Unsupervised All-words Sense Distribution Learning

A thesis presented

by

Andrew Bennett (319653)

to

The Department of Computing and Information Systems

in partial fulfillment of the requirements

for the degree of

Master of Science (Computer Science)

Project type: Research Project (75 points)

Subject code: COMP60004

Main supervisor: Timothy Baldwin

Secondary supervisors: Jey Han Lau, Diana McCarthy, Francis Bond

The University of Melbourne

Melbourne, Australia

June 2016

Declaration

I certify that:

(i) This thesis does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any university; and that to the best of my knowledge and belief it does not contain any material previously published or written by another person where due reference is not made in the text.

(ii) Where necessary I have received clearance for this research from the University's Ethics Committee and have submitted all required data to the Department.

(iii) The thesis is approximately 28,500 words in length (excluding text in images, table, bibliographies and appendices).

Thesis advisor(s)
Timothy Baldwin
Jey Han Lau, Diana McCarthy, Francis Bond

Author
Andrew Bennett

Unsupervised All-words Sense Distribution Learning

Abstract

There has recently been significant interest in unsupervised methods for learning word sense distributions, or most frequent sense information, in particular for applications where sense distinctions are needed. In addition to their direct application to word sense disambiguation (WSD), particularly where domain adaptation is required, these methods have successfully been applied to diverse problems such as novel sense detection or lexical simplification. Furthermore, they could be used to supplement or replace existing sources of sense frequencies, such as SemCor, which have many significant flaws. However, a major gap in the past work on sense distribution learning is that it has never been optimised for large-scale application to the entire vocabularies of a languages, as would be required to replace sense frequency resources such as SemCor.

In this thesis, we develop an unsupervised method for all-words sense distribution learning, which is suitable for language-wide application. We first optimise and extend HDP-WSI, an existing state-of-the-art sense distribution learning method based on HDP topic modelling. This is mostly achieved by replacing HDP with the more efficient HCA topic modelling algorithm in order to create HCA-WSI, which is over an order of magnitude faster than HDP-WSI and more robust. We then apply HCA-WSI across the vocabularies of several languages to create LexSemTm, which is a multilingual sense frequency resource of unprecedented size. Of note, LexSemTm contains sense frequencies for approximately 88% of polysemous lemmas in Princeton WordNet, compared to only 39% for SemCor, and the quality of data in each is shown to be roughly equivalent. Finally, we extend our sense distribution learning methodology to multiword expressions (MWEs), which to the best of our knowledge is a novel task (as is applying any kind of general-purpose WSD methods to MWEs). We demonstrate that sense distribution learning for MWEs is comparable to that for simplex lemmas in all important respects, and we expand LexSemTm with MWE sense frequency data.

Contents

List of Figures

List of Tables

Citations to Previously Published Work

Large portions of Chapter 4 and Chapter 5 will appear in the following paper:

> Andrew Bennett, Timothy Baldwin, Jey Han Lau, Diana McCarthy, and Francis Bond (2016) LexSemTm: A Semantic Dataset Based on All-words Unsupervised Sense Distribution Learning, to appear in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Berlin, Germany.

Acknowledgments

My greatest thanks go to my supervisors, especially my main supervisor Tim Baldwin, who has provided countless hours of support in order to help me throughout my Masters research project. He has helped me develop as a researcher, and is both a mentor and a friend. My strong thanks also extend to my co-supervisors: Jey Han Lau, who has also spent countless hours meeting and discussing the project, and providing his expertise; Diana McCarthy, for her expert knowledge of the field and invaluable advice; and Francis Bond, who has also provided helpful advice, as well as much of the data that has made this research possible. Without the assistance of everyone involved, this project could not have been as fruitful as it was, for which I am deeply grateful.

My thanks also extend to my close friends and family, who have supported me during the course of my research. It is in part thanks to their presence that I have remained sane and motivated throughout this project, so in many respects the success of this research is as much due to them as anyone else.

Finally, this work was supported in part by a Google Cloud Platform award, which facilitated the computation with the Google Compute Engine required to perform language-wide sense distribution learning and produce our LexSemTm dataset.

Chapter 1

Introduction

1.1 Problem Background

Word senses and methods for disambiguating or otherwise dealing with word senses have been of great interest to the Natural Language Processing (NLP) community for several decades (for a detailed overview, see Agirre and Edmonds (2007) and Navigli (2009), and the citations therein). A word sense is a possible meaning of a word. For example the lemma *crane* has at least two senses: a `bird` sense if the word is referring to the animal, or a `machine` sense if it is referring to the vehicle for lifting heavy objects. Given this, word sense disambiguation (WSD) is the problem of deciding the most likely sense of a word given some context. For example, given the sentence *the crane flew towards the horizon*, we could reasonably disambiguate *crane* as having the `bird` sense, due the presence of key words like *flew* and *horizon*. WSD is potentially of utility anywhere where distinctions in word meaning are important — which is arguably every single problem involving natural language data, including text and speech — and has been successfully applied to a range of NLP or related problems.

One area where WSD has been applied successfully is in information retrieval (IR), specifically in text retrieval (Krovetz and Croft 1992; Gonzalo *et al.* 1998; Zhong and Ng 2012). This is the problem of obtaining relevant documents from a database based on a text query, for example as solved by the Google search engine. As an example to see how WSD is relevant to text retrieval, consider the query *Java programming language*. It is clear given the context within this query — specifically due to the keywords *programming* and *language* — that *Java* is being used in this query in its `programming language` sense (as opposed to its `Indonesian island` sense, or its `coffee` sense). This information could be used to refine retrieval results; for example, documents containing *Java* in its `programming language` sense could be given preference relative to documents containing *Java* in its other senses.

Another example area where WSD has shown utility is in machine translation (Agirre *et al.* 2008; Agirre *et al.* 2011). This is the problem of using computers to au-

tomatically translate text or speech from one language to another. Sense distinctions are important in this task because there is not a simple one to one correspondence in word meanings between languages. As an example to illustrate this, consider the problem of translating *the crane* from English to German: if *crane* was being used in its `bird` sense we would translate this to *der Kranich*, whereas if it was being used in its `machine` sense we would translate it to *der Kran*. Therefore, accurate machine translation requires WSD to be performed either explicitly or implicitly.

An important distinction in WSD research is between supervised and unsupervised methods. Supervised WSD methods are those that make use of sense-labelled usages[1] to learn from, whereas unsupervised WSD methods are those that learn from raw, unlabelled text (or speech). Unsupervised WSD methods have been attracting increasing attention in recent years, as they address some of the serious shortcomings of supervised WSD methods (Navigli 2009). First of all, the labelled data needed to perform supervised WSD is expensive to obtain, since it involves manually annotating large numbers of usages per lemma.[2] This is especially true if we want to obtain sense-labelled data on a language-wide scale, as we would need to train a supervised model that could be applied to all words in a language's vocabulary. Secondly, different text domains[3] have different patterns in sense occurrences; in order to take advantage of this, supervised methods require separate labelled usages from each possible text domain, which would further inflate the cost of obtaining labelled data. Thirdly, supervised methods usually require the use of a fixed sense inventory;[4] they learn from sense-labelled data, which means it is non-trivial to use them to perform WSD with a different sense inventory than was used to label the training data.

While there exist a wide range of approaches to unsupervised WSD (Navigli 2012), many methods are based on the difficult-to-beat most frequent sense (MFS) or first sense heuristic (McCarthy *et al.* 2004a). The MFS heuristic simply disambiguates every usage of a given polysemous[5] lemma with the sense that is most common in general (that is, the MFS). For example, if we knew that the noun *bank* was most commonly used in its `financial institution` sense, we would apply the MFS heuristic by disambiguating every usage of the noun with this sense, regardless of context. This heuristic is also sometimes referred to as the "first sense" heuristic, because in some major dictionaries (such as WordNet, which is described in detail in Section 3.1.2) the MFS of each word is listed first.[6] This heuristic is surprisingly strong

[1] That is, example usages of the target word that is being disambiguated, which have been labelled with their correct senses.

[2] Note that in this thesis we will often use "lemma" and "word" interchangeably. For our purposes we can define a lemma to be the dictionary form of a word, along with a part of speech (such as noun, verb, adjective, or adverb).

[3] For example, news articles about sport, or journal articles about physics.

[4] A sense inventory is a dictionary that defines the set of all possible senses for each lemma in a given vocabulary.

[5] A polysemous lemma is a lemma with more than one possible sense.

[6] For this reason, we use the terms "MFS" and "first sense" interchangeably in this thesis.

in practice because word sense frequencies tend to follow power law distributions according to Zipf's Law (Zipf 1936; Zipf 1949), meaning that the MFS is typically very dominant. This is particularly true when we look at sense frequencies within a specific domain (Piantadosi 2014), and for this reason the MFS heuristic can also be applied specifically within a given domain, based on the MFS of each word within that domain (Koeling *et al.* 2005). For example, the MFS of *bank* may be `financial institution` within domains related to finance or economics, whereas it may be `river bank` within domains related to nature or geography.

Using the MFS heuristic, the problem of unsupervised WSD can be reduced to that of learning probability distributions over the senses of each lemma, which we refer to as sense distribution learning. This is because the sense distribution of any given lemma directly gives the lemma's MFS (the sense with maximum probability). Furthermore, sense distributions can be used to automatically determine when the MFS heuristic is appropriate to use, based on the sense distribution's entropy (Jin *et al.* 2009); if a sense distribution has relatively high entropy, it means the MFS is less dominant.

In addition to their direct use in WSD and the associated applications, sense distributions, or MFS information, have been successfully used in a variety of other applications. For example, they have proven useful in performing automatic lexical simplification (Biran *et al.* 2011), which is the problem of automatically simplifying text to be more readable while preserving its meaning (for example, simplifying *rotund fauna* to *fat animals*). Sense frequency information can be used here to find candidate word substitutes for simplification, by searching for words whose MFS matches the sense of the complex word to be replaced. In addition, sense distribution learning methods have been extended to detect novel senses of words (Lau *et al.* 2012; Lau *et al.* 2014). As an example, the noun *click* now has a `web traffic click` sense that didn't exist in the past, which is missing from many major dictionaries including WordNet; this could be automatically detected using sense distribution learning methodology. To this end, these methods have been successfully applied in the semi-automatic construction of new dictionary entries (Cook *et al.* 2013).

Of particular interest, sense distribution learning provides the promise of either supplementing or replacing existing resources containing the relative frequencies of word senses. These resources are currently very limited; the most prominent example of such a resource is SemCor (Miller *et al.* 1993), which is based on the Brown Corpus (Kucera and Francis 1967) — a large, balanced corpus[7] consisting of mid-20th century American literature — whose words have been labelled with WordNet senses.[8] Unfortunately, the data in SemCor contains many irregularities due to its age and limited coverage. For example, the MFS of *pipe* according to SemCor is `tobacco pipe`, whereas one might expect it to be `tube carrying water or gas`, which is

[7]In the context of this thesis, a corpus is a collection of text documents.

[8]The WordNet dictionary and the SemCor sense frequency resource are both described in detail in Section 3.1.2.

likely due to the age of the Brown corpus. Similarly, the MFS of *tiger* according to SemCor is `audacious person`, whereas one might expect it to be `carnivorous animal`. This is due to the fact that *tiger* has only two sense-labelled occurrences in the Brown corpus, which in this case is clearly insufficient to identify the "correct" MFS. Furthermore, SemCor contains significant gaps; indeed, of the roughly 31,000 polysemous lemmas in WordNet, approximately 61% have no sense-labelled occurrences in the Brown corpus (and therefore no sense frequency data in SemCor), and less than half of the remaining lemmas have at least five occurrences (which means that sense distributions for these lemmas are less likely to be reliable).

However, despite the various applications of sense distribution learning, and its potential to supplement or replace these flawed sense frequency resources, there are some glaring gaps in the past work that limits its use. Most importantly, there has been no prior work investigating how to apply sense distribution learning at the scale of a full lexical resource such as WordNet. Updating language-wide sense frequency resources like SemCor would require learning sense distributions over the entire vocabularies of languages, which could be extremely computationally expensive. Indeed, in our preliminary experiments using HDP-WSI (Lau *et al.* 2014) — the previous state-of-the-art method in sense distribution learning — we found that approximately one hour of computation time was needed per lemma on average.[9] Obviously over an entire vocabulary, containing potentially tens of thousands of lemmas, this quickly becomes intractable! To make matters even worse, dealing with domain differences[10] could require learning numerous sense distributions per word. While these considerations may suggest learning sense distributions as cheaply as possible, we would not want to make sense distribution learning scalable at the expense of sense distribution quality. Therefore, it would be desirable to understand the tradeoff between the accuracy and computation time of sense distribution learning, and how to optimise this tradeoff for large-scale application.

A second major limitation is that past work on sense distribution learning has focussed on simplex lemmas, which are lemmas consisting of a single word. However this neglects multiword expressions (MWEs), which can also be polysemous. A MWE is a lexical item consisting of multiple words, with a meaning that is not trivially predictable from the individual words, such as *top dog* (Baldwin and Kim 2010).[11] Indeed, of the approximately 31,000 polysemous lemmas in WordNet, close to 3,000 of these are MWEs. While there has been some past work dealing

[9]This number was obtained by applying HDP-WSI to the BNC dataset used by the authors in their work, which is described in Section 3.1.3.

[10]That is, dealing with the fact that text from different domains will have different sense frequencies.

[11]Note that for the purposes of this thesis we are not overly concerned with the precise definition of what it means for the meaning to not be trivially predictable from the component words. As a rough heuristic rule, we can define a MWE in practice as an item consisting of more than one word that would be reasonable to include in a dictionary.

with MWEs that is related to WSD,[12] to the best of our knowledge these have all addressed very specific cases of WSD, and general-purpose WSD or sense distribution learning for MWEs are still unsolved problems. Given this, it is unknown whether the general-purpose methods of sense distribution learning for simplex lemmas can be applied to MWEs. Furthermore, if they can be applied to MWEs, this introduces additional challenges such as automatically identifying[13] MWE usages from unlabelled text. Another possible challenge in applying sense distribution learning to MWEs is disambiguating between simple compositions of the individual words and actual MWE usages (Hashimoto and Kawahara 2008; Fothergill and Baldwin 2012).[14] However, making this distinction is very subtle and is methodologically questionable, since in many cases the simple composition has a corresponding dictionary sense (for example the `elderly man` sense of *old man* in WordNet), so we mostly ignore it in the rest of the thesis.[15]

1.2 Research Aims

Given this context, the primary aim of our thesis is to develop a generic method for unsupervised all-words sense distribution learning, which is capable of supplementing or replacing existing sources of sense frequencies such as SemCor. By generic we mean that the method can be applied as broadly as possible — that is, it is language-independent and can applied to all kinds of lemmas — and by all-words we mean that the method can be applied efficiently on a language-wide scale. In addition we define a secondary aim, which is to actually apply our method to create a new multilingual, language-wide, domain-independent sense frequency resource, which will hopefully be state-of-the-art. This resource could then be used either alongside or in place of existing resources like SemCor, as a general-purpose source of sense frequencies.

In order to address our primary and secondary aims, we define three core research questions that will help define the structure of our thesis:

1. What does a practical blueprint look like for efficiently applying sense distribution learning on a large scale, and achieving an optimal balance between accuracy and computation time?

[12]For example, work on named entity recognition, supersense tagging, or disambiguating between literal and idiomatic interpretations. We provide a brief overview of these tasks in Section 2.4.

[13]Note that in this thesis, we use the term "identification" in the context of MWEs specifically to describe recognising occurrences of known MWEs.

[14]As an example, consider the expression *red tape*, which could be a MWE referring to bureaucracy, or a simple composition of the component words referring to red coloured tape (which arguably should not be classed as a MWE).

[15]With the exception of the creation of our MWE sense-tagged dataset in Section 5.4, where we asked annotators to give sentences a specific label if they were simple compositions and there was no corresponding WordNet sense.

2. To what extent can unsupervised all-words sense distribution learning be used to supplement or replace existing sense frequency resources?

3. Can sense distribution learning also be applied to MWE lemmas, and if so how does this task compare to simplex sense distribution learning?

These research questions are made complex by the wide range of languages and classes of lemmas they could be applied to. In order to narrow the scope of our thesis, we mostly limit our evaluations in answering these questions to English nouns. Although in our aims we are interested in sense distribution learning for all languages and parts of speech (POS), this restriction in scope for our evaluation greatly simplifies analysis, and makes the cost of obtaining labelled data reasonable. We focus on English because this is the language most frequently studied in prior work, and is strongly resourced. Similarly, we focus on nouns because this is the POS most frequently studied in prior work, and because these lemmas are generally the most important to be able to disambiguate.[16]

Given our aims, research questions, and defined scope, the rest of this thesis proceeds as follows: In Chapter 2 we provide a review of the literature relevant to our research questions. In this review we make the case for building our unsupervised all-words sense distribution learning method upon the state-of-the-art HDP-WSI method of Lau *et al.* (2014), which is a modular method using HDP topic modelling. In addition, we identify HCA as a compelling potential replacement for HDP in HDP-WSI, given our aim of efficient large-scale application. Then in Chapter 3 we provide a detailed description of the resources and methods from past work that form the building blocks of our investigation, including the HDP and HCA topic modelling methods, and the HDP-WSI sense distribution learning method. In Chapter 4 we provide a detailed series of experiments in order to address our primary aim and answer our first research question. In this process we extend HDP-WSI to HCA-WSI by replacing HDP with HCA, and provide some guidelines for efficiently applying HCA-WSI on a large scale. Subsequently in Chapter 5 we address our secondary aim by applying HCA-WSI vocabulary-wide across English, Japanese, Italian, Mandarin and Indonesian to create our new LexSemTm sense frequency resource. This then allows us to more thoroughly evaluate HCA-WSI, and in doing so answer our second and third research questions. Specifically, we show that LexSemTm can at worst supplement SemCor by supplying sense frequency data for the majority of lemmas missing from SemCor, and can possibly replace SemCor-based sense frequencies altogether. Furthermore, we show that MWE sense distribution learning is possible, and that this task appears to be comparable with simplex sense distribution learning in all important respects. Finally, in Chapter 6 we provide a detailed summary of our

[16]This is because they more often contain context information, as well as domain-specific meanings (McCarthy *et al.* 2007).

findings, discuss their relevance, and provide some possible directions for extending this study in future work.

Chapter 2

Literature Review

2.1 Introduction

We now provide a review of the literature relevant to our aims of developing and applying a method of unsupervised all-words sense distribution learning. We break this literature review up into several sections. First, in Section 2.2 we review the past work directly related to sense distribution learning, which covers sense distribution and first sense learning, as well as the related tasks of unsupervised WSD and word sense induction (WSI: see Section 2.2.3). Next, in Section 2.3 we review some of the past work on topic modelling, since this is relevant to HDP-WSI (described in detail in Section 3.2.3), which is the particular sense distribution learning method we have chosen to build upon. Finally, in Section 2.4 we review some of the relevant literature related to identifying and disambiguating MWE usages, given that we wish to apply our sense distribution learning on as wide a scale as possible by targeting MWEs as well as simplex lemmas.

2.2 Sense Learning

Although to the best of our knowledge nobody has attempted to automatically learn sense frequencies on a language-wide scale before, there has been extensive work on learning sense frequencies — including sense distribution and first sense learning — in general. In addition to this, there has been extensive work on the problems of unsupervised WSD and WSI, which although not exactly the same problem as sense distribution learning are directly related. The bulk of this section is dedicated to exploring the existing methods of sense distribution and first sense learning. We systematically compare the competing methods (a summary of which is presented in Table 2.1), and from this identify HDP-WSI as an appropriate method to build upon. In addition, we provide a short overview of recent work on unsupervised WSD and WSI, placing this work in the context of our aims.

Method	Sense Inventory	Other Limitations
McCarthy *et al.* (2004a)	WordNet-like	Requires parsing of corpus
Boyd-Graber and Blei (2007)	WordNet-like	Requires parsing of corpus
Mohammad and Hirst (2006)	Thesaurus-like	—
Lapata and Brew (2004)	Levin (1993) verb classes	Requires POS tagging of corpus
Lapata and Keller (2007)	WordNet-like	—
Lau *et al.* (2014)	Any containing glosses	—
Bhingardive *et al.* (2015)	WordNet-like	—
Chan and Ng (2006)	Any	Requires trained WSD classifier
Brody *et al.* (2006)	Any	Relies on combining output of other methods (ensemble learning approach)
Loukachevitch and Chetviorkin (2015)	RuThes	Requires labelled sense frequency data

Table 2.1: Summary of different methods for first sense or sense distribution learning. For each method the kind of sense inventory it can be applied to is listed, as well as any other limitations of the method. By "WordNet-like" we mean that the sense inventory is assumed to have WordNet specific features (such as hypernym relations), and by "Thesaurus-like" we mean that the sense inventory is assumed to have a simple coarse-grained structure based on overlapping categories (such as in the Macquarie Thesaurus).

Note that in this chapter we use the phrases "sense distribution learning" and "first sense learning" interchangeably, since we consider them to be equivalent tasks (although authors tend to frame their methods as being one or the other). This is because first sense learning methods involve calculating some kind of predominance score for each sense in order to rank them, which could be normalised to give a distribution over the senses, and sense distribution learning automatically provides the first sense via the mode of the distribution. The terms used reflect the particular emphasis of the methods we review.

2.2.1 Sense Distribution and First Sense Learning

The original work on first sense and sense distribution learning came from McCarthy *et al.* (2004a). They proposed a method of calculating predominance scores for each sense of a target word from unlabelled corpora. Their method uses Word-

NET[1] as a sense inventory, and is based on calculating distributionally similar words with the target word based on the method of Lin (1998); these distributionally similar neighbours represent the contexts in which the target word is used. For each available WORDNET sense of the target word, their method calculates a predominance score by summing the similarity of the sense with each distributionally similar neighbour,[2] weighted by the neighbour's distributional similarity score. Their initially proposed method was tailored specifically to nouns, but was subsequently extended to other parts of speech (POS) (McCarthy *et al.* 2004b). Their first sense predictions were shown to be competitive with those based on SEMCOR[3] (although slightly worse), as evaluated by their accuracy for the purposes of unsupervised WSD on a general domain corpus using the first sense heuristic. However, the method is limited in that the distributional similarity calculations require parsing[4] of the input corpora (which is computationally expensive, and is not available for all languages), and it relies on using a WORDNET-like sense inventory — the methods used to calculate similarity between senses and distributionally similar neighbours make use of the network structure of WORDNET— which limits its applicability.

Subsequent work has demonstrated the effectiveness of McCarthy *et al.*'s (2004a) method. For example, Koeling *et al.* (2005) showed that their automatically generated first sense predictions outperformed SEMCOR when applied to domain-specific corpora, where they could learn domain-specific sense frequencies. In addition, McCarthy *et al.* (2007) showed that the method was able to outperform SEMCOR on general domain data for lemmas with fewer than five sense-labelled occurrences in SEMCOR, and Jin *et al.* (2009) showed that it performed particularly strongly in instances where the produced sense distribution was relatively skewed.

McCarthy *et al.*'s (2004a) method can be generalised by viewing it as a process of finding the contexts in which the target word appears (represented by the distributionally similar neighbours), and aligning these to the provided sense inventory (by calculating WORDNET similarities with each sense). This generalisation provided the basis for many of the subsequent first sense learning methods. For example, Boyd-Graber and Blei (2007) formalised the method of McCarthy *et al.* (2004a) as a graphical probabilistic model, which they then extended by introducing latent topic variables to model document-specific context patterns throughout the input corpus. However this extension was shown to result in minimal improvement (roughly a 1% increase in WSD accuracy), and their method comes with the same limitations as

[1]WORDNET is a dictionary with a specialised network structure, and is discussed in detail in Section 3.1.2.

[2]Similarity is given by the maximum WORDNET similarity between the target word sense, and each of the neighbour's senses. The WORDNET similarity functions of Banerjee and Pedersen (2003) and Jiang and Conrath (1997) were experimented with, which are discussed in Section 2.2.2.

[3]A large, balanced corpora with WORDNET sense annotations. SEMCOR is described in Section 3.1.2

[4]In the context of natural language processing, this is the problem of calculating the grammatical structure of text.

McCarthy *et al.*'s (2004a) method.

Mohammad and Hirst (2006) also presented a method similar to that of McCarthy *et al.* (2004a). However, their approach relies on using a sense inventory similar in structure to thesauri such as the Macquarie Thesaurus (Bernard 1986), where words are organised into overlapping categories, and each category of a word corresponds to a single sense. They proposed multiple methods for inferring sense frequencies with respect to such thesauri, based on co-occurrences of the target word with other words in the corresponding categories. They achieved very strong results in terms of raw WSD accuracy, using the Macquarie Thesaurus. However, their results are not directly comparable to those of McCarthy *et al.* (2004a) and others who use WordNet, since sense distinctions in WordNet are far more fine grained than those in the Macquarie Thesaurus,[5] which means their reported accuracy numbers are far higher than those that could be obtained if working with WordNet.

Similar to Mohammad and Hirst (2006), Lapata and Brew (2004) proposed a method that relies on a very specific non-WordNet sense inventory. In their case they used a sense inventory of verbs only, in which verbs are grouped into overlapping classes based on the English verb categorisation of Levin (1993). As with Mohammad and Hirst (2006), each category is assumed to correspond to a single sense. Lapata and Brew (2004) proposed a probabilistic model based on these verb categories — as well as known information of POS patterns for each category — which can be fit to the input unlabelled corpus. This is done by first POS tagging[6] the corpus, and then comparing the POS tags surrounding each verb usage to the allowed POS tags for each sense of the verb. Sense frequencies are present as latent variables in their model, which can be read out after fitting the model to data. As with Mohammad and Hirst (2006), despite the fact that they obtained strong results in terms of raw accuracy numbers, their method is limited by the very specific sense inventory used. That is especially true in this case, because it is only applicable to verbs.

A slightly different kind of approach to first sense learning was provided by Lapata and Keller (2007), who approached the problem using information retrieval techniques. Their method involves indexing a search engine over the input unlabelled corpus, and then inferring sense frequencies by making a series of targeted queries for each sense. For example, to infer the relative frequency of the `river bank` sense of *bank*, a query is made in the form of "*bank* AND *s*"[7] for each synonym or hypernym[8] *s* of the `river bank` sense of *bank* in WordNet, and the number of results returned by each query is counted. The authors experimented with multiple methods for com-

[5]This refers to the fact that words in WordNet often have multiple senses that are very similar in meaning. For example, the word *bank* has separate `financial institution` (referring to the company) and `bank building` (referring to the building itself) senses.

[6]POS tagging is explained in Section 3.1.4.

[7]The query is also expanded by adding inflected forms of words, such as plurals or different verb endings.

[8]See the WordNet description in Section 3.1.2 for details on how hypernyms are defined.

bining the resultant counts from these queries to produce sense frequency estimates. However, in all of their evaluations, the results of their method were at best comparable with those from McCarthy *et al.* (2004a).[9] Lapata and Keller's (2007) method has slightly less limitations than that of McCarthy *et al.* (2004a), since it does not require parsing, though it still relies on WordNet-specific features (in this case, the use of hypernyms).

More recently, Lau *et al.* (2014) proposed their HDP-WSI method for sense distribution learning using topic modelling. Like McCarthy *et al.* (2004a), theirs is a two-step method of identifying contexts of the target word, and aligning these to the given sense inventory, and as with Boyd-Graber and Blei (2007) they make use of topic modelling. Their method discovers contexts of the target word using the WSI (see Section 2.2.3) approach of Lau *et al.* (2012), which is based on topic modelling. However unlike Boyd-Graber and Blei (2007), this topic modelling is not performed on entire documents in the input corpus, but rather on individual usages of the target word, where each usage is treated as a separate "document". The results of this topic modelling are aligned to the provided sense inventory by comparing each topic to the gloss[10] of each possible sense. Therefore their method is very general, and unlike prior methods it can be applied to any sense inventory with glosses, with no language-specific restrictions. They evaluated their method against that of McCarthy *et al.* (2004a) on multiple datasets, and showed that it obtained comparable performance in terms of WSD accuracy with the MFS heuristic, and superior performance in terms of the overall quality of the resultant sense distributions (evaluated by comparing these sense distributions against gold-standard distributions obtained from manually sense-labelled data).

Another recent method has been proposed by Bhingardive *et al.* (2015). They proposed a method using word embeddings,[11] which were calculated using the method of Mikolov *et al.* (2013). Their method involves calculating vector representations for each sense of the target-word, in the same space as the word embeddings, based on related words to the sense (according to relations in the WordNet network). Sense predominance scores are then calculated according to the similarity of the target word with each of its senses, based on their respective vector representations (using cosine similarity). While on the surface they appear to deviate from past work by not making use of usage patterns of the target word, in reality such information should implicitly be encoded in the word vectors. They evaluated their method on both

[9] An exception was the performance for verbs, where McCarthy *et al.*'s (2004a) method performed relatively poorly (McCarthy *et al.* 2004b). However, these verbs are not usually the focus of sense learning work, since they do not contain as much context information or the same tendency for domain specific meanings as nouns for example (McCarthy *et al.* 2007).

[10] Worded definitions of the senses, as appear in standard dictionaries.

