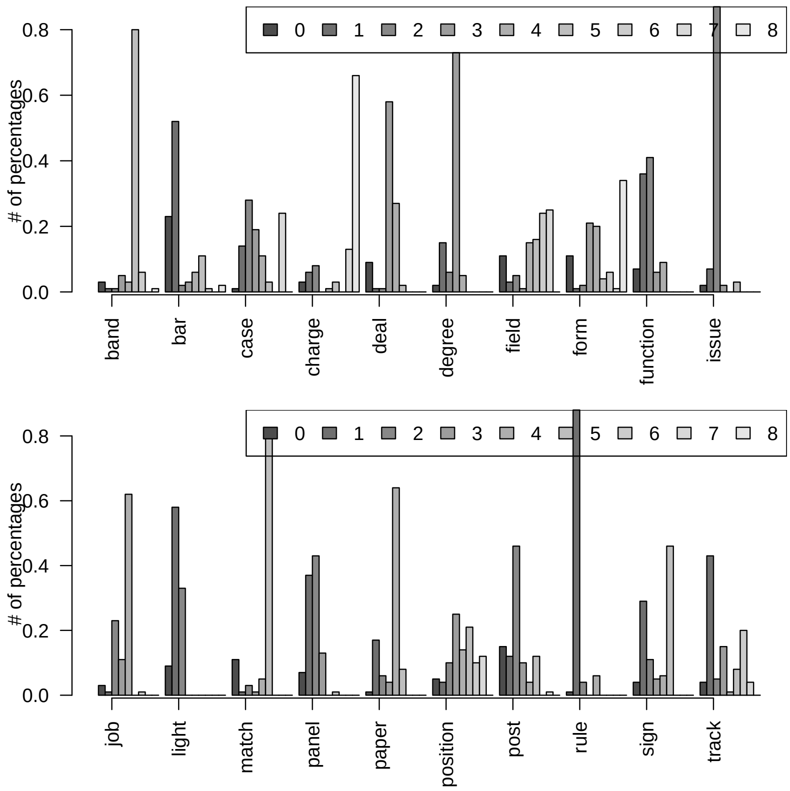


Figure 4.3: Sense distribution per word over randomly sampled data from ukWac corpus

to our spam detection heuristics. Overall sense distribution for each lemma is shown in Figure 4.3. Overall inter-annotator agreement with weighted Fliess kappa is 0.637 for this dataset and it is observed to be much higher than the measure observed over random sample form Twitter.

4.3.3 Analysis of Twitter/ukWac random sample

In this section we present the overall sense distribution across Twitter and ukWac. As mentioned in earlier sections we compare the sense distribution of 20 lemmas with 100 Twitter messages for each lemma. Figure 4.2 and Figure 4.3 shows the overall sense distribution per lemma over its senses on Twitter messages and ukWac sentences respectively. We have observed that sense distributions varied over both datasets and to obtain a stronger evidence we provide the entropy difference and Jensen Shannon divergence for the sense distribution of both datasets.

Word	JS	Entropy diff.	Word	JS	Entropy diff.
band	0.017	0.08	bar	0.061	−0.161
case	0.072	0.208	charge	0.193	0.754
deal	0.137	0.478	degree	0.078	0.237
field	0.157	−0.154	form	0.127	0.479
function	0.085	0.179	issue	0.072	0.639
job	0.017	0.212	light	0.052	0.267
match	0.075	0.764	panel	0.028	-0.163
paper	0.152	0.355	position	0.039	0.076
post	0.051	−0.516	rule	0.016	0.292
sign	0.024	0.156	track	0.076	-0.096

Table 4.2: Jensen-Shannon divergence and Entropy difference of sense distributions (Random sample Twitter subtracts random sample ukWac) for each word in two datasets over all sentences

Jensen Shannon (JS) divergence (Equation 4.4) is a popular method used to measure the distance or dissimilarity between two probability distributions p and q (p: probability distribution of senses for a word in random sample of Twitter, q: probability distribution of senses for a word in random sample of ukWac) and is defined as the average of the Kullback-Leibler divergence (Equation 4.3) of each of two distributions to their average distribution (Lapata *et al.* 2001). Kullback-Leibler divergence is defined as an information-theoretic non-symmetric measure of the difference between two probability distributions p and q. It measures the divergence of q from p and $KL(v||m)$ is the measure of information lost when q is used to approximate p. "Entropy is the the measure of uncertainty in a random variable"[14]. In this case we calculate the entropy difference (Equation 4.5) of two probability distributions which is the difference of their average unpredictability in a random variable and it is equivalent to the difference in its information content. The value of JS divergence and absolute value of entropy difference show the dissimilarity in probability distributions p and q. They are equal to zero only if both the distributions are identical.

$$KL(p||q) = \sum_{i=1}^{n} p_i log_2(p_i/q_i) \qquad \text{where n=no.of senses of the target word} \tag{4.3}$$

$$JS(p,q) = KL(p||m) + KL(q||m) \qquad \text{where m= } 1/2x(p+q) \tag{4.4}$$

[14]http://en.wikipedia.org/wiki/Entropy_(information_theory)