[11] This refers to real-valued vector representations of words, which capture subtle patterns in word meaning and word usage. A canonical example of a word embedding method is word2vec (Mikolov *et al.* 2013)

Hindi and English data: although they achieved strong results on the Hindi dataset relative to their benchmarks, their results for English were poor and substantially worse than those of McCarthy *et al.* (2004a). Furthermore, their method is limited in applicability by its reliance on the WordNet network structure.

Some other methods deviate wildly from the general paradigm of McCarthy *et al.* (2004a), which was based on aligning sense inventories to usage patterns of the target word from unlabelled corpora. For example, Chan and Ng (2006) presented a method for learning sense distributions from domain-specific corpora, given a supervised WSD system that has previously been trained on a labelled general domain corpus. This method is an extension of the previous method by the same authors (Chan and Ng 2005), which addressed the same task. Chan and Ng (2006) proposed a generative probabilistic model for the domain-specific corpora, based on the constraint that the probability of observing any usage given a fixed sense is the same as in the domain-independent corpus on which the WSD system was trained, and only the prior sense probabilities (that is, the sense frequencies) change in the new corpus. In this model, sense frequencies in the new domain appear as latent variables. Their method involves fitting this model to the domain-specific corpus using the EM algorithm, after which the domain-specific sense frequencies can be read off. Although this method relies on having a sense-labelled general domain corpus on which to train their supervised WSD model, and is therefore not completely unsupervised, they also proposed a semi-automatic method for creating this labelled data. This semi-automatic method is based on the work of Ng *et al.* (2003), and involves using parallel corpora,[12] as well as hand-annotated relationships between the senses and possible translation of each word. The results of their method were shown to be very promising; in particular, they were stronger than those from fully unsupervised methods such as that of McCarthy *et al.* (2004a). However, because of the use of labelled data, these methods are not directly comparable.

A different kind of approach again was taken by Brody *et al.* (2006), who presented an ensemble learning approach to first sense learning. They experimented with several ensemble learning methods to combine the output of the McCarthy *et al.* (2004a) method, along with the outputs of several unsupervised WSD systems, which were based on WordNet similarity (Banerjee and Pedersen 2003), lexical chaining (Galley and McKeown 2003), and graph-based (Navigli and Velardi 2005) approaches (these methods are all discussed in Section 2.2.2). Although they were able to improve first sense learning accuracy beyond what could be achieved by any of the individual methods, we do not view this as a competing method for large-scale sense distribution learning, due to its ensemble nature. Instead, we consider it as a method of using sense frequencies obtained from unsupervised all-words sense distribution learning, and combining them with other methods to obtain higher accuracy in applications.

[12]This means a corpus consisting of pairs of documents in two languages, such that the documents in each pair are translations of each other (Brown *et al.* 1988). Parallel corpora were first applied to WSD by Dagan *et al.* (1991).

Finally, Loukachevitch and Chetviorkin (2015) recently proposed a fully supervised method for first sense learning. Their method is designed around RuThes (Loukachevitch and Dobrov 2014), which is a Russian thesaurus with a graph-based structure vaguely similar to that of WordNet, but with relations between words based on concepts rather than lexical relationships (such as synonymy or hypernymy). They proposed a set of features that could be extracted from an input corpus for each target word, for example a feature for each sense of the target word based on the total count of words or phrases related to the sense (based on RuThes relations) within documents containing the target word. These features could be used to train a supervised machine learning model using gold-standard sense frequencies for a small set of words (the training data), which could then be generalised to predict sense frequencies for words outside this training set. They performed an evaluation of their method on a Russian news corpus annotated with sense frequencies, using several general-purpose supervised machine learning algorithms and a train/test split of the annotated words in the corpus. However, their results are difficult to relate to past work due to the different kind of sense inventory and datasets used.[13] In addition, while the supervised approach is novel and there may be some scope to use it on top of unsupervised methods,[14] the requirement of annotated sense frequency data is very limiting.

Out of these methods for unsupervised first sense and sense distribution learning, we believe the method most suitable for our aims of unsupervised all-words sense distribution learning, which can be applied generically across languages, is the HDP-WSI method of Lau *et al.* (2014). Firstly, it is the most widely applicable, because: (1) it does not require preprocessing such as parsing of the input corpus, which would limit the languages it could be applied to; (2) it can be applied to any sense inventory that contains glosses;[15] and (3) it doesn't require any kind of sense-labelled data, which is expensive to obtain. Secondly, it has been shown to perform strongly on WordNet— the sense inventory used in all cross-method comparisons that we are aware of — with performance in terms of first sense learning at least on par with competing methods, and performance in terms of overall sense distribution quality superior to others. Finally, because of the two-step process consisting of WSI followed by topic–sense alignment, the WSI results can be saved to provide an additional sense resource. This would allow trivial re-estimation of sense frequencies for new sense inventories, for example.

[13]For example, it is unclear how fine- or coarse-grained their sense inventory is, so accuracy numbers are difficult to interpret.

[14]For example, by using automatically learnt general domain sense frequencies from unsupervised learning as training data for domain adaptation.

[15]This allows application to resource-poor languages that only contain simple gloss-based sense inventories.

2.2.2 Unsupervised WSD

We now present a high-level review of past work on unsupervised WSD (excluding methods based on first sense learning, which have been thoroughly covered above). Although this is not quite the same problem as sense distribution or first sense learning, it is related in that an accurate unsupervised WSD method could infer sense frequencies by disambiguating usages over an entire corpus, and counting the number of usages disambiguated with each sense. Because there has been substantial work on this topic and it is only indirectly related to our aims, we do not delve into individual papers, but instead present a general overview of the subfield.

One of the oldest families of techniques is based on methods that calculate the similarity between senses, usually using WordNet. Examples of such sense similarity methods include those of Lesk (1986) and Banerjee and Pedersen (2003), which compare the word overlap between sense glosses, or that of Jiang and Conrath (1997), which computes similarity using the WordNet graph structure. Unsupervised WSD can be performed using these similarity methods on a per-usage basis, by comparing each possible sense of the target word to all possible senses of the surrounding context words.

Another major family of methods use structural or graph-based methods to simultaneously disambiguate all words within some context. This can involve performing all-words WSD at a per-sentence level (Navigli and Velardi 2005; Chen *et al.* 2014), or at a per-document level (Galley and McKeown 2003; Boyd-Graber *et al.* 2007). What these methods have in common is they make use of known relationships between senses to guide this simultaneous disambiguation. These relationships are often based on the WordNet graph (Galley and McKeown 2003; Navigli and Velardi 2005; Boyd-Graber *et al.* 2007) but can also be from other sources such as vector representations of senses (Chen *et al.* 2014). Disambiguation can be performed for all words at once within the context after fitting some probabilistic model (Boyd-Graber *et al.* 2007) or extracting relationships (Galley and McKeown 2003) within the context. Alternatively, it can be performed iteratively one word at a time, using the results of earlier steps to refine later disambiguation choices (Navigli and Velardi 2005; Chen *et al.* 2014).

A final kind of unsupervised approach involves fully- or semi-automatically extracting sense-labelled data. This can be done by automatically performing web searches (Agirre and Martinez 2004), or by making use of parallel corpora (Ng *et al.* 2003). These methods usually involve making some kind of constraining assumption to facilitate automatic extraction; for example, if a sense of some word has a monosemous[16] synonym then any usage of the word containing the synonym belongs to that sense (Agirre and Martinez 2004), or that there is a direct correspondence between the senses of a word in some primary language and its possible translations in a secondary language (Ng *et al.* 2003). Although these methods are mostly automatic,

[16]This means the word has only a single sense.

some small amount of annotation is sometimes required; for example, the method of Ng *et al.* (2003) requires that for each word in the primary language, all possible secondary language translations are labelled with their corresponding primary language sense. These methods allow WSD to be performed by automatically generating training data that can be fed into a generic supervised WSD system.

In addition to the fact that these methods do not address the exact problem we are concerned with, they are also limited in that they produce disambiguation results for a specific sense inventory. Therefore, if we were to adopt them for language-wide sense resource creation, we would only obtain frequency data for a single sense inventory. In contrast, the HDP-WSI method of Lau *et al.* (2014) produces data that can easily be re-aligned to multiple sense inventories. On the other hand, these unsupervised WSD methods could be seen as complementary to our aims; for example, they could be combined with the results of unsupervised all-words sense distribution learning, using ensemble methods such as that of Brody *et al.* (2006).

2.2.3 Word Sense Induction

We conclude our review of the relevant sense learning literature with a brief overview of the past work on word sense induction (WSI). WSI is the problem of automatically inducing sense inventories from data, which can then be used for disambiguation, as opposed to using fixed inventories like WordNet (Navigli 2012). WSI is related to our aims in that it could be applied to learn language-wide sense frequencies for an automatically induced sense inventory. In addition, it is relevant to discuss since our chosen sense learning method to build upon, HDP-WSI, is based on performing WSI (the results of which are then aligned to an existing sense inventory).

Most WSI methods involve some kind of clustering of word usages, where the clusters correspond to the automatically induced senses. A variety of different kinds of techniques have been employed to do this, including graph-based methods (Véronis 2004; Navigli and Crisafulli 2010), probabilistic methods (Brody and Lapata 2009; Choe and Charniak 2013; Yao and Van Durme 2011; Lau *et al.* 2012; Goyal and Hovy 2014), and spectral clustering (Goyal and Hovy 2014). In addition, word embedding-based methods (Chang *et al.* 2014; Neelakantan *et al.* 2014) have recently been employed, which also learn vector representations of the induced senses in the same space as word vectors.

An alternative approach involves the clustering of words themselves, in order to automatically induce a thesaurus-like sense repository. An example of this is the distributional similarity method of Lin (1998), which was used as part of the prototypical first sense learning method of McCarthy *et al.* (2004a). Using a word similarity method such as this, similar words can be grouped into overlapping clusters, and a sense inventory with similar structure to the Macquarie Thesaurus (discussed in Section 2.2.1) can be automatically induced.

Another kind of approach again involves building an inventory by extracting

glosses from web-based sources. For example, Faralli and Navigli (2012) propose a method for inducing a sense inventory by mining word–gloss pairs using automatically generated web search queries. Their method is semi-automatic, and bootstraps from a supplied list of domains and domain terms to create a hierarchical sense inventory, where the mined glosses for each term in the inventory are grouped by domain. This kind of hierarchical inventory supports both coarse-grained (domain-level) and fine-grained (gloss-level) disambiguation.

The WSI method employed by HDP-WSI is that of Lau *et al.* (2012), which follows a probabilistic clustering-based approach using topic modelling. Because it has been shown to achieve results competitive with the state-of-the-art in WSI, and it comes with a proven method to accurately align its results to existing sense inventories such as WordNet, we choose not to experiment with any of these other WSI methods. However, because HDP-WSI is based on HDP— which is a general-purpose topic modelling algorithm — it has the potential to be customised and improved by substituting HDP for other topic modelling algorithms. This motivates the next section, in which we provide a brief overview of some of the recent literature on topic modelling.

2.3 Topic Modelling

In the previous section we provided a thorough overview of the relevant sense learning literature, and justified our decision to build on HDP-WSI for our aims of language-wide sense distribution learning and resource creation. However, it is a modular method based on topic modelling, and thus could potentially be customised and tailored for our aims by replacing the HDP topic modelling algorithm. Given this, we provide a brief review of some of the recent literature on topic modelling.

In general, topic modelling refers to a family of probabilistic modelling methods, which model document collections using some kind of latent "topic" variables. These topic variables are shared between documents, and typically provide a kind of soft-clustering mixture model for the document collection. Perhaps the most prototypical topic modelling method is LDA (Blei *et al.* 2003), which is based on a directed graphical model, and uses a fixed number of topics that is decided as a hyperparameter. Subsequently HDP was proposed (Teh *et al.* 2006), which is a nonparametric extension of LDA based on Dirichlet processes (Ferguson 1973) that automatically learns the "right" number of topics to use. This algorithm provides the basis of HDP-WSI, and is discussed in more detail in Section 3.2.2.

Subsequent work on LDA-like topic modelling has followed a couple of different tracks. One direction has been the creation of more efficient inference algorithms. An example is the recent table indicator sampling algorithm (Chen *et al.* 2011), which is a Gibbs sampling algorithm that has been demonstrated to converge faster and more accurately than competing approaches. In addition, there has been work on improving the underlying probabilistic models. For example, Teh and Jordan (2010) extended

HDP by using Pitman-Yor processes (Pitman and Yor 1997) instead of Dirichlet processes, which is a better theoretical fit for language data due to Zipf's Law (Zipf 1936; Zipf 1949). Subsequently, Buntine and Mishra (2014) extended HDP further to create HCA,[17] by also introducing a burstiness (Doyle and Elkan 2009) component, which models a natural language phenomenon where words used at least once in a discourse are disproportionately more likely to be used subsequent times.[18] In addition, they make use of the table indicator sampling algorithm of Chen *et al.* (2011) for fast inference, although this comes at the cost of HCA using a fixed number of topics, like LDA.[19] However, they also challenge the wisdom that being nonparametric is important, and argue that as long as enough topics are used and the inference algorithm is accurate, the presence of extra junk topics will be benign.

In addition to the work on LDA-like topic modelling, there has been work on alternative topic modelling paradigms. One such approach has involved the incorporation of WordNet structure into the probabilistic graphical model, in order to explicitly model word senses (Boyd-Graber *et al.* 2007). However this was not very successful, and it did not accurately model WSD as hoped. Another kind of approach has been based on undirected graphical models, for example using models based on restricted Boltzman machines (Salakhutdinov and Hinton 2009; Larochelle and Lauly 2012) or Markov random fields (Xie *et al.* 2015). It is argued that these kinds of approaches better model discrete data such as text than LDA-style models, and they have achieved competitive results in terms of evaluation metrics such as perplexity (which is a measurement of how well the model fits data).

Of these topic modelling methods, we choose to experiment with the HCA method of Buntine and Mishra (2014). This is because it not only achieves strong results in terms of perplexity, but has been shown to be over an order magnitude faster than HDP, which is important for us given our aim to apply sense distribution learning on a very large (language-wide) scale. In addition, it is very similar to HDP— it follows the same LDA-like modelling approach — which means it can probably be substituted successfully without too much work. In contrast, the alternate undirected graphical model-based approaches are quite different, so they may not be as straightforward to apply in our sense distribution learning framework. In addition, unlike with HCA we could not access readily usable implementations of these methods. Given these reasons we elected to not experiment with them, and to instead leave such investigation to future work.

[17] See Section 3.2.2 for a more detailed description of HCA.

[18] This is true even after controlling for the topic of discourse.

[19] The table indicator sampling algorithm relies on the assumption of a fixed number of topics.

2.4 Multiword Expression Analysis

One major gap in the past work on sense learning methods is that, except for some minor exceptions discussed below, these methods have only been applied to simplex lemmas. In order to address this gap and include multiword expressions (MWEs) in our sense distribution resource creation, we need to extend our sense distribution learning methodology to MWEs. As discussed in Section 1.1, this introduces challenges such as identifying MWE usages from unlabelled text. To this end, we provide a brief overview of some of the relevant literature dealing with the identification and disambiguation of MWEs.

One strand of research has dealt with identifying usages of particular kinds of MWEs, based on expected patterns in how they are used. For example, Kim and Baldwin (2010) and McCarthy *et al.* (2003) both dealt with identifying usages of verb particle constructions, such as *stand up*, while Lapata and Lascarides (2003) addressed the identification of compound nouns. In addition, there has been some work on general-purpose MWE identification using supervised learning (Schneider *et al.* 2014). However, to the best of our knowledge there has been no work on general purpose unsupervised MWE identification, which is what we would need to apply sense distribution learning on a language-wide scale to all kinds of MWEs.

Another strand of research deals with disambiguation of MWEs between literal and idiomatic interpretations. As an example, the MWE *red herring* has a literal interpretation (a kind of fish that is red) and an idiomatic interpretation (a distraction). A range of different kinds of approaches have been used for this task, including supervised methods that are trained on labelled usages of the MWEs being disambiguated (Hashimoto and Kawahara 2008), supervised methods that are trained on a fixed set of MWEs but generalise to MWEs outside the training set (Fothergill and Baldwin 2012), or unsupervised methods (Fazly *et al.* 2009). Although idiomatic versus literal interpretations could be viewed as different senses, this is not quite the same problem as we are wanting to address, because a given MWE can have multiple different idiomatic or literal senses. For example, the MWE *old man* has a `father` sense and a `common wormwood` sense, which are both idiomatic.

There has also been some work on identifying and disambiguating named entities (for example names or places) (Yosef *et al.* 2011; Moro *et al.* 2014). These often consist of multiple words, and therefore could be viewed as a kind of MWE; for example, we may wish to determine whether *San Francisco* refers to the United States city or one of the many other cities with the same name. However, this again does not solve the problem we are interested in, since methods for named entity recognition make use of rich knowledge sources and very specific kinds of information not available for general WSD.

Finally, very recently there has been some work on supersense tagging of MWEs,

as set out by Task 10 of SemEval-2016 (Schneider *et al.* 2016).[20] This task in part involves extracting MWEs[21] from text (which the authors of the task describe as segmenting the text into "minimal semantic units"), and labelling occurrences of these MWEs with their correct supersense. Supersenses are an extremely broad version of senses; the task involves 26 different noun supersenses in total (such as `person`, `location`, or `time`), and 15 different verb supersenses (such as `motion` or `communication`), which are shared between all nouns and verbs respectively. However, because the distinctions between supersenses are extremely coarse-grained compared to senses (even compared to relatively coarse-grained sense inventories), and the set of supersenses doesn't contain features standard to regular sense inventories such as glosses, this is a very different task. Therefore, it is reasonable to believe that methods for supersense disambiguation will not translate to general WSD.

In summary, although there is some relevant work on MWEs that could be leveraged to address our sense learning aims — both with regard to disambiguating between MWE senses, and identifying usages of MWEs in unlabelled text – this work either deals with very specific kinds of MWEs and sense distinctions, or requires labelled data. Therefore, given that we want to work with unlabelled data to learn senses across all kinds of words, we can conclude that there are no existing methods for either of these tasks that satisfy our requirements.

2.5 Summary

We have provided an overview of the existing work relevant to our aim of language-wide sense distribution learning. In Section 2.2 we provided a thorough overview of the work on sense learning, focussing mostly on existing methods for first sense or sense distribution learning, and identified HDP-WSI as an appropriate method to build on top of. This is because it has achieved state-of-the-art results in sense distribution learning, it is widely applicable across languages and to any sense inventory containing glosses, and it also produces WSI output that is competitive with the state-of-the-art. In addition, it is modular and can be easily customised by replacing its topic modelling component using other topic modelling algorithms. Then in Section 2.3 we provided an overview of some of the recent work on topic modelling, where we identified HCA a promising method that could possibly be used in place of HDP in HDP-WSI to better tailor it for our aims, particularly in terms of reducing computation time. Finally, given that we wish to apply our sense distribution learning to MWEs as well as simplex lemmas, in Section 2.4 we provided an overview of the past work on identifying and disambiguating MWEs, where we concluded that none of the existing

[20] At the time of the submission of this thesis, the proceedings containing solutions to this task have not yet been published.

[21] This means learning new MWEs, as opposed to MWE identification, which means recognising occurrences of known MWEs.

methods are appropriate for our aims.

In the literature that we have reviewed, there are a few obvious gaps that we aim to address with our research. One is that although there has been extensive work on sense distribution learning and related problems, none of this work has addressed the problem of scalability, and efficiently applying this learning on a language-wide scale. Also, there has been no work in applying any kind of general-purpose sense learning to MWEs; all past work on disambiguating MWEs has dealt with very specific kinds of distinctions (such as between different named entities, between idiomatic and literal interpretations, or between different supersense classes). Furthermore, applying unsupervised all-words sense distribution learning to MWEs introduces the additional challenge of identifying MWE usages for all kinds of words without labelled data, which has not been adequately addressed in past work.

Now that we have analysed the existing literature with regard to how it relates to our research aims, and have identified appropriate methods from this literature that can be used to address these aims, the next step is to detail the existing resources and methods from past work that we use in our experiments. This will then provide the foundation to describe these experiments. In the next chapter we provide a thorough description of these background resources and methods.

Chapter 3

Background

3.1 Resources and Tools

3.1.1 Introduction

In this section we provide a detailed description of the resources used in our investigation. This includes: (1) sense resources, including sense inventories and sense frequency resources; (2) text resources, which are used to train our unsupervised learning methods; (3) NLP resources, which are used to perform various kinds of text-processing tasks as minor components of our experiments; (4) crowdsourced annotation resources, which are used to efficiently obtain high-quality annotated data for the purpose of evaluating our methods; and (5) cloud computing resources, which we employ to facilitate sense learning on a language-wide scale. We proceed with an overview of each of these resource categories in turn.

3.1.2 Sense Resources

WordNet

The sense inventory used for all of our English experiments is Princeton WordNet (Fellbaum 1998) (or simply WordNet for brevity).[1] The structure of WordNet is based on synsets, where each synset corresponds to a set synonyms. Specifically, each sense of any given lemma corresponds to a separate synset, which contains that lemma and all other lemmas that are synonyms with respect to that sense. For example, the `banking company` sense of *bank* in WordNet corresponds to a synset containing the lemmas *bank*, *depository financial institution*, *banking concern* and *banking company*. These synsets define the synonym relationship in WordNet: lemma l_1 is a synonym of lemma l_2 if and only if they belong to a common synset.

[1]Note that when we refer to WordNet in this thesis without qualification, we are referring to Princeton WordNet. Furthermore, except where stated otherwise, we are referring to version 3.0.

Language	Sense Inventory
Japanese	Japanese WordNet (Isahara *et al.* 2008)
Italian	MultiWordNet (Pianta *et al.* 2002)
Mandarin	Chinese Open WordNet (Wang and Bond 2013)
Indonesian	WordNet Bahasa (Mohamed Noor *et al.* 2011)

Table 3.1: Summary of the non-English WordNet sense inventories used from Open Multilingual WordNet.

These synsets are organised in a network based on different kinds of lexical relationships. An example of such a relationship is hypernymy: s_1 is an hypernym of s_2 if and only if s_1 is a generalisation of s_2. Correspondingly, a lemma l_1 is a hypernym of lemma l_2 if and only if l_1 belongs to a synset that is a hypernym of some synset of l_2. For example, *bank* is a hypernym of *commercial bank*, because a commercial bank is a kind of bank (so *bank* is a generalisation of *commercial bank*).[2] In addition to hypernymy there are several other lexical relationships in WordNet, including hyponymy (which is the reverse of hypernymy). However, the only relationships we use in this thesis are synonymy and hypernymy, so we do not discuss these other relationships any further.

Sense distinctions in WordNet are very fine grained, so sense learning with this inventory is a difficult task (Palmer *et al.* 2007; Hovy *et al.* 2006). For example, the lemma *bank* has separate `banking company` and `bank building` senses, even though arguably each of these is replaceable with a single `financial bank` sense. Indeed, the lemma *bank* contains 10 separate noun senses in total, and an additional 8 separate verb senses!

Open Multilingual WordNet

The Japanese, Italian, Mandarin, and Indonesian sense inventories used to create the non-English data in LexSemTm (in Section 5.2) all come from Open Multilingual WordNet (OMW: Bond and Paik (2012)). OMW is a collection of sense inventories covering many languages, all under open-source licences. These inventories have the same network-based structure as Princeton WordNet, and therefore are referred to as WordNet's. The specific sense inventories we use from OMW are summarised in Table 3.1. These non-English WordNet's were accessed using the Natural Language Toolkit (NLTK: Bird *et al.* (2009)).

[2]Technically hypernymy is usually defined as a transitive relation, meaning if s_1 is a hypernym of s_2 and s_2 is a hypernym of s_3, then s_1 is a hypernym of s_3. However, in all cases where it is used in this thesis we refer to immediate hypernymy, meaning there is a direct "is-a" relationship between the two synsets or lemmas.

SemCor

SemCor (Miller *et al.* 1993) is a 220,000 word corpus manually annotated with WordNet sense tags, which accompanies WordNet. It is based on a subset of the Brown Corpus (Kucera and Francis 1967), which is a balanced corpus of mid-20th century American literature. SemCor is often treated as the de-facto standard source of sense frequencies: SemCor sense frequencies are often used for the purposes of unsupervised WSD with the MFS heuristic, which is frequently employed as a strong benchmark in the WSD and sense distribution learning literature (for example in McCarthy *et al.* (2004a), Lapata and Keller (2007), and Bhingardive *et al.* (2015)). Furthermore, WordNet lists SemCor frequencies with each sense, and orders its senses in descending order based on these frequencies.

However as discussed in Section 1.1, SemCor leaves much to be desired as a source of sense frequencies, due to the lack of annotation coverage (most lemmas have no or very few sense annotations), the age of the data (language usage has changed significantly since the mid-20th century), and inconsistencies because of the source of the data (the Brown Corpus is based on literature, which often employs unusual word senses that do not reflect everyday language use, and also fails to cover many topics, for example topics related to science).

Because of its status as the de-facto standard for WordNet sense frequencies, we use SemCor extensively in our experiments as a sense distribution benchmark, and in Section 5.3 we investigate to what extent our LexSemTm sense distributions can supplement or replace SemCor.

3.1.3 Text Resources

BNC

The first corpus used in our experiments is the British National Corpus (BNC: Burnard (1995)), which is a balanced English corpus. This corpus was previously utilised by Koeling *et al.* (2005), who used it to evaluate the first sense learning method of McCarthy *et al.* (2004a). They introduced a set of 40 English nouns (which we refer to as L_{bnc}), and annotated a subset of the BNC usages for each of these nouns with WordNet 1.7[3] senses. From these annotations we can obtain gold-standard sense distributions for each lemma in L_{bnc} (by maximum likelihood estimation), which can be used to evaluate candidate sense distributions for L_{bnc} lemmas using the metrics discussed in Section 3.2.4.

We use this corpus extensively in our experiments in Chapter 4 in order to optimise sense distribution learning for language-wide applications. In these experiments we use the same same set of usages for each lemma in L_{bnc} as training data, as in prior

[3]Note that because the corpus is tagged with WordNet 1.7 rather than WordNet 3.0 senses, all experiments we performed using the BNC corpus (as well as the Sports and Finance corpora discussed below) were done using WordNet 1.7.

work with this corpus (including that of Koeling *et al.* (2005) and Lau *et al.* (2014)), which had previously been preprocessed using tokenisation, POS-tagging, and lemmatisation. These preprocessing steps are described in more detail in Section 3.1.4.

Sports and Finance

In addition to the domain-neutral BNC corpus, Koeling *et al.* (2005) also produced gold-standard WordNet 1.7 sense annotations for the lemmas in L_{bnc} for two domain-specific corpora. These corpora, which we refer to as Sports and Finance, were obtained from the larger Reuters corpus (Rose *et al.* 2002). The Reuters corpus consists of documents from a variety of domains, and these Sports and Finance corpora were obtained by selecting the documents from the Reuters corpus in the sports and finance domains respectively.

We use this corpus in addition to BNC for some of our optimising experiments in Chapter 4, in cases where we wish to be more sure of our conclusions and confirm them over a wider range of data. These corpora were also used by Lau *et al.* (2014) in their evaluation of HDP-WSI, along with BNC. As with BNC, the usages in the corpus had previously been preprocessed using tokenisation, POS-tagging, and lemmatisation.

EnWiki

For the purposes of performing language-wide sense distribution learning and producing LexSemTm (see Section 5.2), which is our attempt to supplement or replace SemCor, we use corpora based on Wikipedia. This was chosen because of its ready availability, its wide coverage over many topics, and because its language use is relatively formal and of high quality.

The specific corpus we use for English, which we refer to as EnWiki, is based on an English Wikipedia dump dated 2009-11-28.[4] This dump had previously been tokenised and POS-tagged (using the maximum entropy model) with OpenNLP.[5] In addition we performed lemmatisation, giving two versions of each section: a surface (unlemmatised) version, and a lemmatised version.

MultiLingualWiki

Analogous to EnWiki, the corpus we use for producing the non-English data in LexSemTm is also based on Wikipedia. We refer to this corpus as MultiLingualWiki, which covers 4 languages: Japanese, Italian, Mandarin, and Indonesian. This

[4]We this dump because it was readily available and had already been POS-tagged, which is a computationally expensive process.

[5]`https://opennlp.apache.org/`

corpus was mined from publicly available Wikipedia text dumps for each language,[6] and from each dump text was extracted using WikiExtractor with default settings.[7] As with EnWiki, the text had previously been preprocessed using tokenisation and POS-tagging. Preprocessing was performed using MeCab for Japanese (Kudo *et al.* 2004), Freeling for Italian (Padr *et al.* 2010), Stanford tools for Mandarin (Tseng *et al.* 2005; Chang *et al.* 2008), and the process built in the creation of the NTU-MC for Indonesian (Tan and Bond 2014).

3.1.4 NLP Resources

Tokenisation

Tokenisation is the process of breaking text into individual "tokens", which can then be used as the input for text algorithms such as topic modelling. This is mostly a simple problem for English (except for dealing with punctuation), but is more challenging for languages such as Japanese or Chinese where text is not conventionally broken up into separated words. Tokenisation in our experiments is performed using the same processes for each language used in the creation of the EnWiki and MultiLingualWiki corpora (see Section 3.1.3).

POS-tagging

POS-tagging is the process of labelling each token in some text with its part of speech (POS), such as noun or verb. Again, this is done using the same methods for each language used to create EnWiki and MultiLingualWiki (see Section 3.1.3).

The sets of POS-tags used for each language are based on the Penn Treebank for English (Marcus *et al.* 1993), IPAdic for Japanese (Asahara and Matsumoto 2003), Penn Chinese Treebank for Mandarin (Xue *et al.* 2005), and the Bahasa POS-tags used by Pisceldo *et al.* (2009) for Indonesian. The actual POS-tags in each of these sets are more complex than noun or verb, for example. However, for the purposes of our experiments, where we match the POS-tags of tokens with those of our WordNet and OMW lemmas, we manually mapped the POS-tags in each of these sets to "noun", "verb", "adjective", "adverb", or "other".

Lemmatisation

Lemmatisation is the process of reducing words in text to their dictionary form, so that text containing different inflected forms of the same word can be compared.

[6] The specific Wikipedia text dumps used for each language were `jawiki` for Japanese, `itwiki` for Italian, `zhwiki` for Mandarin, and `idwiki` for Indonesian. All of these were accessed in June, 2014.

[7] `http://medialab.di.unipi.it/wiki/Wikipedia_Extractor`

For example, lemmatisation would reduce both *throwing* and *threw* to *throw*. Lemmatisation was used in our experiments with English data, and was performed using Morpha (Minnen *et al.* 2001).

Stopword Removal

In order to reduce the impact of stopwords, which are highly frequent, non content bearing words (such as *and* or *at*), we removed these from text in most of our experiments. This process is called stopword removal, and is performed by removing every word contained in a supplied stopword list. Stopword removal is applied to every lemma usage and sense gloss in order to perform HDP-WSI, as described in Section 3.2.3.

We produced stopword lists for each language by merging multiple freely available stopword lists for each language found online, and expanding these with all words from the respective corpora accounting for over 0.1% of the total words in the corpus. Finally these lists were pruned manually by removing all words that we deemed to be obviously content bearing (for example *school* for English). We erred on the side of using fairly large stopword lists, because stopword removal has the added benefit of reducing the computation time of topic modelling, which is useful given our aim of large-scale application of sense distribution learning.

3.1.5 Crowdsourced Annotation Resources

Amazon Mechanical Turk

The main resource we used to obtain annotated data, for the purposes of evaluating our LexSemTm sense distributions in Section 5.3 and Section 5.4, was Amazon Mechanical Turk (AMT). AMT is a platform for crowdsourcing "human intelligence" tasks, where workers are asked to annotate some kind of data. AMT has been extensively used to produce annotated data for NLP research (Callison-Burch and Dredze 2010), and previous work has demonstrated that it can be used to produce annotated data equal in quality to that from expert human annotators, using a small number of non-expert AMT workers per annotation item (Snow *et al.* 2008).

The workflow of AMT consists of providing an HTML template for the entire task, as well as a number of batches to be annotated, where each batch consists of a set of items to be annotated, along with the data to fill the HTML template and generate the web form that is supplied to workers who annotate these items. An annotation task can be customised by specifying the number of annotations required for each batch of items, as well as minimum quality requirements of workers. In all of our experiments using AMT, we set a minimum requirement of a 95% lifetime approval rating for annotation tasks, in order to ensure decent quality workers.

MACE

MACE (Hovy *et al.* 2013) is a tool for analysing the output of multi-item, multi-annotator annotation tasks such as from Amazon Mechanical Turk. It assumes that we have a pool of items to be annotated and a pool of annotators, that each item has been annotated by one or more annotators, and that annotation values are discrete and global (shared between all items). MACE works by providing a probabilistic framework for modelling annotator quality and annotator bias in such a scenario, which can be fit to data. Furthermore, it allows some of the items to be used as controls by providing the "correct" annotations, which can help guide the fitting of the model (in other words, it allows for semi-supervised learning).

The input of MACE consists of a matrix of annotations, where rows correspond to items to be annotated, columns correspond to annotators, and each cell corresponds to the annotation value of the given item by the given annotator (possibly empty). Annotation values are assumed to be categorical (provided in the input as integers). The output of MACE consists of its estimate of the true label of each item.[8] In all experiments where we use MACE, we run it using variational Bayes training under otherwise default settings.

3.1.6 Cloud Computing Resources

Google Compute Engine

We used the Google Compute Engine (GCE)[9] for running our large-scale, sense distribution learning experiment to create LexSemTm (which is discussed in Section 5.2). GCE is a cloud computing service that allows virtual machines (VMs) of different specifications to be booted up, used for computation, and shut down on command. This service makes our large-scale learning tractable, which would otherwise have taken several years of computation if run on a single-core machine, for example. Sense distribution learning using GCE was done using 640 VMs,[10] with a separate batch of lemmas processed per VM.[11]

3.1.7 Summary

We have presented a detailed summary of the key resources and tools used in our experiments. In the next section we present the methods from past work that

[8] In addition it provides estimates of the quality of each annotator, however we do not use this in our experiments.

[9] `https://cloud.google.com/compute/`

[10] These were n1-highmem-16 VMs (16 virtual cores and 104GB RAM) running Ubuntu 14.04, with 30GB of local persistent disk (non-SSD) each.

[11] Computation on each VM was performed in parallel using 16 processes and a producer-consumer architecture.

we use, which together with these resources and tools provide the backbone of our experimental methodology.

3.2 Methodology

3.2.1 Introduction

We now provide a detailed overview of the methodology from past work, which provides the remaining foundation for our experimental design. We first describe the topic modelling methods — HDP and HCA— which form the cornerstone of our sense distribution learning. Next we build on our discussion of topic modelling to describe the HDP-WSI method of Lau *et al.* (2014) for sense distribution learning, which we extend and optimise for large-scale application in Chapter 4. Finally, we provide a quick description of the existing evaluation metrics for sense distributions, which we use to measure sense distribution quality in our experiments.

3.2.2 Topic Modelling

HDP

HDP (Teh *et al.* 2006), which was introduced in Section 2.3, is a nonparametric topic modelling method for modelling document collections. In this model each document has its own distribution over a set of latent "topic" variables, and each topic has its own distribution over words. Given these distributions, each word in a given document is assumed to have been generated independently, by first sampling a topic from the document's distribution over topics, and then sampling a word from that topic's distribution over words. Because these latent topics are shared between documents, this model could be viewed as a probabilistic mixture model. HDP is described as nonparametric because it automatically learns the "right" number of topics to use.[12]

HDP is a Bayesian model, which means it provides a probabilistic model for how the document distributions over topics and the topic distributions over words are generated, as well as the words themselves. This model is defined in terms of Dirichlet processes (Ferguson 1973), which can be understood as infinite dimensional analogues of the Dirichlet distribution. Given a base probability distribution as input, the Dirichlet process randomly generates a new probability distribution over the same support as the input distribution, by sampling a countably infinite number of observations from it. Formally, HDP is based on a hierarchy of Dirichlet processes: a parent

[12]Technically it uses a countably infinite number of topics. However, in practice only a finite number of these will be assigned to words during inference, and we interpret this as the number of topics automatically learnt.

Dirichlet process is run to generate the distributions over words for each topic,[13] and a child Dirichlet process is run for each document with the parent process as its base distribution, to generate the distributions over topics for each document. In order to be tractable, and overcome the fact that an infinite number of latent variables are involved, this model based on a hierarchy of Dirichlet processes is commonly redefined in terms of a stochastic process called the Chinese Restaurant Process (CRP: Aldous (1985)), which is mathematically equivalent. In addition to the topic variables, the CRP uses an additional set of latent variables called "table" variables. The exact mathematical details of hierarchical Dirichlet processes and the CRP — including the interpretation of the table variables and how they are used — are beyond the scope of this thesis.

Inference is usually performed for HDP, given an input document collection, using Gibbs sampling based on the CRP. Given an initial random assignment of topic and table variables to the words in the document collection, Gibbs sampling continuously re-samples the topic and table assignments of each word in the document collection. In each iteration of the algorithm, the topic and table assignments of every word are re-sampled one at a time, which is done using the conditional CRP-based table and topic probabilities, given all of the other topic and table assignments and the observed words.

The idea of Gibbs sampling is that the topic allocations at an arbitrary iteration (after an initial burn in period) will follow their true conditional distribution, given the observed data (the words in the corpora). Therefore we can run Gibbs sampling for a large number of iterations, and the iteration with maximum likelihood (according to the CRP probabilities of all topic and table assignments) will give an approximate MAP estimate of the "true" topic allocations.

Given this process, the training time of HDP scales roughly linearly in both the number of documents provided as input and the average document length, since the amount of computation performed during each iteration of Gibbs sampling increases roughly linearly in the total number of tokens across all documents. Similarly, the training time of HDP also scales roughly linearly in the number of Gibbs sampling iterations used.

In the workflow of HDP from our perspective, we first provide the tokens (words) making up each document in the collection as input (the order is unimportant). Then after training for a set number of iterations using Gibbs sampling, the output is the final topic allocation for each word in every document, from the maximum likelihood iteration. Topic allocation counts per document and word allocation counts per topic[14] can then be normalised to provide document distributions over topics and topic distributions over words respectively.

[13] Using a Dirichlet distribution as its base distribution to generate each of these distributions over words.

[14] This is the inverse of topic allocation counts per word type.

HCA

HCA (Buntine and Mishra 2014) is a topic modelling method that is an extension of HDP. It differs from HDP in four important ways: (1) it is parametric, so the number of topics to be used must be set as a hyperparameter; (2) it uses a much more efficient implementation of Gibbs sampling; (3) it uses Pitman-Yor processes (Pitman and Yor 1997) in addition to Dirichlet processes; and (4) the model includes a burstiness component.

HCA uses an algorithm for Gibbs sampling called "table indicator sampling" (Chen *et al.* 2011) — the precise details of which are beyond the scope of this thesis — which introduces an additional set of latent variables called "table indicator" variables. The CRP can be represented using these variables in a way that allows Gibbs sampling to be performed very efficiently — topics and table indicators can be sampled simultaneously, without the need to sample tables — which the authors find to converge much faster and more accurately than competing methods. However, this algorithm requires that a fixed number of topics must be set as a hyperparameter, and its computational complexity is roughly linear in the number of topics.

In addition, the probabilistic model of HDP is based on Dirichlet processes, whereas the model of HCA also uses Pitman-Yor process, which are a generalisation of Dirichlet processes that are better able to model natural language phenomena (Goldwater *et al.* 2011). By default HCA uses a Pitman-Yor process for its topic distributions over words (the parent process in the hierarchy), and Dirichlet processes for its document distributions over topics (the child processes in the hierarchy), but it can optionally be configured to use Pitman-Yor processes for both.

The probabilistic model is also extended compared to that of HDP by modelling burstiness (Doyle and Elkan 2009), which is a phenomena where words occurring in a discourse at least once are disproportionately more likely to occur additional times (beyond what could be explained by the topic of discourse). This is achieved by allowing every document to have its own specialised distribution over words for each topic, obtained by applying a Pitman-Yor process to the general distribution over words for each topic. Although this appears to introduce many more variables to the model, in practice they lead to minimal overhead in Gibbs sampling using table indicator sampling. This extension can optionally be turned either on or off.

However, despite these differences in the underlying probabilistic model and inference method, the overall workflow of HCA is the same as that of HDP from our perspective. That is, we provide a set of unlabelled documents to train on, and obtain document distributions over topics and topic distributions over words. Also as with HDP, the training time of HCA scales roughly linearly in the number of documents provided, and the number of Gibbs sampling iterations.

Sentence	After their extensive financial reforms, the big **banks** had finally become competitive. However despite this public sentiment had failed to improve.
Tokens	extensive financial reform big bank finally competitive public sentiment fail improve financial_#-3 reform_#-2 big_#-1 finally_#1 become_#2 competitive_#3

Table 3.2: An example of a lemma usage document, for the lemma *bank*. The original usage before processing is listed, as well as the post-processing tokens — including local context tokens — that are used as the input document for HDP.

3.2.3 HDP-WSI

HDP-WSI is a method proposed by Lau *et al.* (2014) for unsupervised sense distribution learning, which is built on top of the WSI method of Lau *et al.* (2012) (these methods were introduced briefly in Section 2.2.1 and Section 2.2.3 respectively). The method can be applied with any sense inventory containing glosses, and consists of two phases: (1) WSI is performed using HDP topic modelling; and (2) the results of WSI are aligned to the provided sense inventory. This two-step process is applied separately for each lemma to be processed.

WSI Phase

In the WSI phase of HDP-WSI, WSI is performed using HDP topic modelling (see Section 3.2.2). The input of this phase is a collection of usages of the target lemma,[15] and the output is a set of topics represented by distributions over words, a distribution over these topics for each document, and a global probability distribution over these topics.

First HDP is run on the collection of lemma usages, by treating each usage as a separate document. Each usage is processed by tokenisation, stopword removal, and also optionally lemmatisation (see Section 3.1.4 for details on these preprocessing steps), and extra local-context tokens are added for all tokens within a distance of 3 from the target lemma. An example of an input document is shown in Table 3.2.

Running HDP on this document collection produces a set of topics and distributions over words for each topic, which represent the automatically induced senses of WSI. In addition, a global probability distribution over these topics is calculated by labelling each document with a single topic (the topic with maximum probability for the document), and applying maximum likelihood estimation to the resultant topic counts.

[15] A lemma usage corresponds to the sentence containing the lemma, and the two neighbouring sentences (without crossing paragraph or section boundaries), except where stated otherwise.

Topic–Sense Alignment Phase

In the topic–sense alignment phase of HDP-WSI, the topics produced from the WSI phase are aligned to the provided sense inventory. In order to do this, a distribution over words is created from the gloss of each sense. Each gloss is processed identically to the lemma usages (by tokenisation, stopword removal, and optionally lemmatisation), and the resultant token counts are converted into a probability distribution by maximum likelihood estimation. Based on these gloss distributions and the results of WSI, a prevalence score is calculated for each candidate sense, s_i, according to:

$$\text{prevalence}(s_i) = \sum_{j=1}^{T}(\text{sim}(s_i, t_j) \times P(t_j)) \tag{3.1}$$

where T is the total number of topics, t_j is the j'th topic, $P(t_j)$ is the probability of t_j according to the global distribution over topics from WSI, and $\text{sim}(s_i, t_j)$ is the similarity between s_i and t_j. Similarity between senses and topics is calculated according to:

$$\text{sim}(s_i, t_j) = 1 - \text{JSD}(s_i||t_j) \tag{3.2}$$

where $\text{JSD}(s_i||t_j)$ is the Jensen Shannon divergence (described in Section 3.2.4) between the gloss distribution of sense s_i and the distribution over words of topic j_j.

Finally, the prevalence scores for each candidate sense are normalised to produce a distribution over these senses.

3.2.4 Sense Distribution Evaluation Metrics

JSD

The first metric we use for evaluating sense distribution quality is Jensen Shannon divergence (JSD). JSD is a measure of divergence between two probability distributions, taking values between 0 (distributions are identical) and 1 (distributions have no overlap). Given a candidate sense distribution and a gold-standard distribution for the same lemma, this metric is equal to the JSD between the candidate and gold-standard distributions.

The purpose of this metric is to evaluate the entire shape of the distribution — including the ranking of senses and the entropy of the distribution — not just the accuracy of the first sense prediction. JSD was first used for evaluating sense distribution quality by Lau *et al.* (2014).

ERR

The second metric we use for evaluating sense distribution quality is the error rate reduction (ERR) of MFS-based WSD, as was also used by Lau *et al.* (2014). Given a candidate sense distribution and a gold-standard distribution for the same lemma, ERR is calculated according to:

$$\text{ERR} = 1 - \frac{\text{Acc}_{\text{UB}} - \text{Acc}}{\text{Acc}_{\text{UB}}} = \frac{\text{Acc}}{\text{Acc}_{\text{UB}}} \tag{3.3}$$

where Acc is the WSD accuracy using the MFS heuristic with the candidate sense distribution,[16] and Acc_{UB} is the upper bound accuracy obtained by using the MFS from the gold-standard distribution.

The purpose of this metric is to evaluate the quality of sense distributions for the specific purpose of unsupervised WSD using the MFS heuristic, where all that matters is the quality of the first sense prediction.

3.2.5 Summary

In this section we have provided a detailed description of the methods we are adopting from past work. This includes two different methods for topic modelling (HDP and HCA), the HDP-WSI method for sense distribution learning, and the JSD and ERR metrics for evaluating sense distribution quality. These methods provide the foundation for our investigation into language-wide sense distribution learning, and in the next chapter we use and extend them in order to explore how to optimise HDP-WSI for large-scale application.

[16] If there is a tie for the MFS, the first-listed tied sense in the sense inventory is chosen.

Chapter 4

Optimising Sense Distribution Learning

4.1 Introduction

In Chapter 2 we explored the literature on sense distribution learning, and identified HDP-WSI as an appropriate method to build on top of. In addition, since HDP-WSI is implemented using HDP topic modelling, we explored the literature on topic modelling and identified HCA as a possible alternative to HDP. This was in major part due to its computational efficiency, which is important for large-scale application. Then in Chapter 3 we described how these methods work in detail, and presented a selection of resources and evaluation metrics that we can use and build upon to experiment with these methods.

In this chapter we present a series of experiments that build on this past work, in order to address our first core research question, regarding how to extend and optimise HDP-WSI for application on a language-wide scale. First in Section 4.2 we explore the convergence properties of sense distribution learning, which is necessary for large-scale application, since we need to know how few lemma usages we can use to achieve an optimal quality versus efficiency tradeoff. Next, in Section 4.3 we explore the use of HCA rather than HDP in HDP-WSI, in order to investigate whether this can result in an improvement in either computational efficiency or sense distribution quality. Finally, in Section 4.4 we explore a novel variation of HDP-WSI, which tries to achieve computation savings — as well as potentially an improvement in sense distribution quality due to statistical sharing — by applying the topic modelling part of sense distribution learning to clusters of multiple lemmas at once.

4.2 HDP-WSI Convergence Experiments

4.2.1 Introduction

In this section we present our experiments exploring the convergence of HDP-WSI, with respect to the number of lemma usages. In these experiments, we are seeking to answer two core questions: (1) how many usages are needed for HDP-WSI to converge? and (2) does the previous answer vary systematically for different kinds of lemmas? Because the computational complexity of topic modelling is roughly linear in the number of documents (which corresponds to the number of usages in our context), knowing how few usages we can get away with is important for large-scale application. Furthermore, if there are systematic differences in the quantity of training data needed for different kinds of lemmas, we could take advantage of this to further optimise large-scale learning.

In exploring the convergence of HDP-WSI, we are interested in the convergence of both the mean and variance of sense distribution quality; we want our optimised sense distribution learning method to be able to produce stable results that have converged in expectation, and also have little volatility. In addition to this, we are interested in exploring the convergence in terms of the number of topics produced by HDP, as this will inform our experiments with HCA (which uses a fixed number of topics).

4.2.2 Experimental Setup

In order to explore HDP-WSI convergence with respect to the number of lemma usages, we ran HDP-WSI on a large number of random subsets of the usages from the BNC corpus of the corresponding L_{bnc} lemmas (see Section 3.1.3). For each of the 40 lemmas in L_{bnc}, we created a large number of sense distributions using random subsets of the lemma's usages.[1] Each distribution was generated by first randomly choosing a number of usages to train on,[2] and randomly sampling that many usages without replacement. Then each distribution was created by running HDP-WSI on the sampled usages, and was evaluated relative to the lemma's gold-standard sense distribution from the BNC corpus, using the JSD and ERR metrics (see Section 3.2.4). Finally, the overall results for each lemma were partitioned into 40 bins of approximately equal size, according to the number of usages sampled.

By applying HDP-WSI to a large number of random subsets of the usages for each lemma, we have produced bootstrapped data that allows the mean and standard deviation of the evaluation metrics to be measured as a function of the number of usages, by calculating these metrics within each bin. Although this is non-standard bootstrapping because we sampled usages without replacement, we did this because: (1) it reflects the actual application scenario of sense distribution learning, where we

[1] Approximately 810 random sense distributions were created per lemma.

[2] Between 500 and the maximum number of usages available, sampled uniformly.

have to sample from a finite set of available usages (from some corpora); and (2) it produces more precise estimates of the mean metric values due to lower variance. In addition to allowing us to measure the convergence of the mean and standard deviation of our evaluation metrics, this bootstrapped data allows us to produce statistics on the number of topics produced by HDP, and how this varies as a function of the number of lemma usages.

4.2.3 Results

Given the two sense distribution quality metrics (JSD and ERR), and the two statistics of these metrics in each bin (mean and standard deviation), we have four sets of core results. These results are displayed in Figure 4.1, wherein for each combination of metric and bin statistic we plot one line per lemma (from one data point per bin).

In addition, results in terms of the number of HDP topics are displayed in Figure 4.2 and Figure 4.3. The former figure was created by binning all HDP topic models encountered during bootstrapping, according to the number of lemma usages (pooling all lemmas together): for each bin we created a boxplot, based on the distribution of the number of HDP topics for the topic models in the bin. The latter figure was created by plotting the convergence of the number of HDP topics on a per lemma basis: we used the same binning strategy as for JSD and ERR, and the average number of topics in each bin was plotted. In order to make these results readable, we plotted separate results for lemmas with fewer than 5,000 usages, and those with at least 5,000 usages.

4.2.4 Discussion

First looking at the results in terms of JSD in Figure 4.1, there is an apparent overall trend that around 5,000 to 10,000 usages are necessary for results to be stable and converged. This is true both in terms of the mean and standard deviation of JSD. In addition, we inspected the data manually on a per-lemma basis and found no outliers.

From the point of view of ERR, the results are messy and less clear. This is likely because ERR is a very discontinuous metric; its value changes when the mode of the sense distribution changes, and therefore it can change substantially from minor changes in the sense distribution, or can fail to change at all from large changes in the sense distribution. As with JSD it appears that the results become much more stable after around 5,000 to 10,000 usages, and furthermore in the case of ERR all variance disappears after approximately 15,000 usages.

In terms of the number of topics, it is clear from Figure 4.2 that although there is high variance when fewer than 5,000 usages are used (with up to 17 topics created in a small number of instances), the number of topics used quickly decreases as the

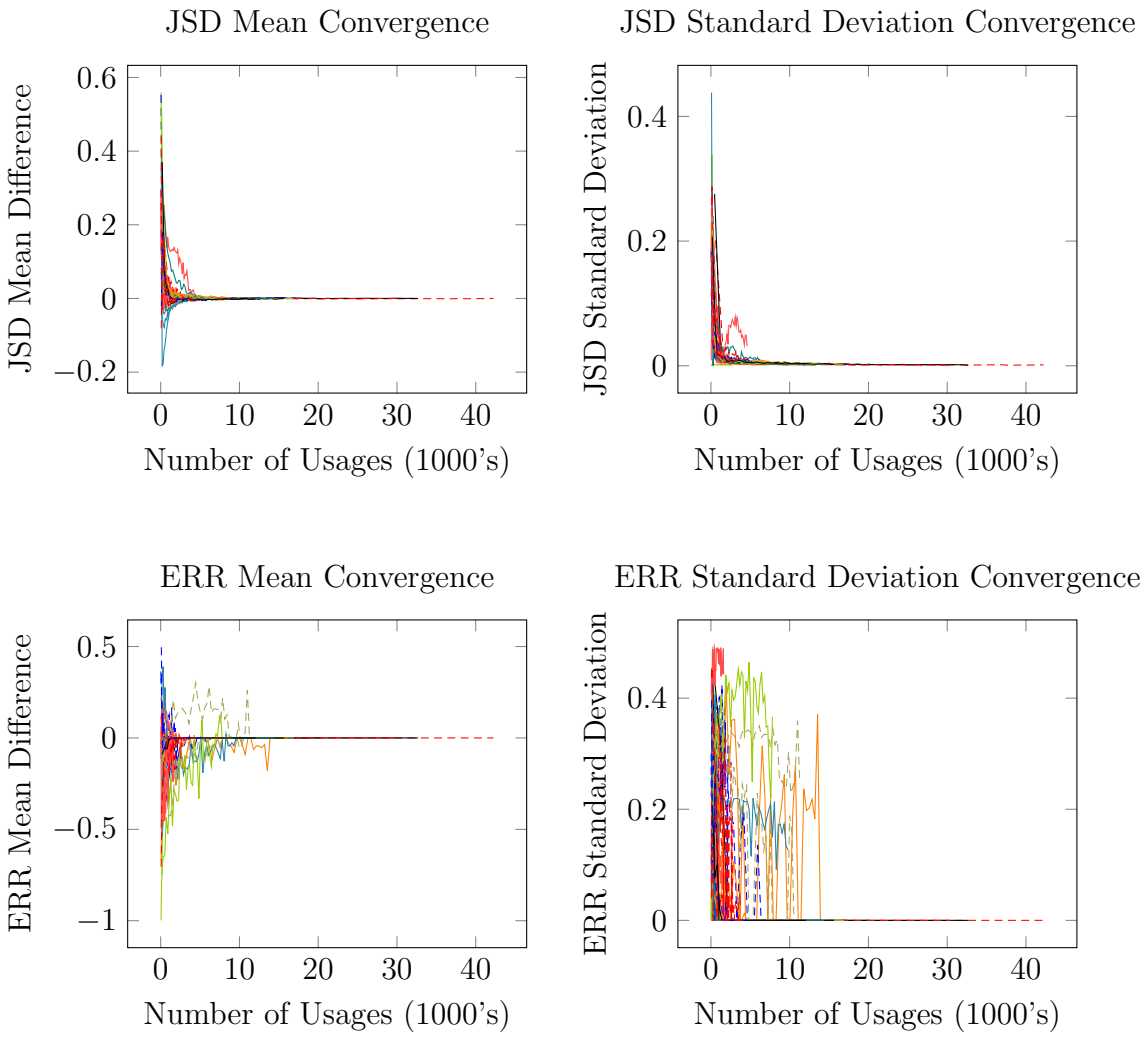

Figure 4.1: Results from our HDP-WSI convergence experiment in terms of sense distribution quality. For each lemma in L_{bnc} (using the BNC corpus), we split the bootstrapped sense distributions into 40 bins of approximately equal size (roughly 20 distributions per bin), and calculated the JSD and ERR for each sense distribution. For each combination of statistic (mean and standard deviation) and metric (JSD and ERR), we plot one line per lemma from one data point per bin, based on the statistic of the metric values in the bin. Note that for the plots using the mean statistic, the y axis measures the difference between the mean metric in each bin and the mean metric in the final bin for the same lemma.

quantity of usages is increased. We observe that when we have more than 5,000 lemma usages, the number of topics used is almost never above 10.

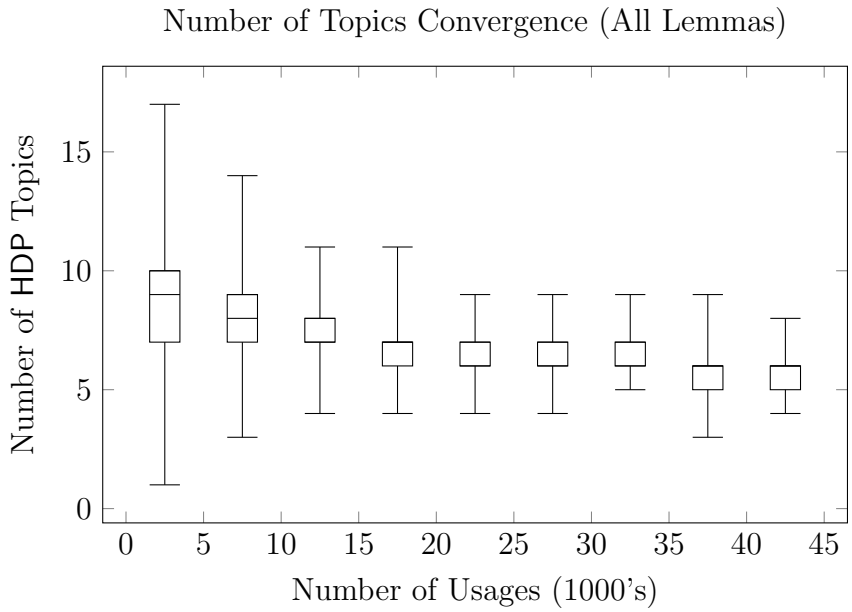

Figure 4.2: Results from our HDP-WSI convergence experiment in terms of the number of HDP topics, for all lemmas in L_{bnc} pooled together (using the BNC corpus). We binned all HDP topic models created during the experiment based on the number of lemma usages they were trained on, with a width of 5,000 usages per bin, and produced a boxplot of the number of HDP topics for each bin. Note that this means each bin contains an unequal number of models, and that the results for most lemmas are split between multiple boxplots.

A possible explanation for the previous result is that some of the lemmas in L_{bnc} with fewer available usages inherently produce a greater number of topics; indeed 18 of the 40 lemmas have fewer than 5,000 usages available in total. In other words, the number of topics may decrease as the number of usages is increased because we are looking at a progressively smaller set of lemmas. If this were true, it could contradict our conclusion that HDP almost never needs more than 10 topics when enough data is present. However, Figure 4.3 contradicts this hypothesis; it clearly shows that for most lemmas, the average number of topics produced decreases as the number of usages is increased. Furthermore, we can see that every lemma needs at most around 10 topics on average at convergence. We speculate that the decrease in the number of topics with more lemma usages occurs because the patterns in lemma usages become clearer with more data, so fewer topics are needed to explain them.