Lemma	Twitter	Sense Definition	ukWac	Sense Definition
band	78%	a small group of musicians who play popular music such as jazz or rock	80%	a small group of musicians who play popular music such as jazz or rock
bar	62.4%	a place where you go to buy and drink alcoholic drinks	53.6%	a place where you go to buy and drink alcoholic drinks
case	28.6%	a situation or set of conditions, especially one involving a particular person or thing	27.7%	an example or instance of something
charge	22.2%	the amount or type of electrical force that something has	66.7%	an amount of money that you have to pay, especially when you visit a place or when someone does something for you
deal	28.4%	what is happening or going to happen	58.9%	a formal agreement, especially in business or politics
degree	50.6%	a qualification that you get after completing a course at a college or university	72.1%	a qualification that you get after completing a course at a college or university
field	29.8%	an area of land used for keeping animals or growing food	25.3%	an area of land used for keeping animals or growing food
form	19.7%	the particular way in which something appears or exists	39.2%	an official document that has spaces where you can put in information
function	31.7%	a social event such as a party, especially one for a large number of people	41.8%	the job that something is designed to do
issue	58%	a subject that people discuss or argue about, especially relating to society, politics, etc.	88.6%	a subject that people discuss or argue about, especially relating to society, politics, etc.
job	49.4%	work that you do regularly to earn money	61.3%	work that you do regularly to earn money
light	34.4%	brightness from the sun or from a light, which allows you to see things	58.3%	brightness from the sun or from a light, which allows you to see things
match	48.9%	in tennis, a competition consisting of a specific number of sets	79.1%	in tennis, a competition consisting of a specific number of sets
panel	62.4%	a group of well-known people who discuss subjects on television or radio programs	42.4%	a group of well-known people who discuss subjects on television or radio programs
paper	49.8%	the thin flat substance that you use for writing on or wrapping things in	63.8%	a piece of writing or a talk on an academic subject
position	21.6%	someone's rank or status in an organization or in society	25.6%	where something is in relation to other things
post	68.3%	a posting , a message sent over the Internet to a newsgroup, etc.	47.2%	a posting , a message sent over the Internet to a newsgroup, etc.
rule	79.8%	a statement explaining what someone can or cannot do in a particular system, game, or situation	88.4%	a statement explaining what someone can or cannot do in a particular system, game, or situation
sign	34.4%	a piece of evidence that something is happening or that something exists	45.1%	a flat object with words or pictures on it, put in a public place to provide information or advertise something
track	48.8%	a song or piece of music that is recorded on a CD, tape , or record	43.8%	a song or piece of music that is recorded on a CD, tape , or record

Table 4.3: Predominant sense labels and percentage statistics over Twitter and ukWac for all the target lemmas.

$$ED(p||q) = \sum_{i=1}^{n} p_i log_2(p_i) - \sum_{i=1}^{n} q_i log_2(q_i) \qquad \text{where n= no. of senses of the target word} \tag{4.5}$$

In Table 4.2 we present the Jensen Shannon divergence and entropy difference and for each lemma across two datasets. Lemmas *band* and *track* had the least difference in the sense distribution across both the datasets whereas lemmas match, charge, issue, post etc. showed high difference.

The highest percentage of predominant sense was observed for lemmas *rule, band, post, bar, panel, issue, degree* over Twitter and for lemmas *issue, rule, band, match, degree, charge, job*. Lemmas *rule, band, issue, degree* topped the predominant list in both the datasets. In Table 4.3 we show the frequent label percentage observed for both datasets. We have observed that 12 out of 20 lemmas have the same frequent label or predominant sense across both datasets. These differences in distribution show that the usages of words in Twitter are different when compared to standard English sentences sampled from ukWac corpus. The highest difference in frequent label percentage was observed for the lemma *charge*. We have manually verified this case and found that few of the sampled Twitter messages contained the phrase **in charge** and that led to annotators assigning the sense label as *Other*.

4.3.4 Sense distribution across users - Twitter

In this section we intend to analyse the one sense per user phenomena over Twitter users who use at least one of the target lemmas 5 times in a 2 week period. In total we had 73 turkers who participated in this annotation task out of which 46 had accuracy percentages of 80 and above, 26 had accuracy of 50 - 80%, only 1 had 20 - 50% and 10 of them who had less than 20% according to our spam detection heuristics. We observed that the number of users who showed one sense phenomena varied from 7/20 (for lemma *form*)to 20/20 (for lemma *degree*). We observed that overall 65% of the users showed one sense phenomena over Twitter at 80% of agreement at each user level and 68% of agreement at sentence level annotations. Overall inter annotator agreement with weighted Fliess kappa was 0.705 for this annotation task which is much higher than the value observed for sense assigning task over random sample from Twitter.

4.3.5 Sense distribution across documents - ukWac

In this section we intend to analyse one-sense-per-discourse / document phenomena over ukWac documents which had at least one of the target lemmas mentioned 5 times. In total we had 136 turkers who participated in this annotation task out of

	No. of words	Sentence Pairs	Agreed	Percentage
(Gale *et al.* 1992)	9	54	51	94%
User Twitter	20	2668	2544	95.35%
Document ukWac	20	2511	2366	94.22%

Table 4.4: Statistics for sentences pairs analysed for one-sense-per-discourse phenomena

which 70 had an accuracy percentages of 80 and above, 50 had accuracy of 50 - 80%, only 4 had 20 - 50% and 12 who had less than 20% according to our spam detection heuristics. We observed that the number of documents which showed one sense phenomena varied from 1/20 (for lemma *case*)to 20/20 (for lemma *band*). We observed that overall 63% of the documents showed one-sense-per-discourse phenomena over ukWac at 80% of agreement at each document level and 68% of agreement at sentence level annotations. Overall inter annotator agreement with weighted Fliess kappa is 0.641 for this annotation task which is similar to the random sample over ukWac and lower when compared to user level sample over Twitter.