Based on these results, we can conclude by answering our initial questions. It appears that around 5,000 to 10,000 usages are required for stable results, and this number does not appear to vary significantly between lemmas. Furthermore, it appears that when we have enough data for stable results, the number of topics required

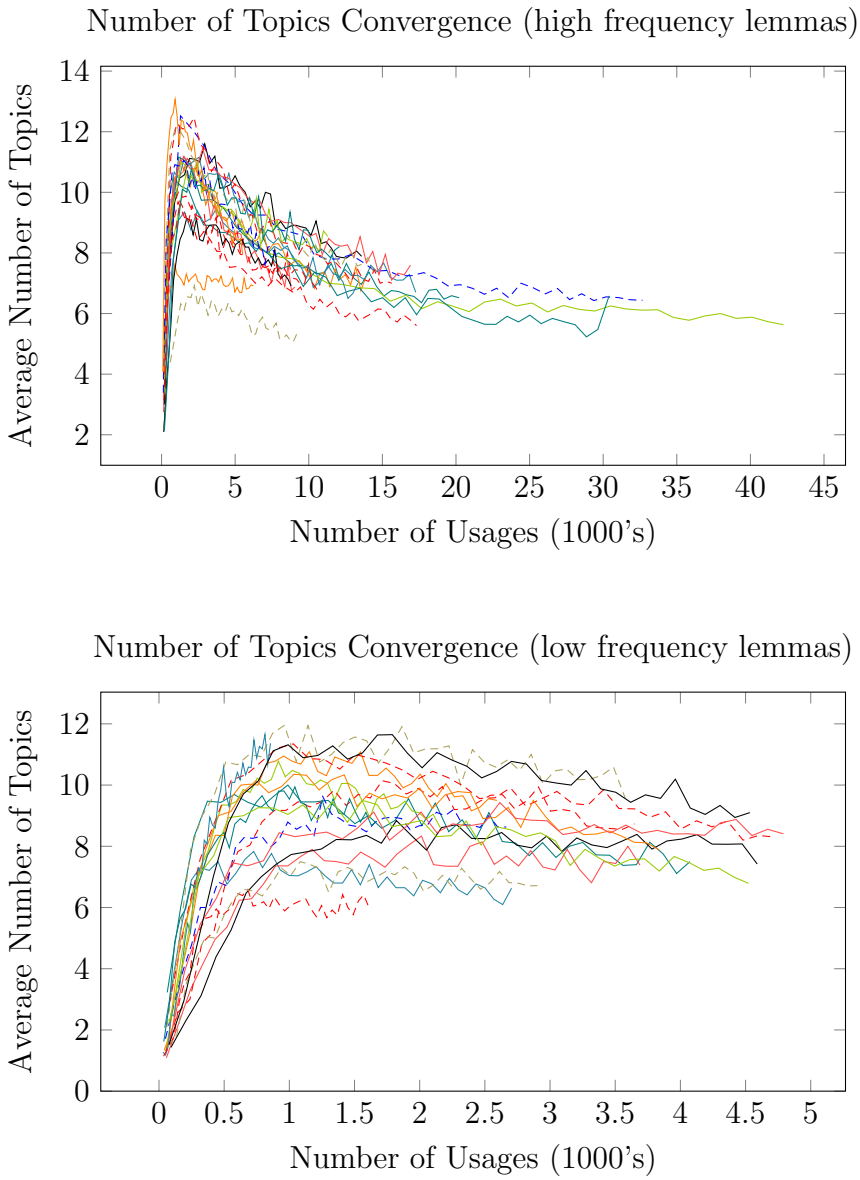

Figure 4.3: Results from our HDP-WSI convergence experiment in terms of the number of HDP topics, for each lemmas in L_{bnc} individually (using the BNC corpus). We partitioned the HDP models for each lemma into 40 bins of approximately equal size based on the number of lemma usages used for training. A separate line is plotted per lemma, with one data point per bin, based on the average number of HDP topics in the bin. Separate plots are provided for low frequency lemmas (those with fewer than 5,000 usages) and high frequency lemmas (those with at least 5,000 usages). Note that the x axis scales differ significantly between these plots.

Hyperparameter	Values	Description
T	10 or 20	Number of HCA topics.
hyp	True or False	Whether the Pitman-Yor Process extension of HCA is turned on for creating document distributions over topics.
burst	True or False	Whether the burstiness extension of HCA is turned on.

Table 4.1: Summary of the HCA hyperparameter settings we chose to experiment with in our HCA-WSI hyperparameter optimisation experiment. For each hyperparameter, the possible values are provided, along with a short description.

is almost never more than 10. This latter result in particular will guide our experiments with HCA topic modelling, which we describe next.

4.3 HCA Experiments

4.3.1 Introduction

Now that we have explored the convergence properties of HDP-WSI, in terms of both sense distribution quality and the number of HDP topics produced, we turn to our experiments trying to improve the HDP-WSI method. Since the core of HDP-WSI involves HDP topic modelling, the method can easily be customised by changing the topic modelling method.

In Section 2.3 we identified HCA (described in detail in Section 3.2.2) as an appropriate alternative topic modelling method. This was due to its similarity with HDP in terms of the structure of the underlying model, the very efficient inference algorithm used, and the refinements of its underlying probabilistic model. In addition, although this method requires a fixed number of topics to be used (unlike HDP), we found in Section 4.2 that HDP almost never used more than 10 topics when enough training data was provided for stable results.

Given these reasons, we proceed with a series to experiments exploring whether HDP-WSI can be extended by replacing HDP with HCA. In particular, we wish to discover whether using HCA results in an improvement in either training time or sense distribution quality that isn't at the expense of the other.

4.3.2 Experimental Setup

Because the only dependence of HDP-WSI on the output of HDP is via the topic distributions over words and the document distributions over topics generated by HDP, substituting HDP for HCA is straightforward: we simply run HCA instead of

HDP, and use the corresponding topic and document distributions output by HCA. We refer to this extension of HDP-WSI as HCA-WSI.

First we experimented with the effects of HCA hyperparameter settings on HCA-WSI. There are a large number of possible hyperparameter options, so we limited our analysis to a small number of key settings, which are summarised in Table 4.1. In Section 4.2 we found that the number of topics used was always less than 20, so we trial 20 as a conservative number of topics. However, because the computation of HCA increases with the number of topics used we want to use as few topics as possible. Therefore, given that we also concluded from Section 4.2 that we can probably get away with using 10 topics, we also trial this value. The other two hyperparameter options relate to the configuration of the underlying probabilistic model. By default the Pitman-Yor extension of HCA is only applied to the creation of topic distributions over words, so we experiment with extending this to the document distributions over topics as well.[3] In addition, we experiment with whether the burstiness extension of HCA is enabled or not.[4] Our default setup was to use 10 topics (using as few as possible), and turning the Pitman-Yor extension for topic distributions over words off and the burstiness extension on (as recommended in the HCA documentation). We evaluated each hyperparameter setup for all L_{bnc} lemmas on the BNC corpus (see Section 3.1.3) using 300 Gibbs sampling iterations,[5] in order to decide on an optimal setup.

Next, we experimented with the number of Gibbs sampling iterations separately and in more detail. This choice is particularly important to us, because the running time of HCA is roughly linear in the number of Gibbs sampling iterations. We ran HCA-WSI multiple times for each lemma in L_{bnc} on the BNC corpus, using our optimised hyperparameter settings. For each lemma we varied the number of Gibbs sampling iterations from 20 to 1000 in increments of 20: in each instance we calculated the perplexity,[6] and evaluated the resulting sense distribution using JSD. From this we could determine the minimum number of Gibbs sampling iterations needed for HCA-WSI to consistently converge.

Given our choices of hyperparameters and the number of Gibbs sampling iterations, we compared the performance of HCA-WSI to HDP-WSI. We did this comparison on the BNC corpus for all lemmas in L_{bnc}, as with the previous experiments.

[3]We enable the Pitman-Yor extension for the document distributions over topics by setting the initial values of the corresponding discount hyperparameters to 0.05.

[4]We enable burstiness by setting the initial values of the concentration and discount parameters for burstiness to 100 and 0.5 respectively, as is recommended in the HCA documentation.

[5]This was the number of iterations used for HDP in previous experiments in Section 4.2 and by Lau *et al.* (2014). We justify using the same number here because the inference algorithm used by HCA has been found to converge more quickly than that used by HDP (Chen *et al.* 2011).

[6]Perplexity in this instance is a measurement of how well the model fits the data it was trained on. It is calculated as a function of the log-likelihood of the topic model on the training data, which is normalised by the total number of tokens in all documents (this means that perplexity values for different lemmas are comparable).

Setup	JSD		ERR	
	Mean JSD	p	**Mean ERR**	p
T10-burst	.211±.117	-	.635±.402	-
T10	.212±.117	.175	.635±.402	1.000
T10-py	.212±.116	.162	.635±.402	1.000
T10-py-burst	.211±.116	.778	.652±.395	.317
T20	.212±.115	.979	.635±.402	1.000
T20-py	.212±.116	.122	.635±.402	1.000
T20-burst	.212±.117	.129	.652±.395	.317
T20-py-burst	.213±.116	.013	.635±.402	1.000

Table 4.2: Results of our HCA-WSI hyperparameter optimisation experiment. For each hyperparameter setup, we list the average JSD and ERR values for the lemmas in L_{bnc} (using the BNC corpus). In the name of each hyperparameter setup, the prefix indicates how many topics were used, the suffix "py" is present if the Pitman-Yor extension for document distributions over topics was turned on, and the suffix "burst" is present if the burstiness extension was turned on. For each setup and evaluation metric, a p value is provided comparing the metric values pairwise to those from the default setup (T10-burst), using two-sided Wilcoxon signed rank tests.

However, since replacing HDP with HCA is a significant change and we want to be as sure as possible that it won't negatively affect performance, we repeated this experiment with the SPORTS and FINANCE corpora (see Section 3.1.3), again for all lemmas L_{bnc}. For each lemma and each corpus, we evaluated the sense distributions from each method using JSD and ERR, and also the time taken to train the respective topic model.[7]

Finally, we repeated the experiments from Section 4.2 exploring the convergence of sense distribution quality with HCA-WSI instead of HDP-WSI,[8] in order to determine whether the previous conclusions about the minimum number of usages required still hold for HCA-WSI.

4.3.3 Results

The results of our hyperparameter optimisation experiment are presented in Table 4.2. We list the average JSD and average ERR for each hyperparameter setup, as well as the p-values comparing the JSD and ERR values pairwise with those from

[7]We measure this time since it accounts for the majority of the computation time of HDP-WSI and HCA-WSI. These experiments where we measured computation time were run using separate cores on Intel Xeon CPU E5-4650L processors, on a Dell R820 server with 503GiB of main memory.

[8]This was performed identically to Section 4.2, except that approximately 580 sense distributions were created per lemma, as opposed to 810 in the previous experiment.

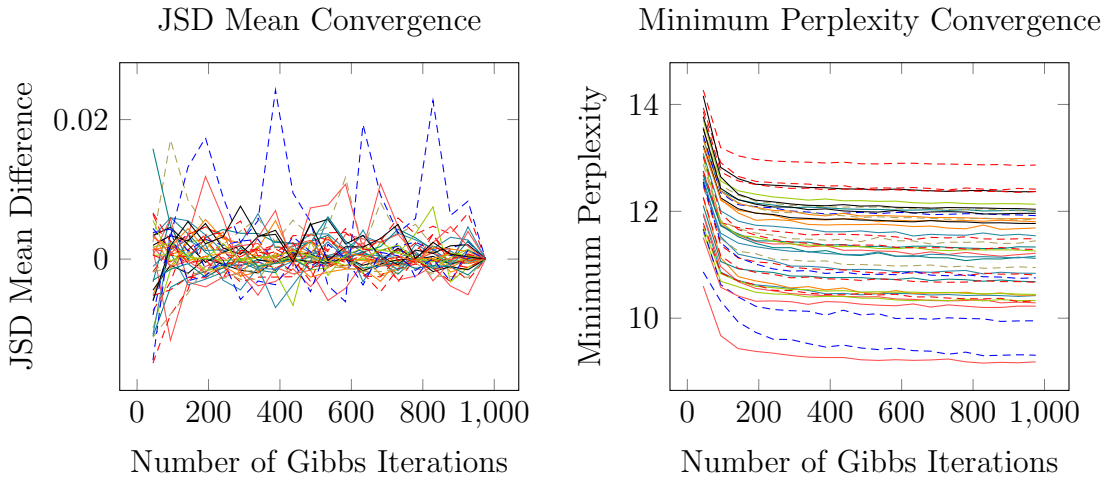

Figure 4.4: Results of our HCA-WSI Gibbs sampling convergence experiment. The results from independent runs of HCA-WSI for each lemma in L_{bnc} (using the BNC corpus) were binned, based on the number of Gibbs sampling iterations (giving 20 bins per lemma, with on average 2.5 runs per bin). For each lemma, we plot the average JSD and perplexity for each bin, plotting one line per lemma, and one data point per bin. Note that for the JSD plot, the y axis measures the difference between the mean JSD in each bin and the mean JSD in the final bin for the same lemma.

Dataset	JSD			ERR		
	HCA-WSI	**HDP-WSI**	p	**HCA-WSI**	**HDP-WSI**	p
BNC	.211±.117	**.209±.116**	.221	**.635±.116**	.633±.406	.715
SPORTS	.318±.212	**.318±.212**	.645	**.534±.434**	.553±.437	.317
FINANCE	.345±.148	**.342±.146**	.485	.604±.456	**.630±.451**	.285

Table 4.3: Results of our comparison of HCA-WSI and HDP-WSI in terms of sense distribution quality. For each combination of corpus (BNC, SPORTS, and FINANCE) and evaluation metric (JSD and ERR), we list the average metric value from both methods, and a p value comparing these values pairwise (using two-sided Wilcoxon signed rank tests).

the default setup (using two-sided Wilcoxon signed rank tests). Since no setup provided performance that was statistically significantly different than that of the default setup, we chose to use the default setup in subsequent experiments.

Next, the results of our experiments exploring the number of Gibbs sampling iterations are shown in Figure 4.4. There we display the convergence of the average JSD and average perplexity as the number of Gibbs sampling iterations is increased.

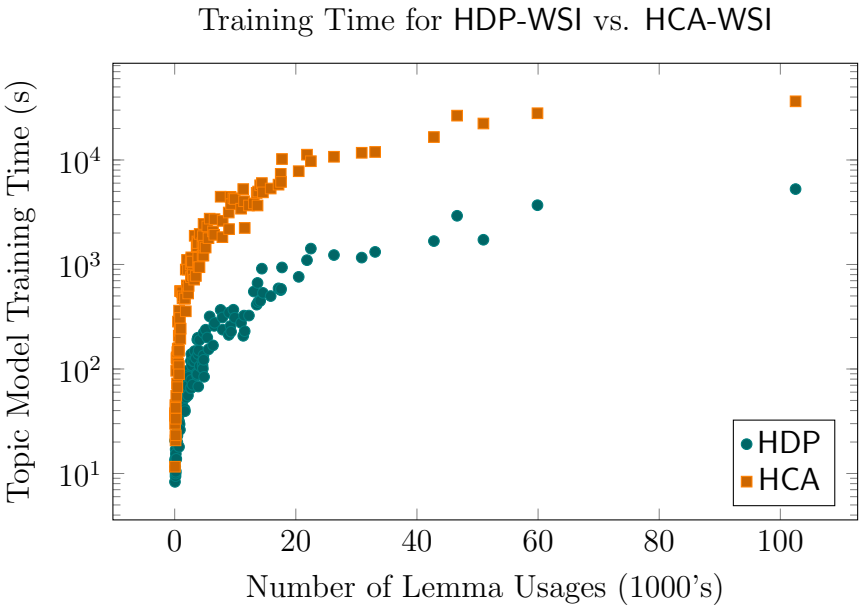

Figure 4.5: Results of our comparison of HCA-WSI and HDP-WSI in terms of computation time. For each combination of lemma in L_{bnc} and corpus (BNC, SPORTS, and FINANCE), we plot the corresponding topic model training times with both HDP and HCA.

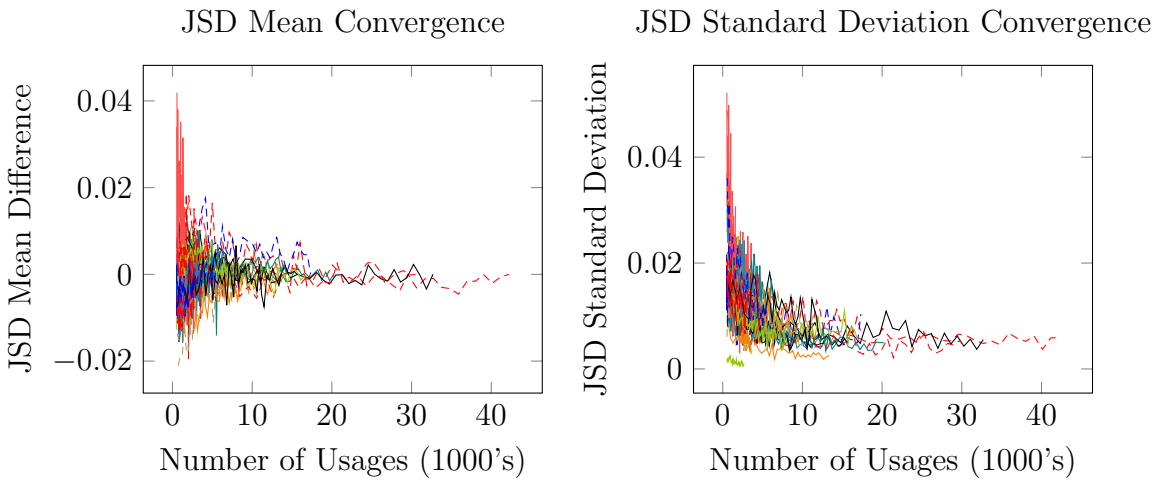

Figure 4.6: Results from repeating our convergence experiment in terms of JSD from Section 4.2 with HCA-WSI. For each lemma in L_{bnc} (using the BNC corpus), we split the bootstrapped sense distributions into 40 bins of approximately equal size (roughly 15 distributions per bin), and calculated the JSD for each sense distribution. For each statistic (mean and standard deviation), we plot one line per lemma from one data point per bin, based on the statistic of the JSD values in the bin. Note that for the mean JSD plot, the y axis measures the difference between the mean JSD in each bin and the mean JSD in the final bin for the same lemma.

In each case the graphs were created by grouping all data points into 20 bins of approximately equal size (based on the number of Gibbs sampling iterations), and the average JSD or perplexity is plotted for each bin. From these results we decided that the 300 iterations previously used with HDP seems about right for HCA as well, since both JSD and perplexity seem mostly converged around this point. Therefore we continued to use 300 iterations in subsequent experiments.

Finally, given our choice of hyperparameter setup and number of Gibbs sampling iterations, we present our results comparing HCA-WSI to HDP-WSI and repeating the number of usages convergence experiment for HCA-WSI. We compare the quality of sense distributions from the two methods using JSD and ERR in Table 4.3, and for each combination of corpus and quality metric we list the p-value comparing the quality values pairwise (using two-sided Wilcoxon signed rank tests). The comparison of time taken by HCA versus HDP is plotted in Figure 4.5, over all lemmas in L_{bnc} for all three corpora. For the sake of brevity we only present results in terms of JSD for the convergence of HCA-WSI over varying numbers of usages (ERR results were not significantly different), which are displayed in Figure 4.6.

4.3.4 Discussion

The first observation we can make from the above results is that HCA-WSI seems to be very robust to hyperparameter settings. Indeed, none of the hyperparameter setups we tried gave results that were distinguishable from the others at a level of statistical significance. This is perhaps not surprising, given that: (1) the input documents (lemma usages) are very short,[9] meaning that burstiness in unlikely to be significant; (2) given the conclusions of Section 4.2 we would expect any topics used above 10 to be unimportant; and (3) unlike the topic distributions over words, there is not as clear a basis for using Pitman-Yor processes for creating the document distributions over topics.[10] Furthermore, HCA-WSI does not seem to be very sensitive to the number of Gibbs sampling iterations; after around 300 iterations perplexity has mostly converged, and JSD does not appear to change significantly compared to its overall variance. These findings are encouraging, since it suggests we can easily apply HCA-WSI on a large scale without worrying about biased results due to bad hyperparameter settings.

Next, we can observe from Table 4.3 that the performance of HDP-WSI and HCA-WSI in terms of sense distribution quality are comparable, and the methods cannot be distinguished at any reasonable level of statistical significance with either evaluation metric on any of our corpora ($p > 0.2$ in all cases). We can conclude from this that the extensions to the HDP probabilistic model implemented by HCA do not seem to offer any significant improvements on average, and also that the use of a fixed number of topics does not seem to harm performance noticeably.

However, we can observe from the bootstrapping results in Figure 4.6 that the overall variance in JSD as the number of usages is decreased is lower using HCA-WSI compared to what was previously observed for HDP-WSI (as is displayed in Figure 4.1). For HCA-WSI, we can see that the mean change in JSD for each lemma as the number of usages is decreased is almost always less than 0.02, and never greater than 0.04, and similarly the standard deviation of the change in JSD is almost always always less than 0.02, and never greater than 0.05. In contrast to this for HDP-WSI, we can see that for many lemmas, both the mean and standard deviation of the change in JSD as the number of usages is decreased is above 0.1. This suggests that HCA-WSI is more robust than HDP-WSI overall, with less variation in output.

Furthermore, it is clear from Figure 4.5 that HCA-WSI is significantly better than HDP-WSI in terms of training time. Indeed, in our experiment HCA was consistently at least an order of magnitude faster than of HDP. Given this and the previous findings, we can conclude that there appears no good reason to use HDP-WSI for

[9] They are a maximum of three sentences: the sentence containing the lemma, and its neighbours (if available).

[10] This is because for topic distributions over words we would theoretically expect Zipfian distributions (Piantadosi 2014), whereas we don't have a clear theoretical expectation for document distributions over topics (especially given that the topic variables are artificial).

sense distribution learning rather than HCA-WSI.

Finally, the general trend in HDP-WSI convergence with respect to the number of lemma usages displayed in Figure 4.6 is consistent with the trend for HDP-WSI found in Section 4.2. In both cases, it appears that around 5,000 to 10,000 usages are required for stable results.

Now that we have explored how to use more efficient topic modelling to optimise and extend HDP-WSI sense distribution learning to HCA-WSI, we can explore whether there are any further extensions of HCA-WSI that can make it even more efficient or improve its performance. In the next and final section of this chapter, we explore one such possible extension.

4.4 Multi Lemma Topic Modelling Experiments

4.4.1 Introduction

In Section 4.2 we explored the convergence of HDP-WSI with respect to the quantity of training data to find out how little data we could use and still obtain optimal performance, and subsequently in Section 4.3 we extended HDP-WSI to HCA-WSI by replacing HDP topic modelling with HCA, which we found to perform comparably to HDP-WSI in terms of sense distribution quality and the quantity of data required, but was at least an order of magnitude faster in terms of training time and more robust. We now explore whether this improvement can be extended further by applying HCA-WSI to multiple lemmas at once.

Previously, HDP-WSI and HCA-WSI had been applied individually to every lemma by learning a separate topic model for each. However, it is possible that related lemmas whose underlying topics are similar could make use of a common set of usages, allowing one topic model to be trained per cluster of lemmas. Alternatively, it is possible that relatively dissimilar lemmas could benefit from pooling their usages to train a common topic model, by providing a more diverse selection of topics and facilitating statistical sharing.

If we could get away with training only one topic model per group of lemmas the training time of sense distribution learning could be reduced dramatically, and it is also possible that the associated statistical sharing could result in higher quality sense distributions. Therefore, we perform a pilot experiment exploring a variant of HCA-WSI (respectively HDP-WSI) that performs topic modelling on clusters of lemmas, which we name ML-HCA-WSI (respectively ML-HDP-WSI). The main aim of this pilot experiment is to determine whether ML-HDP-WSI or ML-HCA-WSI have merit, in which case we could optimise them to provide a more efficient method for language-wide sense distribution learning.

4.4.2 Experimental Setup

In order to apply ML-HCA-WSI to a cluster of lemmas, we provide it with a set of usages of all lemmas in the cluster, and it produces a separate sense distribution for each lemma. As with HCA-WSI, ML-HCA-WSI has a WSI phase and a topic–sense alignment phase. In the WSI phase we run HCA on the pooled set of usages of all lemmas in the cluster, in order to produce a common topic model. The corresponding topics represent a common set of induced senses for the lemmas in the cluster. However, a separate global distribution over these topics is created for each lemma, which is done almost identically to HCA-WSI, except that we only count topic allocations for documents that are usages of that lemma. Then in the topic–sense alignment phase we perform alignment separately for each lemma, using the common set of topics and the lemma's global distribution over topics, which is done identically to HCA-WSI. ML-HDP-WSI is defined analogously, using HDP instead of HCA.

We first experimented with ML-HDP-WSI in order to obtain statistics on the number of HDP topics required. As with our prior experiments on optimising sense distribution learning, this was performed using the L_{bnc} lemmas and the BNC corpus. We ran ML-HDP-WSI on a large number of random subsets of L_{bnc} (223 subsets in total), in each case pooling all available usages of each lemma in the cluster. Our configuration of HDP was the same as in previous experiments, except that we conservatively increased the number of Gibbs sampling iterations from 300 to 1,000 since we were operating on a larger quantity of data on average. Each of these random lemma clusters was sampled by selecting a number of lemmas between 2 and 40 (from a uniform distribution), and then selecting that many lemmas from L_{bnc} without replacement. These results were used to decide how many topics were required for ML-HCA-WSI.

Given these results, we then repeated the previous experiment with ML-HCA-WSI. We used the same HCA setup from Section 4.3, except we conservatively increased the number of Gibbs sampling iterations to 1,000 again. In total we ran ML-HCA-WSI on 452 random lemma clusters.

The idea behind running these methods on a large number of random lemma clusters is that we can measure the quality of output for each cluster, and search for any correlations between high quality outputs and cluster features. In order to produce quality scores for the sense distribution output of lemma clusters that can be compared between different clusters, we defined two new metrics for this experiment: JSD Gain and ERR Gain. JSD Gain (respectively ERR Gain) is defined as the difference between the JSD (respectively ERR) of a sense distribution created using this method, and that of the corresponding benchmark distribution obtained using HCA-WSI or HDP-WSI on the BNC corpus (as was obtained in Section 4.3). Therefore, the sense distribution quality of different lemma clusters can be compared by calculating the average JSD Gain (respectively ERR Gain) of each cluster. A negative JSD Gain or positive ERR Gain indicates that the quality of output from ML-HCA-WSI (respectively ML-HDP-WSI) is better than that from HCA-WSI (respectively

HDP-WSI), and vice versa for positive JSD Gain or negative ERR Gain.

In order to search for correlations with the sense distribution quality of lemma clusters, we experimented with two simple lemma cluster features. The first is simply the number of lemmas in the cluster. The motivation behind this feature is that ML-HCA-WSI may only work well on clusters of a certain size (for example, it may only work well on small clusters). The second feature we experimented with is the average WordNet similarity between all distinct pairs of lemmas in the cluster. The WordNet similarity function we used for this feature is the method of Jiang and Conrath (1997) — this method was introduced briefly in Section 2.2.2 — which we refer to as JCN.[11] This feature is intended to measure how cohesive a cluster is; we hypothesise that clusters containing either very similar or dissimilar lemmas may particularly benefit from ML-HCA-WSI.

Since we are using all available usages of every lemma in each cluster, we are not achieving any savings in computation time compared to running sense distribution learning on each lemma individually;[12] in practice, if we were to use ML-HCA-WSI and achieve any savings we would need to use relatively fewer usages of each individual lemma. Therefore this experiment is intended to provide an upper bound of the sense distribution quality from ML-HCA-WSI. Unless it is capable of producing sense distributions of at least similar quality to those from HCA-WSI, at least for some subset of lemma clusters that can be separated based on cluster features, it is not worth exploring further.

4.4.3 Results

Of the 223 HDP models trained on random L_{bnc} lemma clusters from our ML-HDP-WSI experiment, we found that in approximately 93% of instances the number of topics used was 10 or less. In addition, the number of topics used was never greater than 12.

As a result of this, we decided that it was appropriate to use 10 topics in our subsequent experiment with ML-HCA-WSI, as was used to create our benchmark HCA-WSI sense distributions in Section 4.3. A summary of the results from this experiment, in terms of sense distribution quality versus lemma cluster features, is displayed in Figure 4.7. We plot results using our two lemma cluster features (number of lemmas and average JCN similarity), and our two quality measures (Average JSD Gain and Average ERR Gain).[13]

[11] Technically Jiang and Conrath's (1997) method works on synsets, not lemmas. We calculated the JCN similarity of a given pair of lemmas by the maximum JCN similarity between any pair of synsets of the two lemmas from WordNet.

[12] This is because the running time of topic modelling is at least linear in quantity of training data.

[13] We chose not to plot corresponding results for ML-HDP-WSI, for the sake of brevity. However, these results were almost identical to those for ML-HCA-WSI.

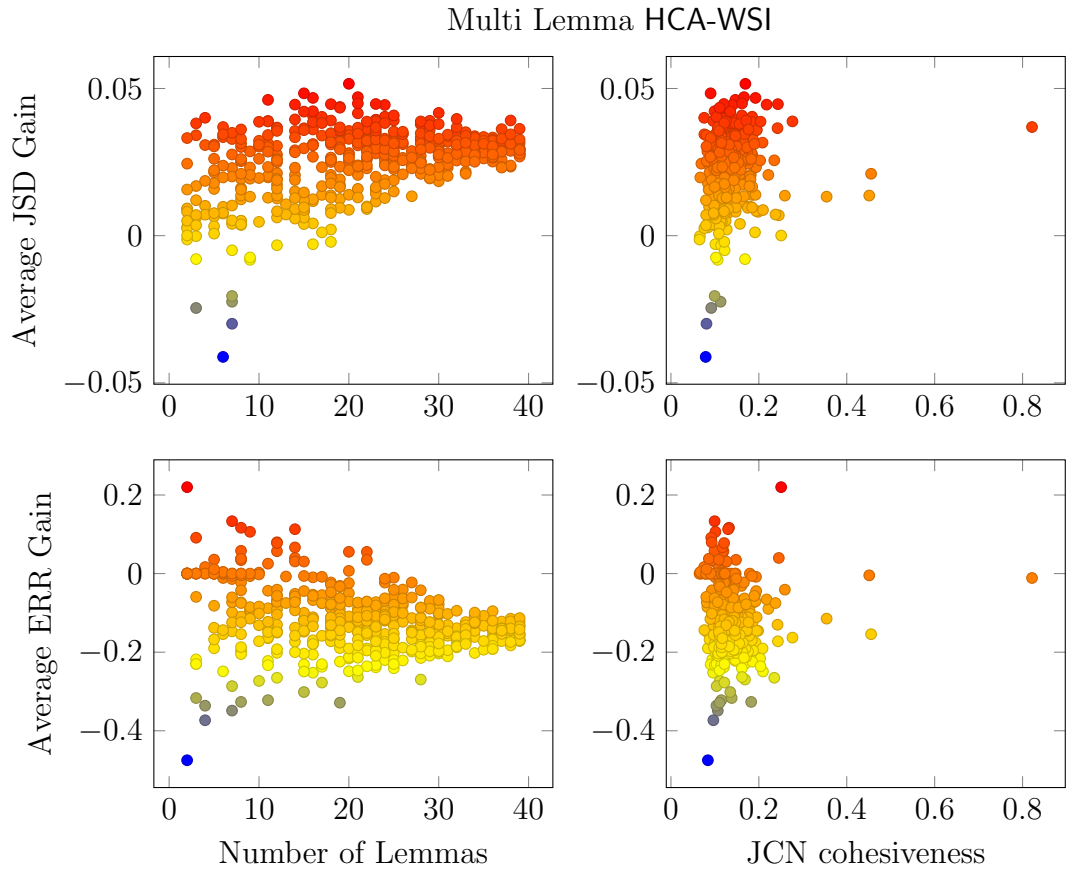

Figure 4.7: Results of our experiment with ML-HCA-WSI, comparing the sense distribution quality of random lemma clusters (from lemmas in L_{bnc}, using the BNC corpus) to cluster features. For each combination of quality metric (average JSD Gain and average ERR Gain) and cluster feature (number of lemmas and average JCN similarity) we plot a scatterplot of results, where each data point corresponds to a single lemma cluster.

4.4.4 Discussion

We can observe in Figure 4.7 that according to both quality metrics, the performance of ML-HCA-WSI is almost always worse than that of HCA-WSI, except in a small handful of cases. Furthermore, it is also clear that this small handful of cases are not separable from the vast majority of cases where performance is worse, using either lemma cluster feature.[14]

[14]We also experimented briefly with using both features together. However, there was no combination of the two features where the sense distribution quality of ML-HCA-WSI wasn't significantly worse on average. These results are not shown for the sake of brevity.

This experiment was run under very generous conditions: all available usages of each lemma were pooled together, and a very large number of Gibbs sampling iterations were used. However, despite this the results fail to demonstrate that ML-HCA-WSI is capable of producing sense distributions whose quality is not worse on average than those from HCA-WSI. This holds even if we attempt to control for lemma cluster features, such as the number of lemmas in the cluster or the similarity of lemmas in the cluster. Indeed, we found that according to either sense distribution quality metric, the ML-HCA-WSI sense distributions were worse than those from HCA-WSI in over 95% of instances. Therefore, we conclude that performing topic modelling on multiple lemmas at once is not worthwhile, and that we would be better off performing HCA-WSI on each lemma individually.

4.5 Conclusion

In Section 4.2 we concluded that around 5,000 to 10,000 lemma usages are required to achieve stable results with HDP-WSI. We found this number to be very consistent over a range of lemmas, and that there was little benefit to be gained by using more training data than this. This is an important result for applying HDP-WSI in settings where a potentially unbound quantity of lemma usages are available.

Then in Section 4.3 we experimented with the use of HCA instead of HDP in sense distribution learning. We thus extended HDP-WSI to HCA-WSI, which we found to perform almost identically in terms of sense distribution quality over a range of lemmas and corpora. However, we found that HCA-WSI is consistently at least an order of a magnitude faster than HDP-WSI, and is more robust than HDP-WSI, with significantly less variation in sense distribution quality when trained on random sets of usages. In addition, we found that the previous result for HDP-WSI regarding the number of lemma usages required also applies to HCA-WSI, and also that HCA-WSI does not appear to be sensitive to the hyperparameter settings used.

Finally, in Section 4.4 we experimented with ML-HCA-WSI, which is an extension of HCA-WSI that tries to be more efficient by running topic modelling on clusters of multiple lemmas at once. However, we found that this extension significantly hurt the average quality of sense distributions produced, and we failed to discover any lemma cluster features that could be used to identify clusters where ML-HCA-WSI performs as well as HCA-WSI.

The main outcome of this chapter is a clear blueprint for performing cost-efficient, language-wide sense distribution learning: HCA-WSI should be used as the sense distribution learning method, around 5,000 to 10,000 usages of each lemma should be used if available, at minimum around 300 Gibbs sampling iterations should be used by HCA, and around 10 topics for HCA should be sufficient. Other HCA settings were found to be relatively unimportant, though we can recommend using the default HCA settings with burstiness turned on (as described in Section 4.3) as a safe setup.

While most of these results were only obtained using the BNC corpus — except for our experiments comparing HDP-WSI to HCA-WSI, which also used the SPORTS and FINANCE corpora — this corpus is domain-independent, and we evaluated over a diverse set of lemmas. Therefore, it is reasonable to believe that these results will hold for other domain-independent corpora, which we may use to produce language-wide sense frequencies. In addition, given that — as was decided in Section 1.2 — we restricted our scope to English nouns, we need to hedge our conclusions accordingly, noting that we can only make strong conclusions about this class of lemmas.

Now that we have built a clear blueprint for performing efficient language-wide sense distribution learning, we have addressed our first core research question, regarding how to optimise sense distribution learning for large-scale application. In the next chapter we take this blueprint and apply it on a language-wide scale, in order to address the remaining questions regarding replacing existing sense frequency resources, and MWE sense distribution learning.

Chapter 5

Application of Unsupervised All-words Sense Distribution Learning

5.1 Introduction

In Chapter 4 we provided a template for efficient large-scale sense distribution learning. In doing so we addressed our first research question, regarding how to optimise sense distribution learning for language-wide application. Now we apply our optimised method in order to address the follow up questions: (1) can unsupervised all-words sense distribution learning be used to replace or supplement existing sense frequency resources such as SemCor? and (2) can this sense distribution learning be applied to multiword expressions (MWEs) as well as simplex lemmas? In addition, we address our secondary research aim of actually creating a new language-wide sense frequency resource, using our sense distribution learning method.

First in Section 5.2 we describe the creation of LexSemTm, which is a sense frequency dataset containing the output of HCA-WSI— including distributions over WordNet senses for English lemmas, and the WSI output from HCA-WSI for all lemmas, which can easily be aligned to any sense inventory with glosses — over the vocabularies of several languages (English, Japanese, Italian, Mandarin, and Indonesian). We describe not only the creation of the dataset for simplex lemmas, but how we extend HCA-WSI to MWEs. This involves proposing two simple but novel methods for identifying MWE usages, which are used to add two sets of WSI and sense distribution data to LexSemTm for MWE lemmas.

In the subsequent sections of this chapter, we use LexSemTm in order to answer our remaining research questions. In Section 5.3 we address the question of whether LexSemTm can be used to supplement or replace the existing sense frequency data in SemCor. We describe the creation of a new gold-standard evaluation dataset, which

we use to answer this question in the affirmative. Then in Section 5.4 we address the question of sense distribution learning for MWEs. We create a new gold-standard dataset for evaluating MWE sense distribution learning, which we use to provide a thorough exploration of the novel tasks of MWE sense distribution and first sense learning. In the course of this analysis we demonstrate that MWE sense distribution learning is indeed possible, and that similar results can be obtained for this task as for simplex lemmas.

5.2 LexSemTm Creation

5.2.1 Introduction

Throughout Chapter 4 we explored how to optimise sense distribution learning for large-scale application. By the end of the chapter we had constructed a blueprint for doing this, based on extending the HDP-WSI method of Lau *et al.* (2014) (see Section 3.2.3) to HCA-WSI by replacing HDP with HCA (see Section 3.2.2), together with some guidelines for applying HCA-WSI efficiently. Most importantly we found that: (1) around 5,000 to 10,000 lemma usages are required to be confident of stable results; (2) around 300 Gibbs sampling iterations are required for HCA to converge; and (3) around 10 topics is generally sufficient for HCA. In addition, we concluded that our default HCA setup — turning on the burstiness extension of HCA, using the numbers of Gibbs sampling iterations and topics described above, and using otherwise default settings — was appropriate for sense distribution learning, and that our method is robust to variations in this setup.

We now take that blueprint, and apply it language-wide across several languages (English, Japanese, Italian, Mandarin, and Indonesian) in order to create a new sense frequency resource: LexSemTm. In this section we detail the construction of LexSemTm, focussing on: (1) how usages of simplex lemmas from each language were obtained; (2) how simplex lemmas from each language were selected for sense distribution learning; (3) how HCA-WSI was run on the simplex lemmas from each language; and (4) how the previous steps were extended to MWEs. We only provide a brief, qualitative evaluation of LexSemTm in this section, and leave more detailed analysis of the dataset to subsequent sections.

5.2.2 Creation of LexSemTm for Simplex Lemmas

Obtaining Simplex Lemma Usages

The corpora we used for creating LexSemTm were EnWiki for English lemmas, and MultiLingualWiki for non-English lemmas (see Section 3.1.3). In both cases, we identified usages of simplex lemmas by searching through the respective corpora

for all sentences containing the lemma with a matching POS tag.[1] In addition, in the case of English where we had a lemmatised and unlemmatised version of each sentence, we selected all sentences where either the lemmatised or surface form contained the lemma. In the case that a sentence contained more than one usage of the target lemma, we treated each occurrence in the sentence as a separate usage, which are distinct because they have different words neighbouring the target lemma (and therefore different local-context tokens in the usage document that is part of the input of HCA).

As with previous work on HDP-WSI by Lau *et al.* (2014), as well as our experiments in Chapter 4, we included the immediately neighbouring sentences in each usage (without crossing paragraph or section boundaries). However, we found that the sentences in EnWiki were somewhat shorter on average than those in BNC or MultiLingualWiki, giving usages for English lemmas containing substantially fewer tokens on average. Therefore for English lemmas only, we included up to one additional neighbouring sentence on either side.[2]

Selection of Simplex Lemmas

In order to obtain a set of simplex lemmas to process, we started off with a list of all possible simplex lemmas for each language. For English we obtained a list of all simplex lemmas in Princeton WordNet, and for other languages we obtained a list of all simplex lemmas from their respective WordNet in OMW (see Section 3.1.2 for details).[3]

For each simplex lemma of every language, we obtained the set of all usages of the lemma from the respective corpora using the procedure described above, and discarded all lemmas with fewer than 20 usages available. This gave us our final set of simplex lemmas to be included in LexSemTm.

Configuration of HCA-WSI

Our configuration for performing HCA-WSI on simplex lemmas was almost identical to the recommended setup from the conclusion of Chapter 4, which was sum-

[1]Both corpora had previously been POS tagged. As described in Section 3.1.4, we manually mapped all POS tag types in the corpora to "noun", "verb", "adjective", "adverb", or "other".

[2]To be more specific, for the non-English lemmas we included up to one neighbouring sentence on either side of the lemma occurrence, allowing up to two neighbouring sentences in total. However, for the English lemmas we included up to two neighbouring sentences on either side, allowing up to four neighbouring sentences in total. In both cases this is a maximum number of neighbouring sentences, because we do not cross paragraph or section boundaries.

[3]We defined lemmas in each WordNet as simplex if they did not contain any underscores in them. It should be noted that this is a somewhat naive definition for Mandarin and Japanese, however our aim as set out in Section 1.2 was to devise a sense distribution learning methodology that is language-independent. Furthermore, dealing with complex issues of tokenisation and the definition of words in these languages is well beyond the scope of this thesis.

marised in the introduction of this section. However, we had sufficient computational resources available to be slightly more conservative in the hyperparameter settings, compared to these lower-bound recommendations.

Therefore in order to be safe, since we are applying HCA-WSI over a much larger range of data than previously, we increased the number of Gibbs sampling iterations from 300 to 1,000, and we increased the number of topics used from 10 to 20. In addition, we sampled up to 40,000 usages of each lemma (if available), rather than 5,000 or 10,000. It should be noted that the results of Section 4.3 indicate that these changes should be unlikely to have any adverse effect. Indeed, we found in a preliminary experiment, where we calculated sense distributions for the lemmas in L_{bnc} using the BNC, SPORTS, and FINANCE corpora, that these changes did not have a statistically significant impact on sense distribution quality according to the JSD or ERR metrics.[4]

Other than these more conservative hyperparameters and the increase in the maximum number of usages, HCA-WSI was set up identically as in the conclusions of Chapter 4. The actual execution of HCA-WSI for each lemma in order to produce LEXSEMTM was done using the Google Compute Engine, as described in Section 3.1.6.[5]

5.2.3 Creation of LexSemTm for MWEs

In order to add MWE data to LEXSEMTM, we followed the same procedure as was used for simplex lemmas (described in Section 5.2.2) except for the identification of lemma usages. Clearly, given a set of usages for each MWE from WORDNET and OMW, we can still select those MWE lemmas with at least 20 usages, and apply HCA-WSI to the usage documents as above. However, identifying MWE usages in the input corpora introduces some additional challenges. Some of the key problems introduced include: (1) the presence of extra words within MWE usages; (2) dealing with morphological variants of individual words within the MWE; and (3) identifying MWE usages with the correct part of speech (POS).

We can see an example of the first problem in the sentence *we threw Dustin to the lions*, which contains the noun phrase *Dustin* within the MWE *throw to the lions*. Clearly in this case, identifying usages of the MWE by searching for contiguous usages of the component words would not be sufficient. While this may be a particular problem for certain kinds of MWEs such as verb–prepositional phrase combinations (as in the previous example), this could also be a problem for senses of nouns that are more compositional. For example *an old wise man* could be considered a valid usage of the `elderly man` sense of *old man*. Unfortunately, allowing such insertions

[4] $p > 0.05$ in all cases, according to two-sided Wilcoxon signed rank tests.

[5] For English lemmas these results are aligned to the senses in WORDNET, whereas for non-English lemmas we only produced the WSI output, since we did not perform any evaluation over these lemmas. However, aligning these to any sense inventory with glosses is trivial.

would likely introduce false positives; for example the sentence *every Friday he went to football training after school* is obviously not a usage of the noun *training school*.

The first example above also highlights the second problem. In this case one of the words in the usage is a morphological variant (*throw* changed to *threw*). However, if we were to simply lemmatise the entire sentence before searching, we would also likely transform *lions* to *lion*, so we still wouldn't have an exact match.

The third problem is that unlike for simplex lemmas, the use of POS tags to ensure we are identifying MWE usages with the correct POS is non-trivial. While we could perform parsing of each sentence and make use of the full parse-tree structure to help address this problem, this would be complex and computationally expensive, and our method would no longer be language-independent.[6] Therefore, we would prefer a simple approach that can be applied to existing POS-tagged corpora lacking parse-tree information.

Unfortunately, as noted in Section 2.4, to the best of our knowledge there are no existing methods to do this. Therefore, balancing the above concerns, we propose two different methods for identifying MWE usages from the EnWiki and MultiLingualWiki corpora, and we add MWE data to LexSemTm for all languages using each of these methods. As with our method for identifying simplex lemma usages, these methods are applied to each sentence in the corpora individually.[7]

High Precision Identification Method

For our high-precision identification method, we require all words in the MWE to appear contiguously in the sentence we are searching. In the case of English where we have surface and lemmatised forms of each sentence, we deal with morphological variations by allowing each word in the MWE to independently match either the surface or lemmatised form in the sentence. In addition, for all languages we require that at least one of the corresponding words in the sentence has a matching POS tag to that of the MWE lemma we are searching for. Using this method, the sentence *Dustin was thrown to the lions* would be identified as a valid usage of the verb *throw to the lions*, as long as *thrown* is correctly tagged as a verb: *throw* would match the lemmatised form, *lions* would match the unlemmatised form, and the remaining words would match both forms. However, *we threw Dustin to the lions* would not be identified, as the four words do not appear contiguously.

[6]This is because parsing is not available to all languages, and also because the way syntactic structure can be used will vary a lot between languages.

[7]Note that in both cases, we add the extra local-context tokens to the usage document based on the relative position of the remaining words in the sentence to the match of the first word in the MWE, after removing the remaining matched MWE words from the sentence.

Language	POS	Simplex		High Recall MWE		High Precision MWE	
		All	≥ 5,000	All	≥ 5,000	All	≥ 5,000
English	noun	36,383	4,664	31,175	975	22,254	316
	verb	10,457	2,303	2,434	414	2,093	243
	adjective	13,533	1,457	412	82	186	13
	adverb	2,348	377	653	181	209	32
Japanese	noun	27,766	3,120	90	9	84	5
	verb	2,227	118	21	1	17	1
	adjective	415	37	32	7	4	1
	adverb	604	64	87	36	5	1
Italian	noun	13,665	1,589	2,119	20	1,902	15
	verb	2,486	173	76	0	67	0
	adjective	3,148	326	107	8	42	0
	adverb	882	168	94	15	16	4
Mandarin	noun	5,625	848	0	0	0	0
	verb	3,005	442	0	0	0	0
	adjective	1,340	109	0	0	0	0
	adverb	739	174	0	0	0	0
Indonesian	noun	10,032	421	3,734	15	2,999	6
	verb	490	88	614	8	275	4
	adjective	124	48	844	12	156	1
	adverb	42	27	293	15	35	1

Table 5.1: Summary of the number of lemmas included in LexSemTm. Lemma counts are provided separately for each class of lemma, and are split by language and POS. In addition, separate counts are provided for all lemmas, and lemmas with a LexSemTm frequency of at least 5,000.

High Recall Identification Method

Our second identification method instead aims for high recall, with a bias towards identifying as many usages as possible. We achieve this via two changes to the high precision method: (1) we allow the insertion of up to 3 extra words between the MWE components; and (2) we completely ignore POS tags in the sentence. Given these changes, *we threw Dustin to the lions* would now successfully be identified as a usage of *throw to the lions*.

Given that we are simply obtaining data to train a statistical model, we hypothesise that the noise introduced by this high recall method may be outweighed by the greater volume of data.

5.2.4 Results

We now provide some summary statistics of the data contained in LexSemTm. In these results we divide the lemmas contained in LexSemTm into three categories: (1) simplex lemmas; (2) MWE lemmas identified using our high recall method; and

Language	**Lemma Class**	**WordNet count**	**Coverage**	
			All	$\geq$ 5,000
English	Simplex	28,429	88.3%	24.0%
	High Recall MWE	2,731	80.9%	13.4%
	High Precision MWE	2,731	67.3%	6.3%
Japanese	Simplex	30,070	45.2%	7.8%
	High Recall MWE	0	—	—
	High Precision MWE	0	—	—
Italian	Simplex	9,651	80.1%	17.5%
	High Recall MWE	179	66.5%	1.7%
	High Precision MWE	179	59.8%	1.7%
Mandarin	Simplex	1,808	42.8%	8.8%
	High Recall MWE	0	—	—
	High Precision MWE	0	—	—
Indonesian	Simplex	13,923	36.0%	3.2%
	High Recall MWE	3,344	36.3%	5.1%
	High Precision MWE	3,344	23.8%	1.5%

Table 5.2: Summary of the coverage of polysemous WordNet and OMW lemmas in LexSemTm. Coverage statistics are provided separately for each combination of language and lemma class, and for each combination we list the number of polysemous lemmas in the respective WordNet, and the percentage of these covered by LexSemTm. Coverage percentages are provided separately for all lemmas, and lemmas with a LexSemTm frequency of at least 5,000.

(3) MWE lemmas identified using our high precision method.[8]

Firstly, in Table 5.1 we provide a summary of the number of lemmas of each class contained in LexSemTm. These lemma counts for each class are split by language and POS, and we provide separate counts for all lemmas, and lemmas with a LexSemTm frequency[9] of at least 5,000 (for which we are more confident of the stability of their sense distributions, given the findings of Chapter 4).

Secondly, in Table 5.2 we provide a summary of the coverage of polysemous lemmas in each language's WordNet, for lemmas of each class. For each combination of language and lemma class, we list the total number of polysemous lemmas in the respective WordNet corresponding to that class,[10] as well as the percentage coverage of these lemmas by LexSemTm. Again, we provide separate statistics for all lemmas, and for lemmas with a LexSemTm frequency of at least 5,000.

5.2.5 Discussion

We can observe from Table 5.1 that the quantity of data in LexSemTm varies wildly between languages. This is because the corpora for each language vary substantially in size, and also because the number of lemmas in each WordNet varies significantly.[11] In addition, because of how we defined lemmas as simplex versus MWE (based on whether the lemma string in the respective WordNet contained an underscore), almost all lemmas in Japanese and Mandarin are defined as simplex, which is why the MWE counts for these languages are so low.[12] Unfortunately, this is the only MWE information available in the Chinese and Japanese OMW WordNet's, and we consider extracting MWEs from these languages for sense learning to be beyond the scope of this thesis.

In terms of the coverage of polysemous lemmas, our results are encouraging, particularly for English. In LexSemTm we have coverage of 88.3% of polysemous simplex lemmas from Princeton WordNet overall, which is very high compared to the corresponding coverage by SemCor (see Section 3.1.2); only 39.2% of these polysemous lemmas have at least one occurrence in SemCor. While the coverage of LexSemTm is much lower if we restrict it to lemmas with a LexSemTm frequency of at least 5,000 (24.0% for English), this is still a strong result if we consider that most lemmas

[8]Note that the two MWE categories overlap, since many MWE lemmas are identified using both methods. Indeed, every MWE usage that can be identified using the high precision method can also be identified using the high recall method, so the second category is a superset of the third.

[9]We use "LexSemTm frequency" to refer to the number of usages a lemma in LexSemTm was trained on.

[10]That is, either the number of polysemous simplex or MWE lemmas.

[11]In both cases, this is most likely due to some languages being better resourced than others, with more thorough efforts to populate their respective WordNet vocabularies and more Wikipedia pages. This is especially true for English.

[12]The counts for Japanese are only non-zero because of the presence of some English words in the Japanese WordNet.

covered by SemCor have a very low frequency in SemCor; the SemCor coverage drops to 17.2% if we require at least 5 occurrences in SemCor, or to 6.4% if we require at least 20 occurrences. Furthermore, of the 6,815 polysemous simplex lemmas in LexSemTm with a frequency of at least 5,000, 1,564 of them do not occur in SemCor at all, which accounts for over 5% of all polysemous simplex lemmas! Again, the lower coverage for other languages is likely due to the relative size of the English and non-English corpora.

In conclusion, we have described the creation of LexSemTm, which contains sense frequency information for a large proportion of the vocabulary of multiple languages. This dataset contains HCA-based WSI output for all of its lemmas, which can be aligned to any sense inventory with glosses, as well as distributions over Princeton WordNet senses. In addition to simplex lemmas, the dataset contains MWE lemmas, and in the process of creating the dataset we proposed two simple but novel methods for the general-purpose unsupervised identification of MWE usages. Finally, we have shown that the coverage of this dataset is far greater than that of SemCor, the de facto source of domain-independent sense frequencies for WordNet.

However, we have not yet investigated how the quality of the sense frequency data in LexSemTm compares to that of SemCor. In the next section we address this by providing a thorough investigation into whether LexSemTm sense frequencies can be used in place of SemCor, and if so whether they are superior to SemCor sense frequencies for lemmas with few occurrences in SemCor.

5.3 Replacing SemCor Sense Frequencies

5.3.1 Introduction

Now that we have described the creation of LexSemTm, we proceed with our experiments using this dataset to address our remaining research questions. The first such question is in regards to whether unsupervised all-words sense distribution learning can be used to replace existing sense resources such as SemCor. We have already shown in Section 5.2 that LexSemTm has greater coverage over polysemous WordNet lemmas than SemCor, however we have yet to establish whether the quality of data in LexSemTm is at least on par with SemCor, or possibly even superior to SemCor.

In order to address this general question relating to replacing sense frequency resources, we ask two specific questions about LexSemTm and SemCor: (1) can sense frequency data from LexSemTm be used in place of SemCor? and (2) if LexSemTm can be used in place of SemCor, is there a threshold such that the data from LexSemTm is clearly superior to SemCor for lemmas with a frequency in SemCor less than that threshold?

As discussed in Section 1.1, we are mostly limiting our scope in this investigation

Lemma Set	Set Size	SemCor Frequencies
$L_{gsc}^{(1)}$	10	0
$L_{gsc}^{(2)}$	10	1–3
$L_{gsc}^{(3)}$	10	4–8
$L_{gsc}^{(4)}$	10	9–20
$L_{gsc}^{(5)}$	10	21+
L_{gsc}	50	0+
$L_{gsc}^{(2–5)}$	40	1+

Table 5.3: Summary of the size and the range of SemCor frequencies covered by each subset of L_{gsc}. These are the lemmas in GoldSemCor: our gold-standard dataset for evaluating the quality of simplex lemma sense distributions in LexSemTm relative to SemCor.

to English nouns, in order to make the analysis and cost of gathering labelled data manageable. Similarly, we further narrow our scope in this section to simplex lemmas, in order to avoid dealing with possible confounding factors due to MWEs (which are analysed separately in Section 5.4). We leave addressing these questions for other classes of lemmas to future work.

5.3.2 Experimental Setup

Creation of GoldSemCor

In order to be able to answer our two questions regarding how LexSemTm compares to SemCor, we created a gold-standard evaluation dataset containing lemmas with a range of frequencies in SemCor. We refer to this gold-standard dataset as GoldSemCor, and the set of lemmas it contains as L_{gsc}. For each lemma in L_{gsc}, the dataset contains 100 usage sentences from EnWiki annotated with WordNet senses, and a sense distribution corresponding to these annotations.

In order to create L_{gsc}, which we desire to be a diverse set of lemmas covering a range of SemCor frequencies, we first obtained a list of all polysemous, simplex nouns in WordNet. In order to reduce the cost of obtaining labelled data and avoid needing to control for LexSemTm frequency, we then filtered out all lemmas with a LexSemTm frequency less than 5,000. This allows us to focus our analysis on the part of LexSemTm where we are most confident of sense distribution quality. Next we split the remaining lemmas into 5 groups of approximately equal size based on SemCor frequency. We randomly sampled 10 lemmas from each group, giving us the lemma sets $L_{gsc}^{(1)}$ through to $L_{gsc}^{(5)}$. Finally, L_{gsc} was obtained by taking the union of these sets, giving 50 lemmas in total. In addition, we define the set of all lemmas except from the first group as $L_{gsc}^{(2–5)}$ (those lemmas in L_{gsc} with at least one occurrence

in SemCor). These sets, and their SemCor frequency ranges, are summarised in Table 5.3.

For each lemma in L_{gsc} we then randomly sampled 100 sentences containing the lemma from EnWiki, as in Section 5.2.2. These were annotated using Amazon Mechanical Turk (AMT), as described in Section 3.1.5. This was done by splitting the 100 sentences of each lemma in L_{gsc} into 4 batches to be annotated. In addition, we created 2 control sentences for each lemma, and added them to all batches of that lemma. These control sentences were created manually such that they had a relatively clear correct sense (to the extent that this was possible), and are listed in Appendix A.[13] Therefore, in total we had 200 batches each consisting of 27 items, and every batch was annotated by 10 separate workers.

For each sentence to be annotated, workers were asked to select exactly one sense, and for each sense they were provided with the sense gloss from WordNet,[14] along with a list of hypernyms and synonyms of the sense from WordNet. Workers were instructed to select the sense they believed was most likely, if they thought a sentence was ambiguous. We did this, rather than allowing workers to select an "invalid" option or to select multiple senses, since our aim is to estimate sense frequencies; having an "invalid" option would make interpretation of the results more difficult, and we believe forcing workers to annotate completely ambiguous sentences with their most likely sense will still provide useful information, based on their belief of how frequent each sense is in general. For more details on the exact interface provided to workers, see Appendix A.