4.3.6 Analysis of Twitter users /ukWac documents

From the above mentioned Section 4.3.4 and Section 4.3.5 we see that around 65% of users and 62% of documents showed one-sense-per-discourse. This shows the analysis at document/user level whereas when we consider sentence pairs taken from the same document/user similar to Gale *et al.* (1992) our data showed similar results. On the user sample Twitter sentence pairs showed 95.35% of having the same sense whereas on ukWac documents showed 94.32% of having the same sense. The sample which we observed is much bigger when compared to the sample of Gale *et al.* (1992) whose study is based on 9 ambiguous words and 54 pairs of sentences. Detailed analysis of number of pairs of sentences is mentioned in Table 4.4. In both user level sentence pairs and document level sentence pairs sentences which had high sense label agreements from annotators were considered. That is, all the sentences or messages which had confident sense labels are analysed. The results below show strong evidence that messages from the same user could be considered as a single document.

A sample of cases for both Twitter users and document ukWac which violated one sense per document/user are:

case: an example or instance of something/a situation or set of conditions

κ	**Interpretation**	Random Twitter	Random ukWac	User Twitter	Document ukWac
≤ 0	Poor agreement	1	0	0	0
$0.01 - 0.20$	Slight agreement	2	0	1	1
$0.21 - 0.40$	Fair agreement	13	7	4	5
$0.41 - 0.60$	Moderate agreement	3	9	8	10
$0.61 - 0.80$	Substantial agreement	1	4	5	4
$0.81 - 1.00$	Almost perfect agreement	0	0	2	0
total	-	20	20	20	20

Table 4.5: Inter-annotator agreement over lemma level that fall under κ interpretation over each dataset.

	Random Twitter	Random ukWac	User Twitter	Document ukWac
κ	0.466	0.637	0.705	0.641

Table 4.6: Inter-annotator agreement for overall datasets

function: brain function/ multi function guitar

light: brightness / electrical equipment that produces brightness

match: an attractive combination with something / a sports match

paper: examination paper / academic paper

deal: a formal agreement / an informal deal

track: racing track / rail track

In most of the cases they are related senses for example for the lemmas *light*, *paper* etc. . However, with the lemma *match* it is also seen in the non related senses are also mentioned by the same user or exist in same ukWac document.

4.3.7 Inter annotator agreements

We have calculated inter-annotator agreement (ITA) for each lemma and for overall annotations over each dataset. We have used Fliess kappa to measure the inter-annotator agreement as the annotations are categorical. Highest ITA is observed for the dataset user level sample from Twitter followed by document level sample from ukWac followed by the random sample from ukWac followed by the random sample from Twitter. Exact ITA scores over each dataset can be found in Table 4.6. We have also computed lemma level Fliess kappa score and the number of lemmas which

Data sample	Highest Agreement		Lowest Agreement	
	Lemma	*kappa*	Lemma	*kappa*
Random Twitter	degree	0.71	field	−0.21
Random ukWac	degree	0.75	function	0.29
User Twitter	degree	0.87	deal	0.13
Document ukWac	degree	0.69	case	0.08

Table 4.7: Inter-annotator agreement at lemma-level for all data samples

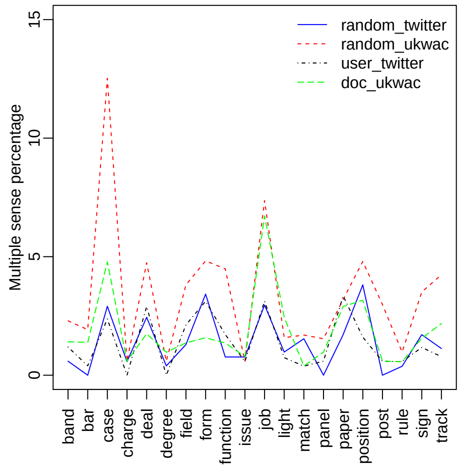
fall into kappa interpretation level are mentioned in Table 4.5.

Lemmas which showed highest and lowest IAA is presented in Table 4.7 for all the four data samples. The lemma degree showed consistently highest IAA across all the four data samples whereas they had different lemmas which showed the least IAA. Over the random sample Twitter dataset most of the lemmas had κ, the IAA in the range $0.21 - 0.40$ whereas for the other three datasets most of the lemmas had κ in the range $0.41 - 0.80$ showing a higher level agreement. These results depict that sense-tagging over Twitter is difficult compared to standard English data.

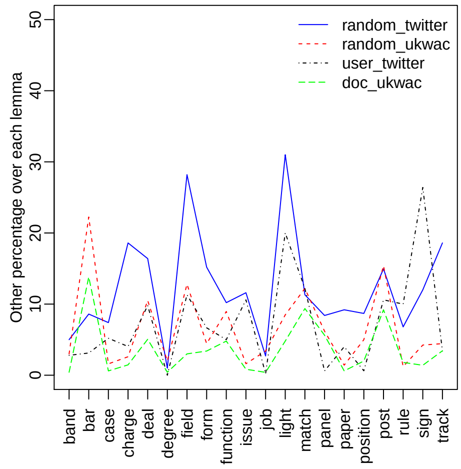
4.3.8 Multiple or Other sense labels

In this section we analyse the multiple sense labels or *Other* label that we observed over each dataset. In Figure 4.4a we show the multiple sense label percentage from the annotations for each lemma over 4 different types of annotations. Overall multiple sense applicability was found to be high in ukWac sentences rather than Twitter. The percentage of multiple labels for the random sample from Twitter was 1.37% and remained the same over user based Twitter sample. However the ukWac random sample showed highest percentage of 3.47%, whereas over document based ukWac sampled showed a decrement and is 1.87%. The highest percentage of multiple senses was observed for lemmas *case, position, form, deal* over sentences form ukWac.