For each lemma we then inferred a single annotation for each sentence based on the output of AMT, using MACE (see Section 3.1.5). We ran MACE separately for each lemma using the AMT output of its batches, with the control sentences and their "correct" labels included to guide MACE.[15] Finally, for each lemma we took the MACE output labels for each of its 100 EnWiki sentences, and from these produced a gold-standard sense distribution using maximum likelihood estimation.

Evaluation of LexSemTm against SemCor

Given our GoldSemCor gold-standard evaluation dataset, we now describe our evaluation of LexSemTm against SemCor. First we generated a SemCor-based

[13]Note that in a small number of cases this was not completely possible. For example, the two senses for *flora* in WordNet have the glosses *all the plant life in a particular region* and *a living organism lacking the power of locomotion*; these are similar enough that even our carefully constructed control sentences were still ambiguous to annotators.

[14]Example sentences from WordNet were included with the gloss only when the example contained the target lemma, rather than a different lemma in the synset.

[15]In a small number of cases where we decided that a control sentence was too ambiguous, we excluded it. In addition, in a small number of cases where the same worker annotated a control sentence with more than one sense over multiple batches, we set their annotation of the control sentence to an extra "invalid" value in the MACE input.

sense distribution for each lemma in L_{gsc}. For those with at least one occurrence in SemCor (those in $L_{gsc}^{(2-5)}$), these were obtained by normalising the SemCor count for each WordNet sense. On the other hand, for lemmas with no occurrences in SemCor (those in $L_{gsc}^{(1)}$), we first assigned one count to its first-listed sense in WordNet, and then applied the same procedure as for the other lemmas. Finally, we evaluated the LexSemTm and SemCor-based sense distributions for each lemma in L_{gsc} using the JSD and ERR metrics. These metrics were calculated using the GoldSemCor gold-standard sense distributions. These sense distribution quality scores can be used to address our two questions relating to whether LexSemTm can replace SemCor, by analysing how the LexSemTm and SemCor-based distributions compare as a function of SemCor frequency.

Because this evaluation based on GoldSemCor can only provide answers for a limited set of lemmas in LexSemTm— namely, English nouns with a LexSemTm frequency of at least 5,000 — we performed a second, broader evaluation. This second evaluation was performed on all polysemous, English simplex lemmas in LexSemTm with at least one occurrence in SemCor. Firstly, in order to control for polysemy (which varies significantly for common versus rare lemmas), we split these lemmas into three sets based on polysemy: (1) lemmas with low polysemy (polysemy 2 or 3); (2) lemmas with medium polysemy (polysemy 4 to 6); and (3) lemmas with high polysemy (polysemy 7 and above). Then we partitioned each of these sets into 10 bins based on LexSemTm frequency, by rounding the frequency to the nearest 1,000, up to a maximum of 10,000.[16] Finally, we calculated the JSD of the LexSemTm distributions for all lemmas in each bin, using the corresponding SemCor-based distributions (calculated as in the previous evaluation) as proxy gold-standards. The results from this secondary evaluation allow us to investigate how the quality of LexSemTm sense distributions varies based on LexSemTm frequency.

5.3.3 Results

The results of our comparison of LexSemTm and SemCor-based distributions over the lemmas in L_{gsc} are listed in Table 5.4. For each subset of the L_{gsc} lemmas listed in Table 5.3, we list the average JSD and ERR from both methods, which we compare pairwise (using two-sided Wilcoxon signed rank tests) in order to test whether there is a statistically significant difference.

In addition, the results of our secondary evaluation over all polysemous, English simplex lemmas in LexSemTm with at least one SemCor occurrence are displayed in Figure 5.1. For each polysemy-based partition, and each corresponding bin, we provide a boxplot of the JSD values (relative to the SemCor-based proxy gold-standards) of the lemmas in the bin.

[16]This means that the first bin contained all lemmas trained on fewer than 500 usages, the second bin contained all lemmas trained on between 500 and 1,500 usages, and so on up to the final bin, which contained all lemmas trained on at least 9,500 usages.

Lemma	**JSD**		**ERR**	
	LexSemTm	**SemCor**	**LexSemTm**	**SemCor**
$L_{gsc}^{(1)}$	**.100±.080**	.615±.407 ($p = .013$)	**.902±.271**	.406±.470 ($p = .027$)
$L_{gsc}^{(2)}$	**.203±.169**	.214±.250 ($p = .959$)	.592±.432	**.735±.367** ($p = .465$)
$L_{gsc}^{(3)}$	**.100±.049**	.103±.133 ($p = .878$)	.699±.379	**.847±.275** ($p = .225$)
$L_{gsc}^{(4)}$	**.148±.069**	.235±.166 ($p = .114$)	.694±.394	**.711±.382** ($p = .917$)
$L_{gsc}^{(5)}$	.162±.121	**.156±.131** ($p = .721$)	.756±.352	**.904±.246** ($p = .285$)
L_{gsc}	**.142±.113**	.265±.301 ($p = .046$)	**.728±.383**	.720±.397 ($p = .587$)
$L_{gsc}^{(2-5)}$	**.153±.118**	.177±.184 ($p = .591$)	.685±.395	**.799±.332** ($p = .145$)

Table 5.4: Evaluation of LexSemTm versus SemCor sense distributions over various subsets of L_{gsc}. All JSD and ERR metric values were calculated relative to the gold-standard sense distributions in GoldSemCor. For each subset of L_{gsc} we list average metric values for the sense distributions from each method, as well as p values from comparing the metric values using two-sided Wilcoxon signed-rank tests.

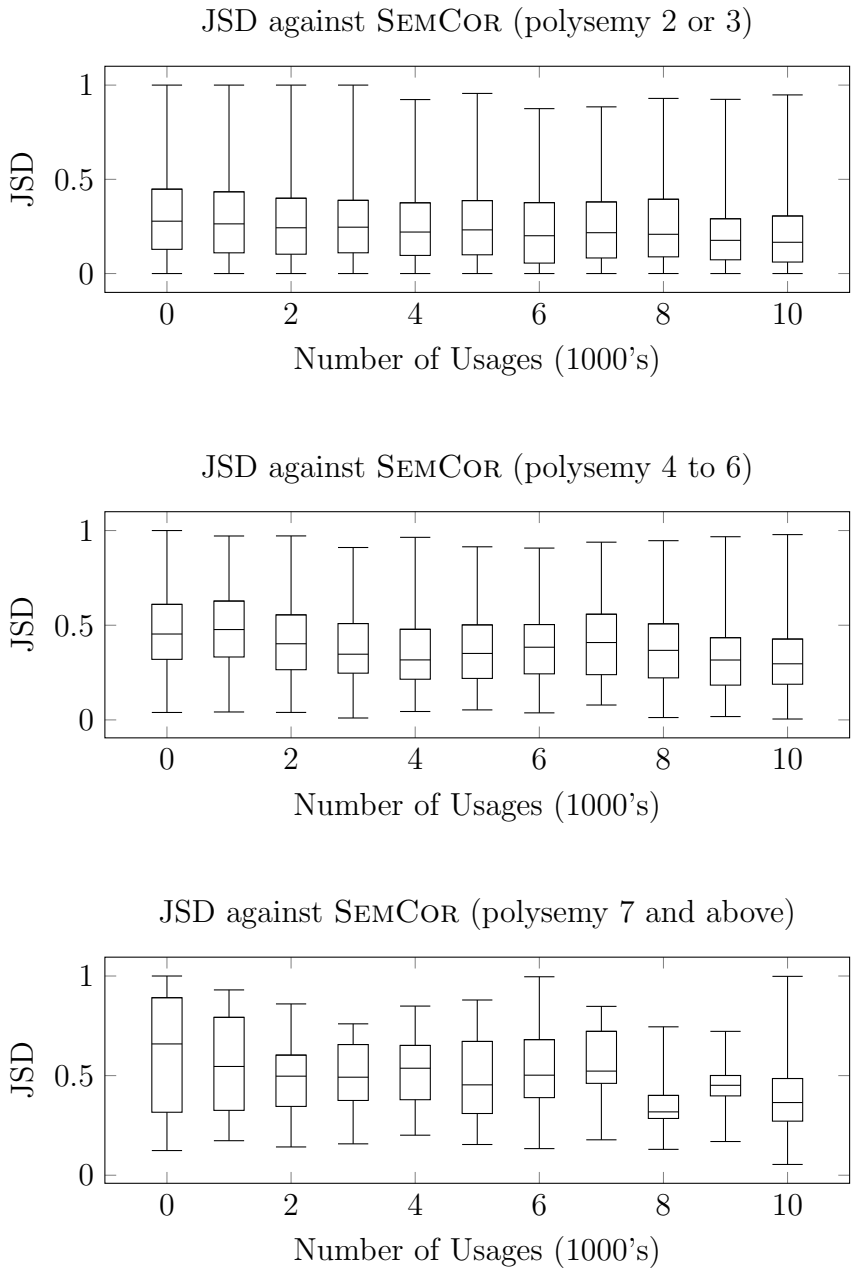

Figure 5.1: Boxplots of the distributions of JSD values of English simplex LexSemTm sense distributions, using SemCor as a proxy gold-standard. The data was split by polysemy, as well as LexSemTm frequency (the number of usages that LexSemTm was trained on). For each polysemy range in the figure, lemmas were binned by their LexSemTm frequency to the nearest 1,000 (so for example, the first bin contains lemmas with frequency less than 500, and the second bin contains lemmas with frequency between 500 and 1,500), and all lemmas with frequency greater than 9,500 were placed in the final bin.

5.3.4 Discussion

Firstly, we can observe from the results in Table 5.4 that LexSemTm distributions strongly outperform SemCor-based ones for lemmas with no occurrences in SemCor (those in $L_{gsc}^{(1)}$). This is exactly what we would be expect, given that the SemCor-based distributions for these lemmas are somewhat arbitrary (they are determined by which sense is listed first in WordNet), and this result holds true according to both JSD and ERR metrics ($p < 0.05$ in both cases). This is very encouraging, given the approximately 1,564 high frequency (trained on at least 5,000 usages) polysemous simplex lemmas in LexSemTm that do not occur at all in SemCor; we now have strong evidence that LexSemTm provides more accurate sense frequency data for this large set of lemmas.

On the other hand, for the lemmas in L_{gsc} that occur in SemCor (those in $L_{gsc}^{(2-5)}$), we do not have clear evidence that their LexSemTm sense distributions differ in quality compared to SemCor-based ones. None of the comparisons between LexSemTm and SemCor for the lemmas in $L_{gsc}^{(2-5)}$ (as well as the corresponding subsets) were found to be statistically significant ($p > 0.1$ in all cases). Given the mean values of the sense distribution quality metrics listed in Table 5.4, it seems that the LexSemTm distributions may be slightly better in terms of overall distribution shape (measured by JSD), and the SemCor distributions may be slightly better in terms of the first sense (measured by ERR). However, even assuming that this is the case, these difference are extremely small compared to the variance in both metrics, and do not appear to be statistically significant.

Adding to this, it appears from the results of our secondary evaluation in Figure 5.1 that the LexSemTm distributions are fairly similar in quality for lemmas with LexSemTm frequency less than 5,000 compared to lemmas with frequency at least 5,000. On the one hand, we can see from the boxplots that for each polysemy range, the average JSD tends to increase slightly as the number of usages is decreased. However, this increase is small compared to the variance in each bin. In particular, even if we compare the JSD values of lemmas trained on the fewest usages to those trained on the most usages, in most cases the median of the former lies within the standard range (within the second or third quartiles) of the latter. This is true even for the bins of lemmas trained on fewer than 500 usages.[17] Furthermore, the difference in median JSD between the first and last bins for each polysemy range (ranging from 0.1 to 0.2) is small compared to the average difference in JSD between LexSemTm and SemCor-based distributions for the lemmas in $L_{gsc}^{(1)}$ (those missing from SemCor).

This conclusion is corroborated by the results of our bootstrapping experiment with HCA-WSI on the lemmas in L_{bnc} (from Section 4.3), which were previously

[17]Except arguably for the lemmas with polysemy 7 and above. However the variance is very large in this case, because most of these bins contain relatively few lemmas; each of the bins in this partition (except for the last) contain only around 20 lemmas on average.

displayed in Figure 4.6. In these results, we observed that when the number of usages of each lemma in L_{bnc} was reduced to 500, the mean change in JSD was almost always less than 0.02, and never greater than 0.04, and the standard deviation of the change in JSD was almost always less than 0.02, and never greater than 0.05. These changes in JSD are small compared to the difference in JSD between LexSemTm and SemCor for every subset of L_{gsc}, except $L^{(3)}_{gsc}$. This observation, together with the results of our secondary evaluation in Figure 5.1, strongly suggest that our conclusions regarding how LexSemTm and SemCor compare can safely be extended at least to lemmas with LexSemTm frequency greater than 500, if not to all lemmas in LexSemTm.

Returning to the questions posed in the introduction of this section, we can answer the first by concluding that it seems reasonable to replace SemCor sense frequencies with LexSemTm in general. Firstly, the primary results in Table 5.4 suggest that the sense distributions in LexSemTm are always at least on-par with those based on SemCor for lemmas with LexSemTm frequency at least 5,000. Or at the very least, there is insufficient evidence to believe that they are worse. Secondly, based on the previous results contained in Figure 4.6, the variation in sense distribution quality when the number of usages is reduced as low as 500 is not significant enough to change this conclusion. Finally, the results in Figure 5.1 strongly suggest that sense distribution quality for lemmas trained on fewer than 500 usages (first bin in each partition) is not significantly different than for lemmas trained on between 500 and 1,500 usages (second bin in each partition). Given this, it seems reasonable to conclude that LexSemTm sense distributions are at least on-par with SemCor-based distributions over all LexSemTm frequencies.

Regarding the second question, which asked whether LexSemTm sense distributions are ever superior to SemCor-based ones, it appears that this is only clearly the case for lemmas missing from SemCor. As for the previous question, given the multiple sources of evidence indicating that sense distribution quality is comparable for lemmas with low versus high LexSemTm frequency, it seems reasonable to conclude that this result holds across all LexSemTm frequencies. Furthermore, given the very significant JSD difference between LexSemTm and SemCor-based distributions in $L^{(1)}_{gsc}$ (on average over 0.5), extending the conclusion to all lemmas seems particularly justified in this case. This is very significant, because there are approximately 14,000 polysemous simplex lemmas in LexSemTm that are missing from SemCor, which accounts for approximately half of all polysemous simplex lemmas in WordNet!

It should be noted that our secondary evaluation is fairly rough, since we are using SemCor as a proxy gold-standard. This is methodologically questionable, due to the limitations of SemCor discussed in Section 3.1.2. However, we justify this by the fact that we are evaluating over a very large number of lemmas, so the resulting noise can likely be safely ignored. It would be better to obtain gold-standard annotated data for lemmas with fewer than 5,000 usages, although we have chosen not to do this due to the annotation cost. We leave this more thorough analysis to future work.

Now that we have argued that LexSemTm can be used in place of SemCor, that

they are roughly on-par for lemmas in SemCor, and that LexSemTm distributions are strongly superior to SemCor-based distributions for lemmas missing from SemCor, we have confirmation that unsupervised all-words sense distribution learning can successfully be used to supplement existing sense resources. This answers our second core research question. Now we turn to answering our final research question, regarding whether sense distribution learning can also be applied to MWEs.

5.4 Evaluating Multiword Expression Sense Distributions

5.4.1 Introduction

We have one remaining core research question unanswered, which is whether sense distribution learning can also be applied to MWEs, and if so how does this compare to simplex sense distribution learning? In Section 5.2 we described the creation of our LexSemTm dataset, containing WSI and sense distribution information over a large number of lemmas, including MWEs, from the WordNet's of multiple languages. In this process, we proposed two methods for extending HCA-WSI sense distribution learning to MWEs, which we used to add two corresponding sets of MWE sense distributions to LexSemTm. These extensions were categorised by two simple but novel general-purpose, unsupervised methods for identifying usages of known MWEs.

In this section we use these two sets of MWE sense distributions in the English section of LexSemTm to address this remaining question, which we break down into three smaller questions: (1) what is the impact of our different MWE usage identification methods? (2) how do the quality of our LexSemTm MWE sense distributions compare to those obtained from simple baseline and benchmark approaches (such as using SemCor data)? and (3) how do the MWE and simplex sense distributions in LexSemTm compare, both in terms of sense distribution quality and the shapes of the distributions?

It should be noted that we narrow our scope in this investigation to English nouns, for the same reasons again as outlined in Section 1.1 (due to the cost of obtaining labelled data, and ease of analysis). In addition, it should be observed that, as far as we are aware, general purpose sense learning on MWEs, including WSD, WSI, first sense learning, and sense distribution learning, are all novel tasks.[18]

[18]There are some minor and very specific exceptions to this that were outlined in Section 2.4, including named entity recognition, supersense tagging, and disambiguating between literal and idiomatic interpretations of MWEs.

Lemmas	Set Size	Description
L_{re}	46	High recall-identified MWE lemmas with at least 5,000 usages
L_{pr}	22	High precision-identified MWE lemmas with at least 5,000 usages
L_{gsc}	50	Simplex lemmas from Section 5.3
L_{int}	22	$L_{re} \cap L_{pr}$
L_{diff}	24	$L_{re} \setminus L_{pr}$
L_{union}	46	$L_{re} \cup L_{pr}$
L^{lp}_{union}	44	Lemmas in L_{union} with polysemy 2 or 3
L^{lp}_{gsc}	28	Lemmas in L_{gsc} with polysemy 2 or 3
L^{sc}_{int}	13	Lemmas in L_{int} present in SemCor
L^{sc}_{diff}	5	Lemmas in L_{diff} present in SemCor
L^{sc}_{union}	18	Lemmas in L_{union} present in SemCor
L^{sc*}_{int}	9	Lemmas in L_{int} missing from SemCor
L^{sc*}_{diff}	19	Lemmas in L_{diff} missing from SemCor
L^{sc*}_{union}	28	Lemmas in L_{union} missing from SemCor

Table 5.5: Summary of the different sets of lemmas used in our MWE evaluation experiments. Note that $L_{int} = L_{pr}$ and $L_{union} = L_{re}$; these sets are named apart for clarity.

5.4.2 Experimental Setup

Creation of GoldMWE

As in Section 5.3, we require the use of additional gold-standard evaluation data to perform our evaluations in this section. In this case, since we are evaluating MWE sense distributions, we require gold-standard annotated data for MWE lemmas. In addition, because we are evaluating sense distributions based on two different MWE identification methods, we require gold-standard data corresponding to usages sampled based on both methods. We refer to this MWE gold-standard dataset as GoldMWE.

In order to create this dataset, we first obtained a list of all polysemous MWE nouns from WordNet with a LexSemTm frequency of at least 5,000 from either identification method. In doing this we obtained separate lists of high recall- and high precision-sampled lemmas from LexSemTm. In total this gave us 46 (respectively 22) high recall-sampled (respectively high precision-sampled) MWE lemmas. These lemma sets, which we name L_{re} (respectively L_{pr}), and others based on these that we refer to below, are summarised in Table 5.5. It should be noted that, because all high precision-identified lemmas can also be identified using the high recall method, L_{pr} is a subset of L_{re}.

Next, we produced our gold-standard annotated data for lemmas in both L_{pr} and

L_{re}, based on usages identified with the respective methods, using Amazon Mechanical Turk (AMT: see Section 3.1.5). This was done almost identically to Section 5.3: for each lemma in L_{re} (respectively L_{pr}), we randomly sampled 100 high recall-identified (respectively high precision-identified) sentences from EnWiki, and created a set of control sentences. However unlike Section 5.3, we provided two additional annotation options for each sentence: (1) annotators could label sentences as having a meaning that was "clearly separate" from all those listed if the sentence was a usage of the MWE and the meaning was clear, but completely distinct from all of the WordNet senses; or (2) annotators could label sentences as "not a valid usage" if the sentence was not a usage of the MWE. The first option was included to deal with cases where the usage was simply compositional and therefore not covered by WordNet (for example a usage of *red tape* that literally referred to red-coloured tape). The second option was included to deal with false positives from our MWE usage identification methods (for instance, refer to the *training school* example in Section 5.2.3). In addition, we included three control sentences for each lemma rather than two: the first two control sentences were clear usages of specific senses (as in Section 5.3), and the third control sentence was an artificial MWE identification false positive (as in the *training school* example). These control sentences are listed in Appendix A.

As in Section 5.2, we split the 100 sentences for each lemma into 4 batches to be annotated and added the control sentences to each. In this case that meant 272 batches in total (since the lemmas in L_{int} were annotated twice), each consisting of 28 items, and again every batch was annotated by 10 separate workers. For more details on the exact interface provided to workers, see Appendix A.

The output from AMT was processed using MACE (see Section 3.1.5) almost identically to Section 5.3. However, in this case we have the complication of the extra "clearly separate" and "not a valid usage" labels. In practice we found that despite our instructions, AMT workers used these labels interchangeably (based on the annotations of our false positive control sentences). Therefore before we ran MACE on the AMT output, we merged these two labels together into a single "invalid" label. All other details in running MACE, including the use of the control sentences, were identical to Section 5.3.

Finally, the sense labels of the EnWiki sentences for each lemma were converted into gold-standard sense distributions. In order to do this we first removed the sentences labelled by MACE as "invalid", and then performed maximum likelihood estimation on the remaining sense label counts. In addition, we recorded which sentences were labelled as "invalid" for each lemma, which we include in GoldMWE in order to facilitate our evaluation of MWE sense distributions. In total, this gave us 68 gold-standard sense distributions: one for each lemma in L_{diff} (based on high recall identification), and two for each lemma in L_{int} (based on both identification methods).

Evaluation of MWE Identification Methods

In order to answer our first question, relating to the impact of the two MWE identification methods, we performed two different comparisons. Firstly, we compared the proportion of usages labelled as "invalid" for the lemmas in L_{int} and L_{diff}, based on both identification methods. This allows us to measure the impact of the methods in terms of how much noise they introduce in the form of false positives. Secondly, we directly compared the sense distributions resulting from each identification method. We not only compared the two sets of gold-standard sense distributions for the lemmas in L_{int}, but we also compared the two sets of sense distributions from LexSemTm over all of L_{union}.[19] All of these sense distribution comparisons were done by calculating the JSD (see Section 3.2.4) between each pair of distributions.

However, we have no point of reference for these comparison JSD values; we have no idea how low we would expect these values to be by chance, for example if we were comparing distributions of different lemmas. Therefore, we also calculated a set of comparison JSD values, by comparing the gold-standard distributions of all pairs of lemmas in L_{union} with equal polysemy.[20]

Evaluation of MWE LexSemTm Sense Distributions

Next we turn to the second question, which asked how the sense distributions for MWE lemmas from LexSemTm compare to available benchmarks and baselines. In this evaluation we used one benchmark and one baseline distribution for each lemma in L_{union}. The benchmark we used was based on SemCor counts: we calculated SemCor-based sense distributions for all lemmas in L_{union} identically to Section 5.3. Our baselines were simple uniform distributions over all available WordNet senses for each lemma. We used this baseline because we found that our automatically learnt distributions were very flat on average (this result is discussed in Section 5.4.4), so we suspected that the strong performance of our distributions in terms of JSD may have simply been due to them being close to uniform.

It should be noted that of the 46 lemmas in L_{union}, 28 of them have no occurrences in SemCor. This is significant because we found in Section 5.3 that the SemCor benchmark is very weak for such lemmas (as would be expected). Therefore, we controlled for this by also performing this evaluation separately on the subsets of L_{union}, L_{int}, and L_{diff} with SemCor frequency greater than zero, and with SemCor frequency zero. (these restricted sets are summarised in Table 5.5).

For all three kinds of candidate sense distributions (LexSemTm, SemCor benchmark, and uniform baseline), we evaluated quality using both JSD and ERR metrics,

[19] This is possible because all but one of the lemmas in L_{diff} is present in the high precision subset of LexSemTm, where they were trained on 2878.8±1344.4 usages on average. Note that the missing lemma was excluded from the comparison.

[20] More specifically, we calculated the JSD between the high recall gold-standard distributions of all distinct pairs of lemmas l_a and l_b in L_{union} such that $l_a \neq l_b$ and polysemy(l_a) = polysemy(l_b).

relative to the gold-standard distributions in GoldMWE. Because the results from our high recall versus high precision identification methods were found to be extremely similar (discussed in Section 5.4.4), we only used our high recall-based LexSemTm and gold-standard distributions in this evaluation.

Comparison of Simplex and MWE Sense Distributions

Finally we address the third question, which asked how the LexSemTm sense distributions for MWE lemmas compare to those for simplex lemmas. We approached this from two perspectives. First we compared the absolute values of the quality score metrics (JSD and ERR) obtained for simplex versus MWE lemmas. We compared the quality scores of the MWE lemmas in L_{union} (evaluated against GoldMWE) with the quality scores of the simplex lemmas in L_{gsc} (evaluated against GoldSemCor, as in Section 5.3). This allows us to investigate whether MWE sense distribution learning achieves either higher or lower performance in absolute terms than simplex sense distribution learning, which would suggest that one task is inherently more difficult than the other.

Secondly, we compared the shapes of the LexSemTm and gold-standard (from GoldSemCor and GoldMWE) sense distributions for the lemmas in L_{gsc} to those lemmas in L_{union}. In this evaluation, sense distribution shape was measured using entropy. The purpose of this secondary evaluation is to investigate whether MWE sense distributions are any different in shape on average compared to simplex sense distributions. If they were systematically different in shape, this would suggest that the two tasks are different enough to possibly warrant separate methodology.

In both of these evaluations, we need to control for polysemy, because the lemmas in L_{gsc} are on average much more polysemous than the lemmas in L_{union}.[21] We do this by performing these evaluations in addition only on the lemmas in L_{gsc} and L_{union} with low polysemy (polysemy of 2 or 3). We refer to these restricted sets as L^{lp}_{gsc} and L^{lp}_{union} respectively.

As in the previous evaluation, only high recall-based MWE distributions from LexSemTm were used.

5.4.3 Results

First we list the results of our evaluation of the high recall versus high precision MWE usage identification methods. In Table 5.6 we list the proportion of sentences labelled as "invalid" by MACE that were sampled using our high recall and high precision methods, and then in Table 5.7 we list the average JSD values from comparing the high recall versus high precision sense distributions. As a reference for

[21] The average polysemy of the lemmas in L_{gsc} is 4.42±3.18, compared to 2.24±0.60 for the lemmas in L_{union}.

Lemmas	**High Recall**	**High Precision**
L_{int}	.068±.075	.018±.043 ($p = .0003$)
L_{diff}	.221±.220	—

Table 5.6: Results of our comparison of high recall versus high precision MWE identification methods, in terms of the proportion of high recall- versus high precision-identified sentences labelled as "invalid" usages. The average proportion of usages labelled as "invalid" is listed for the lemmas in L_{int} and L_{diff}.

Lemmas	**LexSemTm**	**Gold-Standard**
L_{int}	.00008±.0002	.007±.017
L_{diff}	.0004±.0006	—

Table 5.7: Results of our comparison of high recall versus high precision MWE identification methods, in terms of the similarity between LexSemTm and gold-standard distributions resulting from either evaluation method. For each kind of distribution (LexSemTm or gold-standard) and each set of lemmas (L_{int} or L_{diff}) we list the average JSD between the distributions resulting from either identification method. In the case of the LexSemTm distributions of the lemmas in L_{diff}, this comparison was done on the 23 L_{diff} lemmas present in the high precision subset of LexSemTm (which were trained on 2878.8±1344.4 usages on average).

these values, the average JSD from our comparison of all pairs of lemmas with equal polysemy was 0.221±0.220.

Next, we list the results of our evaluation of MWE LexSemTm sense distributions against the SemCor benchmark and uniform baseline. We list results based on the JSD metric in Table 5.8, and results based on the ERR metric in Table 5.9. Note that we do not list uniform baseline results in Table 5.9, because these are equal to the SemCor benchmark results for the ERR metric.[22]

Finally, we list the results of our evaluation comparing MWE and simplex sense distributions. In Table 5.10 we list the average JSD and ERR metrics obtained for our MWE and simplex LexSemTm sense distributions, and in Table 5.11 we list our results comparing the shapes of our MWE and simplex sense distributions. Unlike in all of our previous evaluations, the p-values listed here are from Wilcoxon rank sum tests rather than Wilcoxon signed rank tests.[23]

[22]This is because in both cases the first listed sense in WordNet is always chosen as the first sense (senses in WordNet are listed in descending order by SemCor frequency).

[23]We use Wilcoxon rank sum tests here because unlike in previous experiments, we are not comparing paired data.

Lemmas	**LexSemTm**	**SemCor**	**Uniform**
L_{union}	**.160±.149**	.291±.324 ($p = .036$)	.178±.136 ($p = .002$)
L_{int}	**.171±.172**	.238±.297 ($p = .548$)	.179±.129 ($p = .101$)
L_{diff}	**.150±.297**	.339±.340 ($p = .032$)	.176±.141 ($p = .010$)
L^{sc}_{union}	.214±.175	**.139±.241** ($p = .184$)	.231±.124 ($p = .048$)
L^{sc}_{int}	.230±.195	**.164±.275** ($p = .382$)	.227±.128 ($p = .221$)
L^{sc}_{diff}	.173±.097	**.077±.079** ($p = .225$)	.242±.114 ($p = .080$)
L^{sc*}_{union}	**.125±.116**	.388±.334 ($p = .001$)	.143±.131 ($p = .029$)
L^{sc*}_{int}	**.085±.070**	.346±.295 ($p = .051$)	.109±.096 ($p = .260$)
L^{sc*}_{diff}	**.143±.128**	.408±.349 ($p = .007$)	.159±.142 ($p = .064$)

Table 5.8: Results of our evaluation of MWE sense distributions relative to the gold-standard distributions in GoldMWE, in terms of the JSD metric. Evaluation was done for LexSemTm sense distributions, as well as SemCor benchmark and uniform baseline sense distributions, over various subsets of the lemmas in L_{union}. All p-values are from two-sided Wilcoxon signed rank tests, comparing the JSD values obtained for the benchmark or baseline distributions to those obtained for the LexSemTm sense distributions.

Lemmas	**LexSemTm**	**SemCor**
L_{union}	.766±.354	**.767±.380** ($p = .891$)
L_{int}	.742±.386	**.795±.341** ($p = .583$)
L_{diff}	**.788±.319**	.742±.411 ($p = .594$)
L^{sc}_{union}	.748±.402	**.911±.247** ($p = .116$)
L^{sc}_{int}	.692±.442	**.877±.283** ($p = .225$)
L^{sc}_{diff}	.895±.209	**1.000±.000** ($p = .317$)
L^{sc*}_{union}	**.778±.318**	.675±.420 ($p = .193$)
L^{sc*}_{int}	**.816±.271**	.677±.381 ($p = .398$)
L^{sc*}_{diff}	**.760±.337**	.674±.437 ($p = .445$)

Table 5.9: Results of our evaluation of MWE sense distributions relative to the gold-standard distributions in GoldMWE, in terms of the ERR metric. This was done identically to Table 5.8 with the JSD metric, except that ERR values for the uniform baseline are not listed, since they are identical to the SemCor benchmark in this case.

5.4.4 Discussion

First we discuss the results of our evaluation of our high recall and high precision MWE usage identification methods. On the one hand we can see from Table 5.6 that

Lemmas	**JSD**	**ERR**
L_{union}	.160±.149	.766±.354
L_{gsc}	.142±.113 ($p = .786$)	.728±.383 ($p = .846$)
L^{lp}_{union}	.138±.107	.800±.323
L^{lp}_{gsc}	.120±.124 ($p = .349$)	.731±.393 ($p = .703$)

Table 5.10: Results of our comparison of MWE and simplex sense distributions, in terms of the absolute values of quality metrics. For each set of lemmas (simplex lemmas in L_{union} and MWE lemmas in L_{gsc}) we list average JSD and ERR metrics of the LexSemTm sense distributions, and compare quality values between the lemma sets using two-sided Wilcoxon rank sum tests.

Lemmas	**LexSemTm**	**Gold-Standard**
L_{union}	.975±.050	.473±.365
L_{gsc}	1.648±.810 ($p = .00002$)	1.065±.697 ($p = .00001$)
L^{lp}_{union}	.979±.042	.488±.365
L^{lp}_{gsc}	1.062±.353 ($p = .393$)	.645±.402 ($p = .110$)

Table 5.11: Results of our comparison of MWE and simplex sense distributions, in terms of the shapes of the distributions. For each set of lemmas (simplex lemmas in L_{union} and MWE lemmas in L_{gsc}) and the corresponding subsets with low polysemy, we list the average entropy of the LexSemTm sense distributions, and compare the entropy values between the corresponding simplex and MWE lemma sets using two-sided Wilcoxon rank sum tests.

there is substantially more junk in the high recall-identified sentences, compared to the high precision-identified ones: the proportion of "invalid" usages for the lemmas in L_{int} is approximately four times as high for high recall-identified sentences as compared with high precision-identified ones, which was found to be clearly statistically significant ($p < 0.001$). For the lemmas in L_{diff}, the amount of junk introduced by high recall identification was significantly higher again, which is to be expected given that by definition these are lemmas for which a significant proportion of the high recall-identified usages could not be identified by the high precision method.

On the other hand, it can be seen from Table 5.7 that the sense distributions resultant from both identification methods are extremely similar. The average JSD between high precision and high recall gold-standard distributions for the lemmas in L_{int} is very low (0.007±0.017), and most of this variance came from a small number of outliers.[24] These divergences are tiny compared to our comparison value from averag-

[24]If we remove the three lemmas with the greatest divergence, this average JSD reduces to

ing over pairs of lemmas with equal polysemy, which was 0.221±0.220. Furthermore, the LexSemTm sense distributions resultant from each identification method have even less divergence. Incredibly, this holds true even for the lemmas in L_{diff}, where the high precision method was working with a relatively small amount of data (even in the worse case scenario, the maximum JSD observed was just 0.002).

We can conclude that using high recall identification introduces lots of junk compared to high precision identification, but in the vast majority of cases does not appear to bias the rest of the data. In addition, we can conclude that HCA-WSI is very robust to the resultant noise from these identification methods, and that the automatically learnt sense distributions (those in LexSemTm) resulting from either method are nearly indistinguishable. Given this strong result we chose to only work with high recall-based sense distributions in subsequent evaluations.

Next we discuss the results from comparing our MWE LexSemTm sense distributions with the SemCor benchmark and the uniform baseline. In terms of JSD, our LexSemTm distributions strongly outperform the SemCor benchmark on average over all of the lemmas in L_{union}, which is statistically significant ($p < 0.05$). In addition, it outperforms the baseline uniform distributions in terms of JSD, with a smaller on average but more consistent difference in JSD ($p < 0.01$). Similar results hold over L_{int} and L_{diff}, although not all results are statistically significant on these sets (possibly due to less statistical power on the smaller sets).

However, when we control for whether the lemmas are in SemCor or not, we see a significant change in behaviour for the JSD metric. For lemmas not in SemCor, we observe that LexSemTm beat the SemCor benchmark even more strongly. On the other hand for the lemmas in SemCor, we observe that the SemCor benchmark beat LexSemTm on average, although the set of lemmas in SemCor (L^{sc}_{union}) is relatively small, and we weren't able to establish statistical significance for this result ($p > 0.1$ in all cases).

In terms of ERR there is much less difference between LexSemTm and our benchmark and baseline distributions. The ERR values seemed about equal on average when evaluating over all lemmas, and the same general pattern as for JSD was observed when controlling for membership in SemCor. However, none of the differences observed according to the ERR metric were statistically significant ($p > 0.1$ in all cases).

These results are generally very similar to those we observed when we evaluated our LexSemTm distributions for simplex lemmas in Section 5.3. While it appears that the relative performance of the SemCor benchmark may be higher in this instance for the lemmas present in SemCor, compared to what was observed in Section 5.3 for simplex lemmas, we can't be sure of this due to the small size of L^{sc}_{union} and the lack of statistical significance.

Finally, we discuss the results of our comparison of MWE and simplex sense

0.002±0.002.

distributions. We can observe firstly from Table 5.10 that the JSD and ERR values obtained for both MWE and simplex sense distributions are very similar on average. This is true regardless of whether we control for polysemy or not, and in all cases the difference between JSD or ERR values of the MWE and simplex sense distributions is not statistically significant ($p > 0.3$ in all cases). Looking at Table 5.11, we can observe that in general our MWE sense distributions have lower entropy than simplex ones, which holds true for both the LexSemTm and gold-standard sense distributions. However this is to be expected, because as noted previously the MWE lemmas are much less polysemous on average.[25] When we control for polysemy and only look at lemmas with polysemy 2 or 3, this difference between MWE and simplex sense distributions all but disappears ($p > 0.1$ in both cases). However, interestingly we see that for all sets of lemmas, the gold-standard sense distributions consistently have lower entropy than the LexSemTm distributions. This suggests that HCA-WSI is systematically producing distributions that are too flat on average. This is an interesting result, which strongly suggests there is room for improvement in the simple topic–sense alignment component of HDP-WSI and HCA-WSI. However, further work based on this observation is beyond the scope of this thesis.

Returning to the initial questions posed in the introduction to this chapter, we can make some conclusions. First of all, we can answer the original question in the affirmative: MWE sense distribution learning appears to be achievable, and as far as we can tell the task seems to be comparable with simplex sense distribution learning in all important respects. This includes the relative performance on the task compared with the strong benchmark of using SemCor-based distributions, the absolute performance on the task in terms of JSD and ERR metrics, and the general shape of both the gold-standard and LexSemTm sense distributions. Given that to the best of our knowledge MWE sense distribution learning is a novel task, this is a very significant result.

In addition, we can conclude that there appears to be little impact in practice due to our sense identification methods. While the high recall method was found to introduce a significantly higher proportion of invalid usages, this was found to have a negligible impact upon the results of HCA-WSI. Furthermore, from comparing our two sets of gold-standard distributions, we can observe that the distribution over senses of the remaining usages once the invalid ones were removed was for the most part unchanged by the identification methods. Based on this, we can conclude that HCA-WSI seems to be robust to the exact method of MWE usage identification, which gives us further confidence in the quality of LexSemTm, and allows us to get away with just using the high recall-identified LexSemTm sense distributions.

Finally, from these results we can make a couple of auxiliary conclusions. First of all, we have evidence that HCA-WSI is producing sense distributions that are too flat on average, which may help motivate future work on improving the topic–sense

[25]Due to the definition of entropy, it tends to be higher for distributions over larger supports.

alignment method of HCA-WSI. Given that LexSemTm contains the WSI output of HCA-WSI for all lemmas, such improvements could directly be applied to update and improve LexSemTm. In addition, based on the strong similarity we observed between the high recall- and high precision-identified LexSemTm sense distributions for the lemmas in L_{diff} (where the high precision-identified LexSemTm distributions were trained fewer than 3,000 usages on average), we have further evidence that the quality of LexSemTm sense distributions is still high for lemmas trained on fewer than 5,000 usages. This strengthens our argument in Section 5.3 that our conclusions — regarding LexSemTm distributions being superior to SemCor-based distributions for lemmas missing from SemCor, and on-par with SemCor-based distributions otherwise — extend to lemmas with low frequency in LexSemTm.

5.5 Conclusion

In Section 5.2 we described the creation of our LexSemTm dataset, containing WSI output from HCA-WSI across the bulk of the vocabulary of English, Japanese, Italian, Mandarin, and Indonesian, which can be trivially aligned to any sense inventory containing glosses, as well as distributions over WordNet senses for the English lemmas. This dataset contains MWE as well as simplex lemmas, and in the process of creating it we proposed two simple but novel methods for identifying MWE usages. In addition, we showed that for English, LexSemTm had significantly higher coverage than SemCor over polysemous lemmas (approximately 88% versus 39%).

Then in Section 5.3 we evaluated the English sense frequency data in LexSemTm for polysemous simplex lemmas, relative to SemCor. We first demonstrated that LexSemTm distributions strongly outperform SemCor-based distributions for lemmas that are missing SemCor, and also that they are roughly on-par in quality for lemmas in SemCor. Furthermore, while our main evaluation was performed on lemmas where LexSemTm was trained on at least 5,000 usages, we provided multiple sources of evidence that these results appear to also hold for lemmas with LexSemTm frequency less than 5,000. This justifies supplementing SemCor with LexSemTm frequencies for most polysemous WordNet lemmas, and possibly replacing them altogether. However, we did not find any clear evidence of LexSemTm outperforming SemCor-based distributions for lemmas in SemCor, over any range of SemCor frequencies.

Finally, in Section 5.4 we provided a thorough evaluation of the MWE sense distribution data contained in LexSemTm. In the course of this analysis we concluded that HCA-WSI is robust to our different MWE usage identification methods, and that we can safely get away with just using data from the high recall method (which covers a greater range of lemmas). Furthermore, we concluded that sense distribution learning can successfully be done for MWEs, and that MWE sense distribution learning is comparable to simplex sense distribution learning in all important respects: the

relative performance compared to the SemCor benchmark is similar, the absolute performance as evaluated against gold-standard data using JSD or ERR metrics is about the same on average, and the sense distributions are very similar in shape on average. Finally, we observed that the distributions in LexSemTm appear to be systematically too flat, which we believe may motivate future work on refining the topic–sense alignment method used by HCA-WSI.

Based on these conclusions, we can finally answer the remaining research questions. The first remaining question asked whether unsupervised all-words sense distribution learning can be used to supplement or replace existing sense frequency resources. Based on our results and conclusions from Section 5.3, we can strongly justify supplementing SemCor with LexSemTm-based sense frequencies for lemmas missing from SemCor. Furthermore, we have have evidence to suggest that we can likely replace SemCor with LexSemTm-based sense frequencies altogether, although this latter conclusion was not as strongly justified. Regarding the other remaining question, which asked whether sense distribution learning could be extended to MWEs, not only does sense distribution learning using HCA-WSI appear to behave similarly for MWE lemmas as with simplex lemmas in all important respects, but it is also robust to how we identify MWE usages. However, given the restriction in the scope of our evaluations for the most part to English nouns (as in previous sections), we again need to hedge our answers, and note that they only apply strongly to this class of lemmas.

Finally, we have addressed the other core aim of our research, which was to apply unsupervised all-words sense distribution learning to create a language-wide multilingual sense frequency resource, which was satisfied by the creation of LexSemTm. In addition to this, we have created two accompanying gold-standard datasets: (1) GoldSemCor, containing usage sentences from EnWiki for a set of simplex WordNet lemmas over a wide and balanced range of SemCor frequencies, which are tagged with WordNet senses; and (2) GoldMWE, containing usage sentences from EnWiki for a set of MWE WordNet lemmas with usages sampled based on both MWE identification methods, which are labelled as to whether they are valid MWE usages or not, along with WordNet sense labels for those labelled as valid usages. These gold-standard evaluation datasets are a bonus outcome from this chapter that will support future sense distribution learning research, especially work aiming to supplement or replace SemCor, and performing usage identification or sense learning (including WSD and sense distribution learning) for MWEs.

Chapter 6

Conclusion

6.1 Summary

In this thesis we have extended and optimised existing sense distribution learning methods to produce HCA-WSI, an efficient method for unsupervised all-words sense distribution learning. This method was created by replacing the HDP topic modelling component of the previously state-of-the-art HDP-WSI method with HCA, a more efficient topic modelling algorithm. HCA-WSI was demonstrated to be consistently over an order of magnitude faster than HDP-WSI, and more robust with less random variation in sense distribution quality. In addition, we applied HCA-WSI vocabulary-wide across English, Japanese, Italian, Mandarin, and Indonesian to create LexSemTm, a new sense frequency resource of unprecedented size, containing data for both simplex and MWE lemmas. These respective outcomes directly address the primary and secondary aims of our research, which were to develop a method for unsupervised all-words sense distribution learning, and apply it language-wide to create a novel sense frequency resource. Furthermore, in addressing these aims we have answered our core research questions, which were:

1. What does a practical blueprint look like for efficiently applying sense distribution learning on a large scale, and achieving an optimal balance between accuracy and computation time?

In the conclusion of Chapter 4, we provided a template for applying HCA-WSI efficiently and achieving a reasonable accuracy versus computation time tradeoff. Specifically, we provided conservative lower bounds of the number of lemma usages and Gibbs sampling iterations needed for stable results, which we estimated as 5,000 to 10,000 and 300 respectively, and we showed that HCA-WSI is stable with respect to HCA hyperparameter settings, with 10 topics, burstiness turned on, and otherwise default settings recommended as a safe and efficient general setup.

2. To what extent can unsupervised all-words sense distribution learning be used to supplement or replace existing sense frequency resources?

In Chapter 5 we demonstrated that sense distributions from unsupervised all-words sense distribution learning can be used to supplement SemCor. We concluded that the sense distributions in LexSemTm are clearly superior to SemCor-based distributions for lemmas that are missing from SemCor.[1] This is highly significant, given that in total LexSemTm has approximately 88% coverage of polysemous WordNet lemmas, compared to only 39% for SemCor. Furthermore, we concluded that the LexSemTm sense distributions for lemmas in SemCor are roughly on-par with SemCor-based distributions. While our primary evaluation in Section 5.3, which resulted in these conclusions, was conducted only on lemmas with LexSemTm frequency at least 5,000 (lemmas where LexSemTm was trained on at least 5,000 usages), we presented multiple sources of evidence all indicating that these conclusions likely extend to all lemmas, regardless of LexSemTm frequency.

3. Can sense distribution learning also be applied to MWE lemmas, and if so how does this task compare to simplex sense distribution learning?

In Chapter 5 we also demonstrated that for all practical purposes, the task of MWE sense distribution learning seems to be equivalent to simplex sense distribution learning. In answering this question, we experimented with two simple methods of unsupervised general-purpose MWE identification, and found that the sense distributions obtained using either method — including both the gold-standard sense distributions after filtering out invalid usages, and the LexSemTm distributions — were almost indistinguishable. In addition, we showed that the LexSemTm sense distributions for MWE lemmas were comparable in quality to those for simplex lemmas, both in terms of their quality relative to benchmark and baseline distributions, and their absolute quality as measured by our metrics of sense distribution quality.

6.1.1 Research Outcomes and Impact

The most significant outcome of our work is the LexSemTm dataset. As noted above, this dataset contains English sense distributions with substantially greater coverage than SemCor over WordNet lemmas. This dataset can be of impact anywhere where accurate sense distributions are useful, especially in performing WSD, as discussed in Section 1.1.

In addition to this, LexSemTm contains WSI outputs — specifically the document distributions over topics and the topic distributions over words that were

[1]Recall that in this case, "SemCor-based" means the distribution is based on the first-listed WordNet sense, which is somewhat arbitrary in the absence of SemCor data (WordNet senses are usually listed in descending order by SemCor count).

produced by HCA— for both polysemous and nonpolysemous WordNet lemmas across all five languages. These can easily be aligned to any sense inventory from the respective languages, using the topic–sense alignment method of HCA-WSI, and any future improvements to this alignment method could be used to quickly update the WordNet sense distributions in LexSemTm without the need to run HCA-WSI again.

Furthermore, the presence of the non-English data in LexSemTm may also have significant impact, since SemCor-like resources for other languages are currently very limited. A summary of SemCor-like resources has recently been compiled by Petrolito and Bond (2014); while there exist a handful of such resources for other languages, they are all very limited, and usually substantially smaller than SemCor. However, the non-English data in LexSemTm has not yet been evaluated, so future work is required to be sure of its impact.

A second significant outcome of this thesis is our blueprint for running HCA-WSI efficiently on a language-wide scale, with some generic lemma-independent recommendations for hyperparameter settings and the quantity of data needed. This may have impact in guiding future work that involves large-scale sense distribution learning, for example learning domain-specific sense distributions over entire vocabularies to facilitate domain adaptation, or learning user-specific sense distributions for user modelling applications. The impact of our work in this respect is especially significant, given that our optimised method is over an order of magnitude faster than the previous state-of-the-art method, and also more robust. In addition to this, our methodology is language-independent and can be applied to simplex lemmas as well as MWEs.

Of particular note, to the best of knowledge, this is the first ever general-purpose sense distribution learning or WSD method that has been applied to MWEs. Therefore, an additional impact of our work with regard to this second outcome is that we have provided a solution to a previously unsolved problem, which may motivate future work on this novel task.

In addition to these major outcomes, in the process of evaluating our LexSemTm sense distributions — both for simplex and MWE lemmas — we created two gold-standard evaluation datasets: GoldSemCor and GoldMWE (in Section 5.3 and Section 5.4 respectively). These datasets each contain usages for a variety of lemmas labelled with WordNet senses, and may help facilitate future work on sense distribution learning. More specifically, GoldSemCor contains sense distributions for lemmas over a wide range of SemCor frequencies, which could facilitate future work on replacing or supplementing SemCor, and GoldMWE contains MWE sense distributions, which could facilitate future work on MWE sense distribution learning. In addition, the MWE usages in GoldMWE are labelled according to whether they are actual usages of the MWE or not, which means this dataset may also help facilitate future work on MWE identification.

Finally, our work may have impact due to some of its auxiliary findings. For in-

stance, in Section 4.3 and Section 5.4 we found strong evidence that HCA-WSI sense distribution learning is very robust. This was due to the negligible impact of hyperparameter settings, the negligible impact of the MWE usage identification methods, and the relatively small variance in sense distribution quality (compared to HDP-WSI) observed in our bootstrapping experiments. This is significant, because it indicates that researchers or practitioners using HCA-WSI probably do not need to spend any significant effort optimising hyperparameter or usage identification settings. Furthermore, due to the low variance in sense distribution quality they can be confident in the output of the method.

In addition, in Section 5.4 we found evidence that the sense distributions produced using HCA-WSI are systematically too flat. This finding could also be impactful, as it provides evidence that there is scope for improving the topic–sense alignment method used by HCA-WSI, which may motivate work to improve this method. This could in turn lead to improvements in HCA-WSI, and could be used directly to update the existing sense distributions in LexSemTm, as discussed above.

6.1.2 Research Limitations

One major limitation of our work is that we narrowed the scope of most of our evaluations to English nouns only. This was done in order to make the investigation manageable and the cost of acquiring labelled data reasonable, however it limits the confidence of our conclusions for other languages and other parts of speech (POS). On the other hand, as was argued in Section 1.2, we believe this is the part of the LexSemTm dataset that will be of greatest use to other researchers and practitioners.

A second shortcoming is that many of our evaluations were performed on relatively small sets of lemmas, which limited the statistical power of many of the comparisons in these investigations. For instance, the BNC dataset used extensively in Chapter 4 contained only 40 lemmas, the GoldSemCor gold-standard dataset from Section 5.3 contained only 50 lemmas, and the GoldMWE gold-standard dataset from Section 5.4 contained only 46 lemmas. Unfortunately, this was unavoidable due to the significant cost of obtaining sense-labelled data. Of course, this cost was one of the main motivators for developing an unsupervised method for large-scale sense distribution learning in the first place!

Finally, because there was no strong evaluation of LexSemTm sense distributions compared to SemCor-based distributions for lemmas with LexSemTm frequency less than 5,000, our conclusions regarding how LexSemTm compares to SemCor for these lemmas were somewhat vague. While we performed a secondary evaluation over a wider range of lemmas, and found that sense distribution quality only degraded slightly with low LexSemTm frequency, this evaluation was fairly rough; it was performed using SemCor-based distributions as proxy gold-standards. While these results were corroborated by findings from some of our other evaluations, we would be much more confident of our conclusions for the lemmas with low LexSemTm

frequency if we had evaluated over gold-standard sense-labelled data for such lemmas. Again, the main limitation stopping us from addressing this shortcoming was the cost of obtaining the required sense-labelled data.

6.2 Future Work

We conclude this thesis with some recommendations for future work. For each possible direction of future work, we provide some concrete suggestions for how our work could be extended in that direction.

6.2.1 Evaluating Remaining Data in LexSemTm

The most immediate recommendation would be to extend our evaluations in this thesis, by addressing the shortcomings described above. Specifically, this could involve obtaining sense-labelled data for: (1) English lemmas in LexSemTm covering a wide range of LexSemTm frequencies, including both lemmas present in and missing from SemCor; (2) English lemmas in LexSemTm over all POS (noun, verb, adverb, and adjective), including both lemmas present in and missing from SemCor; (3) English MWE lemmas over all POS; and (4) non-English lemmas. This sense-labelled data could be obtained using Amazon Mechanical Turk, as in Section 5.3 and Section 5.4. Alternatively, the non-English data could be obtained using existing multilingual sense-annotated corpora (for example, those listed in Petrolito and Bond (2014)). However, this would introduce similar limitations to our secondary evaluation in Section 5.3, where we used SemCor-based distributions as proxy gold-standards.

These sense-labelled datasets could be used directly to answer the following questions respectively: (1) how do LexSemTm sense distributions compare to SemCor-based distributions in quality, as a function of LexSemTm frequency? (2) does the relative quality of LexSemTm sense distributions compared to SemCor-based distributions depend on POS? (3) does the performance of MWE sense distribution learning depend on POS? and (4) how does the English contained in LexSemTm compare with the non-English, both in terms of sense distribution quality and qualitatively? These questions could be answered using similar methodology as in this thesis. This is a very appealing direction for work, given the large quantity of data in LexSemTm— specifically the non-English data, and the English data for POS other than noun — that we haven't yet evaluated.

6.2.2 Improving Topic–sense Alignment

Another direction for future work would be to improve the current topic–sense alignment method used by HDP-WSI and HCA-WSI. The current method is very naive,

as it only takes into account exact matches in word occurrences between sense glosses and topics. In particular, our observation from Section 5.4 that the LexSemTm sense distributions are systematically too flat strongly implies that there is scope to improve this method. One issue with the existing method is that it doesn't take into account words with very similar meanings. For example, if a gloss of the `river bank` sense of *bank* contained the word *river*, and a topic contained the words *water* and *stream*, we would like these words to contribute to the prevalence score for that sense. Furthermore, it is likely that some glosses will have a higher concentration of strongly context-bearing words than others, and thus will tend to obtain relatively higher prevalence scores, regardless of the "true" sense distribution.

We believe that this alignment method could be improved by taking advantage of distributed representations (embeddings) of words, senses, and topics.[2] A simple baseline approach for doing this would be to use existing word embeddings (for example word2vec vectors (Mikolov *et al.* 2013)), in order to create an embedding of each gloss and topic, by averaging the word vectors in each. Similarity between glosses and topics could then be computed based on these vectors (for example using cosine similarity), which could be used in place of JSD in performing alignment. Because similar words should have similar word embeddings, this may deal with the problem of similar but non-identical words appearing in topics and glosses, and the problem of some glosses containing more words with exact matches in topics than others.

A more advanced approach to using topic and sense embeddings for topic–sense alignment would be to take advantage of existing methods for calculating embeddings of senses or bags of words.[3] For example, Rothe and Schütze (2015) and Bhingardive *et al.* (2015) have recently proposed methods for calculating sense embeddings in the same vector space as word embeddings, for sense inventories with a WordNet-like network structure. In particular, the AutoExtend method of Rothe and Schütze (2015) looks very promising for this purpose, since AutoExtend sense vectors have been shown to be of high quality.[4] In addition, methods for calculating embeddings of bag of words, such as doc2vec[5] (Le and Mikolov 2014), could be used to create higher quality topic vectors, and also to retrain word embeddings for the language based on patterns in the topics (which could lead to further improvements in sense vectors). These more sophisticated sense and topic vectors could then be used to perform alignment, as described above for the baseline approach.

[2]Recall that an embedding of an object is a representation of that object as a vector in some vector space.

[3]A bag of words is a multiset of words. Topics from HDP or HCA can be represented as bags of words, based on the counts of how many times each word was allocated to a given topic in the topic modelling output (in other words, using the unnormalised versions of the topic distributions over words).

[4]This was measured by how successfully the sense vectors could be used to improve supervised WSD, by including features based on these vectors.

[5]Technically, only the distributed bag of words (DBOW) version of doc2vec can be applied to bags of words.

Furthermore, the observation from Section 5.4 that HCA-WSI sense distributions are systematically too flat should also be taken into consideration in any future work to improve topic–sense alignment. In the existing alignment method, a single topic is assigned to each usage, which are then aligned to to the glosses. We propose that it may be necessary to fundamentally change this architecture to deal with this flatness problem. As a first step in exploring new architectures, we suggest trying to assign a single sense to each usage rather than a single topic. The reason for this is that if we could ensure that most usages were assigned to the most frequent senses, we would be guaranteed of obtaining a relatively skewed sense distribution.

Given an improved topic–sense alignment method, it would also be worthwhile re-aligning the WSI data in LexSemTm to WordNet, and then repeating our experiments comparing LexSemTm sense distributions to SemCor-based distributions. It is possible that improvements to alignment could lead to stronger conclusions regarding whether LexSemTm can supplement or replace SemCor.

6.2.3 Extracting Novel Senses from LexSemTm

A completely different direction for future work would be to use the WSI data in LexSemTm to expand existing sense inventories. For example, Lau *et al.* (2012) and Lau *et al.* (2014) demonstrated that the WSI output of HDP-WSI could be used to detect novel word senses (that is, senses not currently listed in a given sense inventory). These methods could be applied to the WSI data in LexSemTm, in order to extract novel senses for all five languages. Furthermore, following methodology such as that of Cook *et al.* (2013), this could be used in order to expand existing lexical resources (such as WordNet or other dictionaries) with new senses.

This study could be particularly interesting, because LexSemTm contains WSI output for the nonpolysemous lemmas in each language, in addition to the polysemous lemmas. In the case of English, in addition to the approximately 27,000 polysemous lemmas contained in LexSemTm, there are approximately 70,000 nonpolysemous lemmas. Therefore, extracting novel senses from LexSemTm could lead to the discovery of many new polysemous lemmas.

6.2.4 Multiword Expression Sense Learning

Finally, given that MWE sense learning, including WSD, WSI, and sense distribution learning, are novel tasks to the best of our knowledge,[6] this thesis could motivate further work on MWE sense learning. One such direction would be to extend supervised WSD methods, such as IMS (Zhong and Ng 2010), to MWEs. To start with, supervised WSD could be performed on MWEs using the same features as for simplex lemmas, in order to compare the performance of supervised WSD for MWE versus

[6]Other than the minor exceptions noted in Section 2.4.

simplex lemmas. This could then be extended by experimenting with MWE-specific features, for example the insertion of any extra words between the MWE components.