As part of annotation guidelines, annotators were also given an option of *Other* which could be chosen when they do not find any of the listed senses applicable. This option provides an insight of applicability of chosen sense inventory for both the datasets. In Figure 4.4b we show the percentage of other label used in all 4 datasets over each lemma. The percentage of *Other* used in random sample Twitter over all lemmas is 12.31% and over random sample ukWac is 6.57%. User based sample over Twitter showed 7.35% and document based sample over ukWac showed 3.62%. Highest percentage of *Other* label was found for lemmas *light, field, charge, post, deal,*



(a) Multiple sense percentage for all datasets



(b) Other sense label for all datasets

Figure 4.4: Percentage of annotations for each dataset which showed multiple and other sense labels.

sign over Twitter datasets.

Analysis of *Other* label: We executed a manual analysis over *Other* label sentences and annotator input comments. We observed that apart from unlisted sense labels over Twitter there were few cases where the target word was wrongly mapped as noun by POS tagger i.e., due to POS tagger inaccuracy. A few other cases we observed were related to ill-formed words or typos and also quite a few non interpretable texts. This addresses that point which we mentioned earlier that Twitter text is associated with interpretability issues.

Overall multiple label percentage was seen higher over ukWac and *Other* label was seen higher over Twitter. This shows that the sense inventory used was not able to capture all the senses that is being used in Twitter messages. However, it is interesting to see that multiple sense applicability is lower over Twitter.

4.4 Summary

This analysis strengthens the point that sense tagging over Twitter is difficult when compared to ukWac or standard English text. We have observed more multiple labels over ukWac whereas more “Other” label or Twitter datasets. This shows that Twitter data has a higher percentage of instances where the senses could not be matched with the sense inventory used for general English which is a clear indication of novel senses. In Table 4.3 we observed that only 12 out of 20 frequent senses matched over Twitter and ukWac. This addresses our Research Question 5 that sense distribution across social media and general English do not match completely and show significant differences.

This analysis also addresses our Research Question 3 that sense distributions across social media do exhibit one predominant sense. However, overall the percentage of frequent labels is lower when compared to ukWac. This is evident in the Table 4.3.

Our analysis over user level sample from Twitter shows higher agreement than document level over ukWac, both user/document level and also sentence pair level and this address our Research Question 4 that on Twitter all messages from a user containing target word within a time-frame could be considered as a document and 65% of documents observed exhibit one sense per discourse phenomena. We have also executed sentence pairwise analysis and observed that 95.35% of the sentence pairs (which had confident sense label) from Twitter user sample showed one-sense-per-discourse phenomena.

Chapter 5

Conclusion

5.1 Further Work

In the previous chapters we have reviewed and proposed a viable approach to estimate usage similarity over social media texts. However, we observed that there is a scope in improving the performance of our approach using different background corpora or additional features.

In order to improve results of usage similarity over Twitter messages, possible approaches that could be investigated are

1. Extension to other POS categories

   In all of our experiments we have evaluated and worked on nouns. We would like to extend this to other major part of speech categories like verb, adjective and adverb. We did a preliminary analysis over these categories on standard English and observed varying performance with the different categories. A similar study could be performed on Twitter messages to analyse the difference between each part-of-speech category behavior on standard English and Twitter.

2. User based document expansions

   In Chapter 3 we executed usage similarity over 4 different background corpora and we observed that hashtag based expansion have shown good improvements in performance of our systems on all different approaches we tried. This addresses the issue of sparseness in the context of less context tokens. One potential way to expand documents is by concatenating Twitter messages from the same user in a specified time frame which contain a given target word. This idea is also inspired by analysing the sense patterns over Twitter users in Chapter 4 who tend to use one sense per word over a specific time period. This heuristic could be very much related to the one-sense-per-discourse phenomenon explored in sense disambiguation tasks.

3. Considering messages with more English lexical tokens in background collection

   In our background collections we have considered all the Twitter messages which had our target word. However, there are quite a few messages which do not have at least 2 English lexical tokens and contained typo graphical errors and ill-formed words. We observed that when *Usim tweet* dataset is evaluated on the models trained using sentences from I-EN corpus, these models showed a promising performance although it was lower compared to our hash-tag based expanded corpus. Inspired form this we believe working on a background corpus with more English lexical tokens might improve the performance of estimating usage similarity.

4. Experimenting HDP on smaller background collection

   Over all the background collections with which we experimented the topic-modelling approach has been shown to out-perform other approaches. However, T the number of topics we learn is given as a parameter. We experimented with topics in the range of $2 - 500$ to determine the topic number at which our target word performs best. The topic number was shown to vary for each target word. There are non-parametric LDA variant approaches like Hierarchical Dirichlet Processing (HDP) (Teh *et al.* 2006) which automatically learns a huge number of topics that best-fit and represent the background collection. This could be an alternative to make our approach parameter-free. However, we had a huge number of documents and applying HDP over a huge background collection is computationally expensive. A smaller background collection could be used to investigate the application of HDP for this task. HDP was earlier investigated by (Lau *et al.* 2012b) to automatically induce word senses from corpora and they have proved to perform better than LDA based approaches for sense induction tasks.

5. Considering word position information in documents (Exploring syntax)

   To the best of our knowledge there is no efficient dependency parser available over Twitter messages and a parser trained on standard English might not be efficient in understanding relations over Twitter messages. Due to this we had an issue with exploiting syntax over Twitter messages. One possible way to incorporate syntax into the bag-of-words approach is by adding word position information. This was earlier studied by (Lau *et al.*

2012b) for inducing word senses from standard English corpora and was shown to be useful in inducing word-senses.

6. Global topic models for the background collection with more lexical tokens / expanded documents

   Instead of working on word-based models one alternative way to represent meaning is using global topic modeling by trying to capture global senses. This method has been shown in Chapter 2 which work well for estimating meaning for two different words (Dinu and Lapata 2010). This could be investigated when we have enough lexical tokens in background collection or on expanded documents.

Other possible future work could be incorporating usage similarity tweet measures into practical applications to see if they could improve the performance of the application. One possible application is finding similar tweets of a given tweet over Twitter.

5.2 Summary and Final Thoughts

Given the amount of data being generated daily over social media there is huge need for understanding the text to filter important messages. To filter out important messages or texts we should understand the semantics of the message and to understand this, we should be able to interpret the individual word meaning from the context. This is a straightforward task when the target word is monosemous. However, it is observed that most frequent words over English are polysemous and exhibit multiple and related meanings. The traditional way to understand meaning from context is to execute it as a sense disambiguation task. We thoroughly review all approaches related to sense disambiguation task in Chapter 2. However, due to the challenges faced by social media texts and lack of sense tagged resources we chose the usage similarity as the meaning interpretation task. Usage similarity was proposed by (Erk *et al.* 2009) as a manual task to estimate the similarity of two usages of the target word two contexts. In Chapter 3 we automated the usage similarity estimation in an unsupervised fashion.

Estimating usage similarity over social media has not been explored before to the best of our knowledge. As we are targeting to understand word meanings we have focused on word-based models i.e., an individual model for each target word. We have analyzed three different possibilities of estimating usage similarity using unsupervised approaches, a baseline approach, a proposed approach and a benchmark approach. A baseline approach using distributional vector space model, a topic modeling approach (our approach) and a benchmark approach using weighted textual matrix factorisation. We found that our LDA based topic modeling approach out-performed both baseline and benchmark over different background corpora.

To evaluate our approaches of automating usage similarity we created a gold standard dataset *Usim-tweet* using Amazon Mechanical Turk. In the analysis we executed in Chapter 3 we have shown that estimating usage similarity over social media especially Twitter messages is viable using a word-based topic modeling approach. We also address our RESEARCH QUESTION 2 by showing that expanding the twitter messages using hash-tag based expansion showed significant improvement in the performance. However, the overall performance we observed was much less when compared to inter annotator agreement (ITA). On a few lemmas we were able to perform competitively with ITA.

In Chapter 3 we show that topic modeling based approach could be used in the semantic similarity task which is very much related task to usage similarity. We participated in Semantic Textual Similarity shared task 2013 and our systems based on topic modeling features have shown some fruitful results. When topic modeling features are combined with string similarity and information-retrieval based features we have seen good improvement in the performance of our systems.

In Chapter 4 we execute a pilot sense tagging task over Twitter messages using a coarse-grained dictionary. We have analysed overall sense distributions across Twitter messages and compared it with ukWac sentences. We have observed that the sense distribution across Twitter messages are different when compared with standard English text and also showed a higher percentage of unseen senses. This strengthens the point that a sense inventory developed to capture meanings in standard English might not be totally applicable to capture usages or meanings over social media. However, a higher percentage of multiple senses are observed over ukWac sentences.

We analysed one sense per user phenomena over Twitter messages and similarly one-sense-per-discourse/document over ukWac. Our analysis shows that Twitter users exhibit one sense per word over messages in a specific time period. These phenomena are observed to be stronger over Twitter users when compared with ukWac documents.

In this thesis we have reviewed the possibility of understanding word meaning in the context of social media. We experimented with potential approaches and showed that estimating usage similarity in an unsupervised fashion over social media texts is a viable task. There are a few more points that should be addressed like lower performance when compared to IAA which we leave as future work. We also gave a detailed description of possible future work addressing these issues.

Bibliography

2012. *langid.py: An Off-the-shelf Language Identification Tool*, Jeju, Republic of Korea.

Agirre, Eneko, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, 385–393, Montreal, Canada.

——, ——, ——, ——, and Weiwei Guo. to appear. *sem 2013 shared task: Semantic textual similarity, including a pilot on typed-similarity. In **SEM 2013: The Second Joint Conference on Lexical and Computational Semantics*, Atlana, USA. Association for Computational Linguistics.

——, and Philip Glenny Edmonds. 2006. *Word sense disambiguation: Algorithms and applications*, volume 33. Springer Science+ Business Media.

——, and Aitor Soroa. 2009. Personalizing pagerank for word sense disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, 33–41. Association for Computational Linguistics.

——, and Mark Stevenson. 2006. Knowledge sources for wsd. *Word Sense Disambiguation* 217–251.

Aliguliyev, Ramiz M. 2009. A new sentence similarity measure and sentence based extractive technique for automatic text summarization. *Expert Systems with Applications* 36.7764–7772.

Baldwin, Timothy, Colin Bannard, Takaaki Tanaka, and Dominic Widdows. 2003. An empirical model of multiword expression decomposability. In *Proceedings of the ACL 2003 workshop on Multiword expressions: analysis, acquisition and treatment-Volume 18*, 89–96. Association for Computational Linguistics.

Banerjee, Satanjeev, and Ted Pedersen. 2002. An adapted lesk algorithm for word sense disambiguation using wordnet. In *Computational linguistics and intelligent text processing*, 136–145. Springer.

Bennett, Shea, 2012. Twitter on track for 500 million total users by March, 250 million active users by end of 2012. `http://www.mediabistro.com/alltwitter/twitter-active-total-users_b17655`.

Bentivogli, Luisa, Marcello Federico, Giovanni Moretti, and Michael Paul. 2011. Getting expert quality from the crowd for machine translation evaluation. *Proceedings of the MT Summmit* 13.521–528.

Blei, David M., Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research* 3.993–1022.

Castillo, Julio, and Paula Estrella. 2012. Semantic textual similarity for mt evaluation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, 52–58. Association for Computational Linguistics.

Cook, Paul, and Suzanne Stevenson. 2009. An unsupervised model for text message normalization. In *Proceedings of the Workshop on Computational Approaches to Linguistic Creativity*, 71–78. Association for Computational Linguistics.

Cowie, Jim, Joe Guthrie, and Louise Guthrie. 1992. Lexical disambiguation using simulated annealing. In *Proceedings of the 14th conference on Computational linguistics-Volume 1*, 359–365. Association for Computational Linguistics.

De Boni, Marco, and Suresh Manandhar. 2003. The use of sentence similarity as a semantic relevance metric for question answering. In *Proceedings of the AAAI Symposium on New Directions in Question Answering*.

Dinu, Georgiana, and Mirella Lapata. 2010. Measuring distributional similarity in context. 1162–1172, Cambridge, USA.

Earle, Paul, Michelle Guy, Richard Buckmaster, Chris Ostrum, Scott Horvath, and Amy Vaughan. 2010. Omg earthquake! can twitter improve earthquake response? *Seismological Research Letters* 81.246–251.

Erk, Katrin, Diana McCarthy, and Nicholas Gaylord. 2009. Investigations on word senses and word usages. In *Proceedings of the Joint conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL-IJCNLP 2009)*, 10–18, Singapore.

——, ——, and ——. 2012. Measuring word meaning in context. *Computational Linguistics* 1–44.

Ferraresi, Adriano, Eros Zanchetta, Marco Baroni, and Silvia Bernardini. 2008. Introducing and evaluating ukwac, a very large web-derived corpus of english. In *Proceedings of the 4th Web as Corpus Workshop (WAC-4) Can we beat Google*, 47–54.

Fort, Karën, Gilles Adda, and K Bretonnel Cohen. 2011. Amazon mechanical turk: Gold mine or coal mine? *Computational Linguistics* 37.413–420.

Foster, Jennifer, Özlem Çetinoglu, Joachim Wagner, Joseph Le Roux, Stephen Hogan, Joakim Nivre, Deirdre Hogan, Josef Van Genabith, and others. 2011. # hardtoparse: Pos tagging and parsing the twitterverse. In *proceedings of the Workshop On Analyzing Microtext (AAAI 2011)*, 20–25.

Francis, W Nelson, and Henry Kucera. 1979. Brown corpus manual. *Brown University Department of Linguistics* .

Gale, William A, Kenneth W Church, and David Yarowsky. 1992. One sense per discourse. In *Proceedings of the workshop on Speech and Natural Language*, 233–237. Association for Computational Linguistics.

Gella, Spandana, Bahar Salehi, Marco Lui, Karl Grieser, Paul Cook, and Tim Baldwin. 2013. Unimelb nlp-core: Integrating predictions from multiple domains and feature sets for estimating semantic textual similarity. In *Proceedings of the Second Joint Conference on Lexical and Computational Semantics: Proceedings of the main conference and the shared task: Proceedings of the Seventh International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Gilks, Walter R, Sylvia Richardson, and David J Spiegelhalter. 1996. *Markov chain Monte Carlo in practice*, volume 2. Chapman & Hall/CRC.

Go, Alec, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford* 1–12.

Gouws, Stephan, Donald Metzler, Congxing Cai, and Eduard Hovy. 2011. Contextual bearing on linguistic variation in social media. In *Proceedings of the Workshop on Languages in Social Media*, 20–29. Association for Computational Linguistics.

Greenhow, Christine, and Benjamin Gleason. 2012. Twitteracy: Tweeting as a new literacy practice. In *The Educational Forum*, volume 76, 464–478. Taylor & Francis.

Guo, Weiwei, and Mona Diab. 2012a. Modeling sentences in the latent space. In *Proc. of the 50th Annual Meeting of the Association for Computational Linguistics*, 864–872, Jeju, Republic of Korea.

——, and ——. 2012b. A simple unsupervised latent semantics based approach for sentence similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1*, 586–590. Association for Computational Linguistics.

Han, Bo, Paul Cook, Tim Baldwin, and Diana Mccarthy. 2012a. A pilot study on word sense usages in social media, technical report. *Unpublished manuscript* .

——, Paul Cook, and Timothy Baldwin. 2012b. Automatically constructing a normalisation dictionary for microblogs. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning 2012*, 421–432, Jeju, Republic of Korea.

Harris, Zellig S. 1954. Distributional structure. *Word* .

Herdağdelen, Amaç, and Marco Baroni. 2011. Stereotypical gender actions can be extracted from web text. *Journal of the American Society for Information Science and Technology* 62.1741–1749.

Hong, Lichan, Gregoria Convertino, and Ed H. Chi. 2011. Language matters in Twitter: A large scale study. In *Proceedings of the 5th International Conference on Weblogs and Social Media (ICWSM 2011)*, 518–521, Barcelona, Spain.

Hovy, Eduard, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. Ontonotes: the 90% solution. In *Proceedings of the human language technology conference of the NAACL, Companion Volume: Short Papers*, 57–60. Association for Computational Linguistics.

Ide, Nancy, and Jean Véronis. 1998. Introduction to the special issue on word sense disambiguation: the state of the art. *Computational linguistics* 24.2–40.

——, and Yorick Wilks. 2006. Making sense about sense. In *Word Sense Disambiguation*, 47–73. Springer.

Jeon, Jiwoon, W Bruce Croft, and Joon Ho Lee. 2005. Finding similar questions in large question and answer archives. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, 84–90. ACM.

Jiang, Jay J, and David W Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. *arXiv preprint cmp-lg/9709008* .

Kauchak, David, and Regina Barzilay. 2006. Paraphrasing for automatic evaluation. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, 455–462. Association for Computational Linguistics.

Kazai, Gabriella, and Natasa Milic-Frayling. 2009. On the evaluation of the quality of relevance assessments collected through crowdsourcing. In *SIGIR 2009 Workshop on the Future of IR Evaluation*, p. 21. Citeseer.

Kilgarrif, Adam. 1998. Gold standard datasets for evaluating word sense disambiguation programs. *Computer Speech and Language* 12.453–472.

Krishnamurthy, Ramesh, and Diane Nicholls. 2000. Peeling an onion: The lexicographer's experience ofmanual sense-tagging. *Computers and the Humanities* 34.85–97.

Krovetz, Robert. 1998. More than one sense per discourse. *NEC Princeton NJ Labs., Research Memorandum* .

Lapata, Maria, Frank Keller, and Scott McDonald. 2001. Evaluating smoothing algorithms against plausibility judgements. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, 354–361. Association for Computational Linguistics.

Lau, Jey Han, Nigel Collier, and Timothy Baldwin. 2012a. On-line trend analysis with topic models: #twitter trends detection topic model online. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, 1519–1534, Mumbai, India.

——, Paul Cook, Diana McCarthy, David Newman, and Timothy Baldwin. 2012b. Word sense induction for novel sense detection. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2012)*, 591–601, Avignon, France.

Leacock, Claudia, and Martin Chodorow. 1998. Combining local context and wordnet similarity for word sense identification. *WordNet: An electronic lexical database* 49.265–283.

Lesk, Michael. 1986. Automatic 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*, 24–26. ACM.

Lin, Dekang. 1998. An information-theoretic definition of similarity. In *Proceedings of the 15th international conference on Machine Learning*, volume 1, 296–304. San Francisco.

Lui, Marco, Timothy Baldwin, and Diana McCarthy. 2012. Unsupervised estimation of word usage similarity. In *Proceedings of the Australasian Language Technology Workshop 2012 (ALTW 2012)*, 33–41, Dunedin, New Zealand.

MARTINEZ, DAVID, and ENEKO AGIRRE. 2000. One sense per collocation and genre/topic variations. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13*, 207–215. Association for Computational Linguistics.

MCCALLUM, ANDREW KACHITES. 2002. Mallet: A machine learning for language toolkit.

MCCARTHY, DIANA, ROB KOELING, JULIE WEEDS, and JOHN CARROLL. 2004. Finding predominant word senses in untagged text. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, p. 279. Association for Computational Linguistics.

——, SRIRAM VENKATAPATHY, and ARAVIND K JOSHI. 2007. Detecting compositionality of verb-object combinations using selectional preferences. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 369–379.

MIHALCEA, RADA. 1998. Semcor semantically tagged corpus. *Unpublished manuscript* .

——. 2005. Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labeling. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, 411–418. Association for Computational Linguistics.

——, TIMOTHY CHKLOVSKI, and ADAM KILGARRIFF. 2004. The senseval-3 english lexical sample task. In *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, 25–28. Barcelona, Spain, Association for Computational Linguistics.

MILLER, GEORGE A. 1995. Wordnet: a lexical database for english. *Communications of the ACM* 38.39–41.

MITCHELL, JEFF, and MIRELLA LAPATA. 2008. Vector-based models of semantic composition. *proceedings of ACL-08: HLT* 236–244.

MOONEY, RAYMOND J. 1996. Comparative experiments on disambiguating word senses: An illustration of the role of bias in machine learning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 82–91. Philadelphia, PA.

NAVIGLI, ROBERTO. 2009. Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)* 41.10.

NG, HWEE TOU, and HIAN BENG LEE. 1996. Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, 40–47. Association for Computational Linguistics.

OSBORNE, MILES, SASA PETROVIĆ, RICHARD MCCREADIE, CRAIG MACDONALD, and IADH OUNIS. 2012. Bieber no more: First story detection using Twitter and Wikipedia. In *Proceedings of the SIGIR 2012 Workshop on Time-aware Information Access*, Portland, USA.

OWOPUTI, OLUTOBI, BRENDAN O'CONNOR, CHRIS DYER, KEVIN GIMPEL, and NATHAN SCHNEIDER. 2012. Part-of-speech tagging for Twitter: Word clusters and other advances. Technical Report CMU-ML-12-107, Carnegie Mellon University.

Page, Lawrence, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking: bringing order to the web.

Palmer, Martha, Hoa Trang Dang, and Christiane Fellbaum. 2007. Making fine-grained and coarse-grained sense distinctions, both manually and automatically. *Natural Language Engineering* 13.137.

Patwardhan, Siddharth, Satanjeev Banerjee, and Ted Pedersen. 2003. Using measures of semantic relatedness for word sense disambiguation. In *Computational linguistics and intelligent text processing*, 241–257. Springer.

Paul, Douglas B, and Janet M Baker. 1992. The design for the wall street journal-based csr corpus. In *Proceedings of the workshop on Speech and Natural Language*, 357–362. Association for Computational Linguistics.

Pedersen, Ted, Siddharth Patwardhan, and Jason Michelizzi. 2004. Wordnet:: Similarity: measuring the relatedness of concepts. In *Demonstration Papers at HLT-NAACL 2004*, 38–41. Association for Computational Linguistics.

Porter, Martin F. 1980. An algorithm for suffix stripping. *Program: electronic library and information systems* 14.130–137.

Purandare, Amruta, and Ted Pedersen. 2004. Word sense discrimination by clustering contexts in vector and similarity spaces. In *Proceedings of the Conference on Computational Natural Language Learning*, 41–48. Boston.

Reddy, Siva, Ioannis Klapaftis, Diana McCarthy, and Suresh Manandhar. 2011. Dynamic and static prototype vectors for semantic composition. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, 705–713.

Redington, Martin, and Nick Chater. 1997. Probabilistic and distributional approaches to language acquisition. *Trends in Cognitive Sciences* 1.273–281.

Reisinger, Joseph, and Raymond J Mooney. 2010. Multi-prototype vector-space models of word meaning. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 109–117. Association for Computational Linguistics.

Resnik, Philip. 1992. Wordnet and distributional analysis: A class-based approach to lexical discovery. In *AAAI workshop on statistically-based natural language processing techniques*, 56–64.

——. 1995. Using information content to evaluate semantic similarity in a taxonomy. *arXiv preprint cmp-lg/9511007* .

——, and David Yarowsky. 1997. A perspective on word sense disambiguation methods and their evaluation. In *Proceedings of the ACL SIGLEX workshop on tagging text with lexical semantics: Why, what, and how*, 79–86.

Ritter, Alan, Sam Clark, Mausam, and Oren Etzioni. 2011. Named entity recognition in tweets: An experimental study. 1524–1534, Edinburgh, UK.

Schmid, Helmut. 1994. Treetagger. *TC project at the Institute for Computational Linguistics of the University of Stuttgart* .

Schnoebelen, Tyler, and Victor Kuperman. 2010. Using amazon mechanical turk for linguistic research. *Psihologija* 43.441–464.

Schütze, Hinrich. 1998. Automatic word sense discrimination. *Computational Linguistics* 24.97–123.

Sharoff, Serge. 2006. Open-source corpora: Using the net to fish for linguistic data. *International Journal of Corpus Linguistics* 11.435–462.

Sinha, Ravi, and Rada Mihalcea. 2007. Unsupervised graph-basedword sense disambiguation using measures of word semantic similarity. In *Semantic Computing, 2007. ICSC 2007. International Conference on*, 363–369. IEEE.

Srebro, Nathan, and Tommi Jaakkola. 2003. Weighted low-rank approximations. In *Proceedings of the 20th International Conference on Machine Learning*, Washington, USA.

Steyvers, Mark, and Tom Griffiths. 2007. Probabilistic topic models. *Handbook of latent semantic analysis* 427.424–440.

Teh, Yee Whye, Michael I Jordan, Matthew J Beal, and David M Blei. 2006. Hierarchical dirichlet processes. *Journal of the American Statistical Association* 101.

Thater, Stefan, Hagen Fürstenau, and Manfred Pinkal. 2011. Word meaning in context: A simple and effective vector model. In *Proceedings of IJCNLP*.

Vuurens, Jeroen, Arjen P de Vries, and Carsten Eickhoff. 2011. How much spam can you take? an analysis of crowdsourcing results to increase accuracy. In *Proc. ACM SIGIR Workshop on Crowdsourcing for Information Retrieval (CIR11)*, 21–26.

Wallach, Hanna, David Mimno, and Andrew McCallum. 2009. Rethinking lda: Why priors matter. *Advances in Neural Information Processing Systems* 22.1973–1981.

Wang, Alex Hai. 2010. Don't follow me: Spam detection in twitter. In *Security and Cryptography (SECRYPT), Proceedings of the 2010 International Conference on*, 1–10. IEEE.

Yarowsky, David. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, 189–196. Association for Computational Linguistics.

Yuen, Man-Ching, Irwin King, and Kwong-Sak Leung. 2011. A survey of crowdsourcing systems. In *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*, 766–773. IEEE.