A further extension to the above would be to also address the problem of MWE identification; it seems reasonable to believe that these tasks should be performed jointly, as they are related. Therefore, one could attempt to perform WSD and identification for MWEs jointly, in order to investigate whether either task can benefit the other. This could be done using standard supervised learning, or alternatively using semi-supervised learning with co-training (Blum and Mitchell 1998). Furthermore, this could be facilitated by our GoldMWE dataset, which contains sentences for a set of MWEs, along with gold-standard labels for both WSD and identification.

Bibliography

Agirre, Eneko, Timothy Baldwin, and David Martinez. 2008. Improving parsing and PP attachment performance with sense information. In *Proceedings of the 46th Annual Meeting of the ACL: HLT (ACL 2008)*, 317–325, Columbus, Ohio, USA. Association for Computational Linguistics.

——, Kepa Bengoetxea, Koldo Gojenola, and Joakim Nivre. 2011. Improving dependency parsing with semantic classes. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 699–703. Association for Computational Linguistics.

——, and Philip Edmonds. 2007. *Word Sense Disambiguation: Algorithms and Applications*. Dordrecht, Netherlands: Springer.

——, and David Martinez. 2004. Unsupervised WSD based on automatically retrieved examples: The importance of bias. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, 25–32, Barcelona, Spain. Association for Computational Linguistics.

Aldous, David J. 1985. *École d'Été de Probabilités de Saint-Flour XIII — 1983*, chapter Exchangeability and related topics, 1–198. Berlin: Springer.

Asahara, Masayuki, and Yuji Matsumoto, 2003. *IPADIC version 2.7 Users Manual*. Computational Linguistics Laboratory, Graduate School of Information Science, Nara Institute of Science and Technology.

Baldwin, Timothy, and Su Nam Kim. 2010. Multiword expressions. In *Handbook of Natural Language Processing*, ed. by Nitin Indurkhya and Fred J. Damerau. Boca Raton, USA: CRC Press, 2nd edition.

Banerjee, Satanjeev, and Ted Pedersen. 2003. Extended gloss overlaps as a measure of semantic relatedness. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, volume 3, 805–810, Acapulco, Mexico.

Bernard, J.R.L. (ed.) 1986. *The Macquarie Thesaurus*. Sydney, Australia: Macquarie Library.

Bhingardive, Sudha, Dhirendra Singh, V. Rudramurthy, Hanumant Redkar, and Pushpak Bhattacharyya. 2015. Unsupervised most frequent sense detection using word embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1238–1243, Denver, Colorado, USA.

Biran, Or, Samuel Brody, and Noemie Elhadad. 2011. Putting it simply: a context-aware approach to lexical simplification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 496–501.

Bird, Steven, Edward Loper, and Ewan Klein. 2009. *Natural Language Processing with Python*. O'Reilly Media, Inc., 1st edition.

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

Blum, Avrim, and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, 92–100, Madison, Wisconsin, USA. ACM.

Bond, Francis, and Kyonghee Paik. 2012. A survey of wordnets and their licenses. In *Proceedings of the 6th Global WordNet Conference (GWC 2012)*, 64–71, Matsue, Japan. Global WordNet Association.

Boyd-Graber, Jordan, and David Blei. 2007. PUTOP: Turning predominant senses into a topic model for word sense disambiguation. In *Proceedings of the 4th International Workshop on Semantic Evaluation (SemEval-2007)*, 277–281, Prague, Czech Republic. Association for Computational Linguistics.

——, ——, and Xiaojin Zhu. 2007. A topic model for word sense disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 1024–1033, Prague, Czech Republic. Association for Computational Linguistics.

Brody, Samuel, and Mirella Lapata. 2009. Bayesian word sense induction. In *Proceedings of the 12th Conference of the EACL (EACL 2009)*, 103–111, Athens, Greece. Association for Computational Linguistics.

——, Roberto Navigli, and Mirella Lapata. 2006. Ensemble methods for unsupervised WSD. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, 97–104, Sydney, Australia. Association for Computational Linguistics.

Brown, Peter, John Cocke, S Della Pietra, V Della Pietra, Frederick Jelinek, Robert Mercer, and Paul Roossin. 1988. A statistical approach to language translation. In *Proceedings of the 12th conference on Computational linguistics - Volume 1*, 71–76, Budapest, Hungary. Association for Computational Linguistics.

Buntine, Wray L, and Swapnil Mishra. 2014. Experiments with non-parametric topic models. In *Proceedings of the 20th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2014)*, 881–890, New York City, New York, USA.

Burnard, Lou. 1995. Users reference guide british national corpus version 1.0. Technical report, Oxford University Computing Services, UK.

Callison-Burch, Chris, and Mark Dredze. 2010. Creating speech and language data with Amazon's Mechanical Turk. In *Proceedings of the North American Chapter of the Association for Computational Linguistics Human Language Technologies 2009 (NAACL 2009): Workshop on Creating Speech and Text Language Data With Amazon's Mechanical Turk*, 1–12, Los Angeles, USA. Association for Computational Linguistics.

Chan, Yee Seng, and Hwee Tou Ng. 2005. Word sense disambiguation with distribution estimation. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, 1010–1015, Edinburgh, Scotland, UK.

——, and ——. 2006. Estimating class priors in domain adaptation for word sense disambiguation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, 89–96, Sydney, Australia. Association for Computational Linguistics.

Chang, Baobao, Wenzhe Pei, and Miaohong Chen. 2014. Inducing word sense with automatically learned hidden concepts. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, 355–364, Dublin, Ireland.

Chang, Pi-Chuan, Michel Galley, and Christopher D. Manning. 2008. Optimizing Chinese word segmentation for machine translation performance. In *ACL Third Workshop on Statistical Machine Translation*, 224–232, Columbus, Ohio, USA. Association for Computational Linguistics.

Chen, Changyou, Lan Du, and Wray Buntine. 2011. Sampling table configurations for the hierarchical poisson-dirichlet process. In *Machine Learning and Knowledge Discovery in Databases*, volume 6912, 296–311. Springer.

Chen, Xinxiong, Zhiyuan Liu, and Maosong Sun. 2014. A unified model for word sense representation and disambiguation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1025–1035, Doha, Qatar. Association for Computational Linguistics.

Choe, Do Kook, and Eugene Charniak. 2013. Naive Bayes word sense induction. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, 1433–1437, Seattle, Washington, USA. Association for Computational Linguistics.

Cook, Paul, Jey Han Lau, Michael Rundell, Diana McCarthy, and Timothy Baldwin. 2013. A lexicographic appraisal of an automatic approach for detecting new word senses. In *Proceedings of eLex 2013*, 49–65, Tallinn, Estonia.

Dagan, Ido, Alon Itai, and Ulrike Schwall. 1991. Two languages are more informative than one. In *Proceedings of the 29th annual meeting on Association for Computational Linguistics*, 130–137, Berkeley, California, USA. Association for Computational Linguistics.

Doyle, Gabriel, and Charles Elkan. 2009. Accounting for burstiness in topic models. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML 2009)*, 281–288, Montreal, Canada.

Faralli, Stefano, and Roberto Navigli. 2012. A new minimally-supervised framework for domain word sense disambiguation. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2012)*, 1411–1422, Jeju Island, Korea. Association for Computational Linguistics.

Fazly, Afsaneh, Paul Cook, and Suzanne Stevenson. 2009. Unsupervised type and token identification of idiomatic expressions. *Computational Linguistics* 35.61–103.

Fellbaum, Christiane. 1998. *WordNet: An Electronic Lexical Database*. Cambridge, USA: MIT Press.

Ferguson, Thomas S. 1973. A bayesian analysis of some nonparametric problems. *The annals of statistics* 209–230.

Fothergill, Richard, and Timothy Baldwin. 2012. Combining resources for MWE-token classification. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics (*SEM 2012)*, 100–104, Montreal, Canada. Association for Computational Linguistics.

Galley, Michel, and Kathleen McKeown. 2003. Improving word sense disambiguation in lexical chaining. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, volume 3, 1486–1488, Acapulco, Mexico.

Goldwater, Sharon, Thomas L Griffiths, and Mark Johnson. 2011. Producing power-law distributions and damping word frequencies with two-stage language models. *The Journal of Machine Learning Research* 12.2335–2382.

Gonzalo, Julio, Felisa Verdejo, Irina Chugur, and Juan Cigarran. 1998. Indexing with wordnet synsets can improve text retrieval. In *Proceedings of the COLING-ACL Workshop on Usage of WordNet in Natural Language Processing Systems*, 38–44, Montreal, Canada.

Goyal, Kartik, and Eduard H Hovy. 2014. Unsupervised word sense induction using distributional statistics. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*, 1302–1310, Dublin, Ireland. Association for Computational Linguistics.

Hashimoto, Chikara, and Daisuke Kawahara. 2008. Construction of an idiom corpus and its application to idiom identification based on WSD incorporating idiom-specific features. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP 2008)*, 992–1001, Honolulu, Hawaii, USA. Association for Computational Linguistics.

Hovy, Dirk, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard Hovy. 2013. Learning whom to trust with MACE. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1120–1130, Atlanta, Georgia, USA.

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, New York City, New York, USA. Association for Computational Linguistics.

Isahara, Hitoshi, Francis Bond, Kiyotaka Uchimoto, Masao Utiyama, and Kyoko Kanzaki. 2008. Development of the japanese wordnet. In *Proceedings of 6th Language Resources and Evaluation Conference (LREC 2008)*, 2420–2423, Marrakech, Morocco.

Jiang, Jay J, and David W Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of the 10th Research on Computational Linguistics International Conference (ROCLING 1997)*, 19–33, Taipei, Taiwan.

Jin, Peng, Diana McCarthy, Rob Koeling, and John Carroll. 2009. Estimating and exploiting the entropy of sense distributions. In *Proceedings of the North American Chapter of the Association for Computational Linguistics Human Language Technologies (NAACL HLT 2009): Short Papers*, 233–236, Boulder, Colorado, USA. Association for Computational Linguistics.

Kim, Su Nam, and Timothy Baldwin. 2010. How to pick out token instances of English verb-particle constructions. *Language Resources and Evaluation* 44.97–113.

Koeling, Rob, Diana McCarthy, and John Carroll. 2005. Domain-specific sense distributions and predominant sense acquisition. In *Proceedings of the 2005 Conference on Empirical Methods in Natural Language Processing (EMNLP 2005)*, 419–426, Vancouver, Canada. Association for Computational Linguistics.

Krovetz, Robert, and W Bruce Croft. 1992. Lexical ambiguity and information retrieval. *ACM Transactions on Information Systems (TOIS)* 10.115–141.

Kucera, Henry, and Winthrop Nelson Francis. 1967. *Computational Analysis of Present-Day American English*. Brown University Press, 1st edition.

Kudo, Taku, Kaoru Yamamoto, and Yuji Matsumoto. 2004. Applying conditional random fields to japanese morphological analysis. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, 230–237, Barcelona, Spain. Association for Computational Linguistics.

Lapata, Mirella, and Chris Brew. 2004. Verb class disambiguation using informative priors. *Computational Linguistics* 30.45–73.

——, and Frank Keller. 2007. An information retrieval approach to sense ranking. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, 348–355, Rochester, New York, USA. Association for Computational Linguistics.

——, and Alex Lascarides. 2003. Detecting novel compounds: The role of distributional evidence. In *Proceedings of the 11th Conference of the European Chapter for the Association of Computational Linguistics (EACL-2003)*, 235–242, Budapest, Hungary. Association for Computational Linguistics.

Larochelle, Hugo, and Stanislas Lauly. 2012. A neural autoregressive topic model. In *Advances in Neural Information Processing Systems (NIPS 2012)*, 2708–2716, Lake Tahoe, Nevada, USA.

Lau, Jey Han, Paul Cook, Diana McCarthy, Spandana Gella, and Timothy Baldwin. 2014. Learning word sense distributions, detecting unattested senses and identifying novel senses using topic models. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, 259–270, Baltimore, Maryland, USA. Association for Computational Linguistics.

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

Le, Quoc V, and Tomas Mikolov. 2014. Distributed representations of sentences and documents. *arXiv preprint arXiv:1405.4053* .

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, Toronto, Canada. ACM.

Levin, Beth. 1993. *English verb classes and alternations: A preliminary investigation*. Chicago, USA: University of Chicago press.

Lin, Dekang. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 17th international conference on Computational linguistics (COLING 1998)*, volume 2, 768–774, Montreal, Canada. Association for Computational Linguistics.

Loukachevitch, Natalia, and Ilia Chetviorkin. 2015. Determining the most frequent senses using Russian linguistic ontology ruthes. In *Proceedings of the workshop on Semantic resources and semantic annotation for Natural Language Processing and the Digital Humanities at NODALIDA*, 21–27, Vilnius, Lithuania. The Northern EuropeanAssociation for Language Technology.

——, and Boris Dobrov. 2014. Ruthes linguistic ontology vs. russian wordnets. In *Proceedings of the Seventh Global Wordnet Conference*, ed. by Heili Orav, Christiane Fellbaum, and Piek Vossen, 154–162, Tartu, Estonia. Global WordNet Association.

Marcus, Mitchell P, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of english: The penn treebank. *Computational linguistics* 19.313–330.

McCarthy, Diana, Bill Keller, and John Carroll. 2003. Detecting a continuum of compositionality in phrasal verbs. In *Proceedings of the ACL2003*

Workshop on Multiword Expressions: analysis, acquisition and treatment, 73–80, Sapporo, Japan. Association for Computational Linguistics.

——, Rob Koeling, Julie Weeds, and John Carroll. 2004a. Finding predominant word senses in untagged text. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)*, 280–287, Barcelona, Spain. Association for Computational Linguistics.

——, ——, ——, and ——. 2004b. Using automatically acquired predominant senses for word sense disambiguation. In *Proceedings of the ACL SENSEVAL-3 workshop*, 151–154, Barcelona, Spain. Association for Computational Linguistics.

——, ——, ——, and ——. 2007. Unsupervised acquisition of predominant word senses. *Computational Linguistics* 33.553–590.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* .

Miller, George A, Claudia Leacock, and Randee Tengi. 1993. A semantic concordance. In *Proceedings of the ARPA Workshop on Human Language Technology*, 303–308, Plainsboro, New Jersey, USA.

Minnen, Guido, John Carroll, and Darren Pearce. 2001. Applied morphological processing of English. *Natural Language Engineering* 7.207–223.

Mohamed Noor, Nurril Hirfana, Suerya Sapuan, and Francis Bond. 2011. Creating the open Wordnet Bahasa. In *Proceedings of the 25th Pacific Asia Conference on Language, Information and Computation (PACLIC 25)*, 258–267, Singapore.

Mohammad, Saif, and Graeme Hirst. 2006. Determining word sense dominance using a thesaurus. In *Proceedings of the 11th Conference of the EACL (EACL 2006)*, 121–128, Trento, Italy. Association for Computational Linguistics.

Moro, Andrea, Alessandro Raganato, and Roberto Navigli. 2014. Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics* 2.231–244.

Navigli, Roberto. 2009. Word sense disambiguation: A survey. *ACM Computing Surveys* 41.1–69.

——. 2012. A quick tour of word sense disambiguation, induction and related approaches. In *SOFSEM 2012: Theory and practice of computer science*, 115–129. Springer.

——, and Giuseppe Crisafulli. 2010. Inducing word senses to improve web search result clustering. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNL 2010)*, 116–126, Cambridge, Massachusetts, USA. Association for Computational Linguistics.

——, and Paola Velardi. 2005. Structural semantic interconnections: a knowledge-based approach to word sense disambiguation. *Pattern Analysis and Machine Intelligence* 27.1075–1086.

Neelakantan, Arvind, Jeevan Shankar, Alexandre Passos, and Andrew McCallum. 2014. Efficient non-parametric estimation of multiple embeddings per word in vector space. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1059–1069, Doha, Qatar. Association for Computational Linguistics.

Ng, Hwee Tou, Bin Wang, and Yee Seng Chan. 2003. Exploiting parallel texts for word sense disambiguation: An empirical study. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics (ACL 2003)*, 455–462, Sapporo, Japan. Association for Computational Linguistics.

Padr, Llus, Miquel Collado, Samuel Reese, Marina Lloberes, and Irene Castelln. 2010. Freeling 2.1: Five years of open-source language processing tools. In *Proceedings of 7th Language Resources and Evaluation Conference (LREC 2010)*, 931–936, La Valletta, Malta.

Palmer, Martha, Hoa Trang Dang, and Christiane Fellbaum. 2007. Making fine-grained and coarse-grained sense distinctions, both manually and automatically. *Natural Language Engineering* 13.137–163.

Petrolito, Tommaso, and Francis Bond. 2014. A survey of wordnet annotated corpora. In *Proceedings of the 7th Global WordNet Conference (GWC 2014)*, 236–245, Tartu, Estonia. Global WordNet Association.

Pianta, Emanuele, Luisa Bentivogli, and Christian Girardi. 2002. Multiwordnet: developing an aligned multilingual database. In *Proceedings of the First Global WordNet Conference*, 293–302, Mysore, India. Global WordNet Association.

Piantadosi, Steven T. 2014. Zipfs word frequency law in natural language: A critical review and future directions. *Psychonomic Bulletin & Review* 21.1112–1130.

Pisceldo, Femphy, Ruli Manurung, and Mirna Adriani. 2009. Probabilistic part-of-speech tagging for bahasa indonesia. In *Third International MALINDO Workshop, Colocated Event ACL-IJCNLP*, Singapore.

Pitman, Jim, and Marc Yor. 1997. The two-parameter poisson-dirichlet distribution derived from a stable subordinator. *The Annals of Probability* 855–900.

Rose, Tony, Mark Stevenson, and Miles Whitehead. 2002. The reuters corpus volume 1-from yesterday's news to tomorrow's language resources. In *Proceedings of 3rd Language Resources and Evaluation Conference (LREC 2002)*, volume 2, 827–832, Las Palmas, Spain.

Rothe, Sascha, and Hinrich Schütze. 2015. Autoextend: Extending word embeddings to embeddings for synsets and lexemes. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 1793–1803, Beijing, China. Association for Computational Linguistics.

Salakhutdinov, Ruslan R, and Geoffrey E Hinton. 2009. Replicated softmax: an undirected topic model. In *Advances in Neural Information Processing Systems (NIPS 2009)*, 1607–1614, Vancouver, Canada.

Schneider, Nathan, Emily Danchik, Chris Dyer, and Noah A. Smith. 2014. Discriminative lexical semantic segmentation with gaps: running the MWE gamut. *Transactions of the Association for Computational Linguistics* 2.193–206.

——, Dirk Hovy, Anders Johannsen, and Marine Carpuat. 2016. Semeval-2016task 10: Detecting minimalsemantic units and their meanings (dimsum). In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, San Diego, California, USA. Association for Computational Linguistics.

Snow, Rion, Brendan O'Connor, Daniel Jurafsky, and Andrew Y Ng. 2008. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP 2008)*, 254–263, Honolulu, Hawaii, USA. Association for Computational Linguistics.

Tan, Liling, and Francis Bond. 2014. Ntu-mc toolkit: Annotating a linguistically diverse corpus. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: System Demonstrations*, 86–89, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Teh, Y. W., and M. I. Jordan. 2010. Hierarchical Bayesian nonparametric models with applications. In *Bayesian Nonparametrics: Principles and Practice*, ed. by N. Hjort, C. Holmes, P. Müller, and S. Walker. Cambridge University Press.

Teh, Yee Whye, Michael I Jordan, Matthew J Beal, and David M Blei. 2006. Hierarchical dirichlet processes. *Journal of the American Statistical Association* 101.1566–1581.

Tseng, Huihsin, Daniel Jurafsky, and Christopher Manning. 2005. Morphological features help POS tagging of unknown words across language varieties. In *The Fourth SIGHAN Workshop on Chinese Language Processing, 2005*, 32–39, Jeju Island, Korea.

Véronis, Jean. 2004. Hyperlex: lexical cartography for information retrieval. *Computer Speech & Language* 18.223–252.

Wang, Shan, and Francis Bond. 2013. Building the chinese open wordnet (cow): Starting from core synsets. In *Sixth International Joint Conference on Natural Language Processing*, 10–18, Nagoya, Japan.

Xie, Pengtao, Diyi Yang, and Eric Xing. 2015. Incorporating word correlation knowledge into topic modeling. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 725–734, Denver, Colorado, USA.

Xue, Naiwen, Fei Xia, Fu-Dong Chiou, and Marta Palmer. 2005. The penn chinese treebank: Phrase structure annotation of a large corpus. *Natural language engineering* 11.207–238.

Yao, Xuchen, and Benjamin Van Durme. 2011. Nonparametric bayesian word sense induction. In *Proceedings of TextGraphs-6: Graph-based Methods for Natural Language Processing*, 10–14, Portland, Oregon, USA. Association for Computational Linguistics.

Yosef, Mohamed Amir, Johannes Hoffart, Ilaria Bordino, Marc Spaniol, and Gerhard Weikum. 2011. Aida: An online tool for accurate disambiguation of named entities in text and tables. *Proceedings of the VLDB Endowment* 4.1450–1453.

Zhong, Zhi, and Hwee Tou Ng. 2010. It makes sense: A wide-coverage word sense disambiguation system for free text. In *Proceedings of the ACL 2010 System Demonstrations*, 78–83, Uppsala, Sweden. Association for Computational Linguistics.

——, and ——. 2012. Word sense disambiguation improves information retrieval. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL 2012)*, 273–282, Jeju Island, Korea. Association for Computational Linguistics.

Zipf, George Kingsley. 1936. *The psychobiology of language*. London: Routledge.

——. 1949. *Human behavior and the principle of least effort*. New York: Addison-Wesley.

Appendix A

Amazon Mechanical Turk Details

In this appendix we show in detail the Amazon Mechanical Turk (AMT: see Section 3.1.5) interfaces provided to workers for our annotation tasks. In addition, we list the control sentences used for each lemma in our AMT experiments. These details are of importance for the reproducibility of our annotation experiments.

A.1 AMT Interface

The interface provided to workers for creating our GoldSemCor gold-standard dataset (see Section 5.3) is shown in Figure A.1 and Figure A.2. At the start of each batch the instructions shown in Figure A.1 were shown to workers, in order to define the annotation task. Subsequently, the examples in Figure A.2 were provided, to help make the instructions more clear. After the instructions and examples they were provided with the 27 sentences to annotate, which were presented in the same style and with the same information as the examples.

Similarly, Figure A.3 and Figure A.4 show the instructions and examples respectively provided to workers at the start of each batch in the annotation task for creating GoldMWE (see Section 5.4). Again, these examples show the style of information and annotation options provided to workers for the subsequent 28 sentences they were asked to annotate.

Note that in each batch, the order of the annotation options was randomised once in order to reduce bias, but kept consistent throughout the batch in order to make the task as simple as possible for workers.

Annotating Word Usage with corresponding sense definition

Please be aware we do some quality control checks on submissions. HITs may be rejected if there is strong evidence that the input is spam; for example, if the provided answers are obviously all random.

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 contains a meaning definition in black, which will sometimes by accompanied by one or more example uses (in blue). In addition, a list of synonyms of the meaning, and categories that the meaning belongs to (things it is a type of), are also listed in blue. These may or may not make the definition more clear.

Your task is to choose, for each sentence, the most appropriate definition, which best reflects the meaning of the boldfaced word in the sentence.

WARNING: Please note that sentences included in this task were randomly selected from the Web, or other publicly available data, and may occasionally include content or language that you find offensive. They do not represent our views, policies or opinions and we accept no responsibility for them.

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 feel that multiple definitions are appropriate, select the definition that you believe is **most** appropriate. Similarly, if you don't think any of the definitions are correct, select the definition that you think is the closest to being relevant.

Figure A.1: Detailed instructions provided to AMT workers for the annotation of GoldSemCor. These are the general instructions provided at the start of each batch to be annotated.

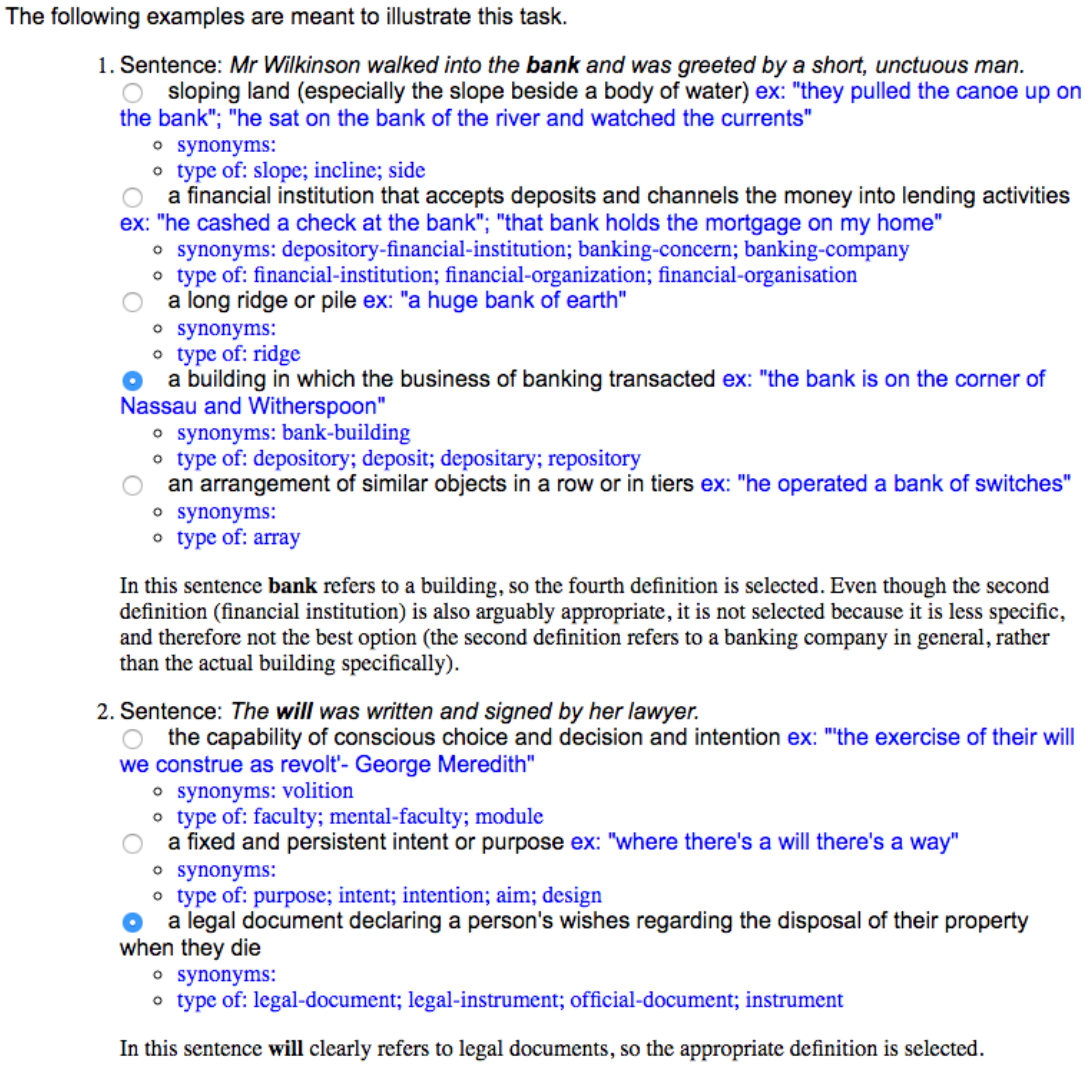

Figure A.2: Examples provided to AMT workers for the annotation of GoldSemCor. These were provided at the start of each batch after the detailed instructions in order to help define the annotation task by example.

Instructions:

In this experiment, you will be presented with a series of sentences. In each sentence, a given phrase 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 phrase. Each description contains a meaning definition in black, which will sometimes by accompanied by one or more example uses (in blue). In addition, a list of synonyms of the meaning, and categories that the meaning belongs to (things it is a type of), are also listed in blue. These may or may not make the definition more clear.

Your task is to choose, for each sentence, the most appropriate definition, which best reflects the meaning of the phrase made up by the boldfaced words in the sentence. If you believe that the sentence is not a valid usage of the phrase, select the "not a valid usage" option. On the other hand, if you believe that the sentence is a valid usage of the phrase but has a meaning that is clearly different than all those listed, select the "usage has a different meaning" option.

WARNING: Please note that sentences included in this task were randomly selected from the Web, or other publicly available data, and may occasionally include content or language that you find offensive. They do not represent our views, policies or opinions and we accept no responsibility for them.

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").

Please also be aware that a "valid" usage of a phrase may contain some gaps between the boldfaced words. For example, "**put** the lamp **down**" is a valid usage of the phrase "put down", however "stay **put** while I head **down** there" is not.

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 feel that multiple definitions are appropriate, select the definition that you believe is **most** appropriate. If you think the sentence is a valid usage of the phrase but you can't figure out the meaning, select the one you think is most probable. Only select "usage has a different meaning" if it has a clear meaning, which is obviously different than all of the supplied options.

Figure A.3: Detailed instructions provided to AMT workers for the annotation of GoldMWE. These are the general instructions provided at the start of each batch to be annotated.

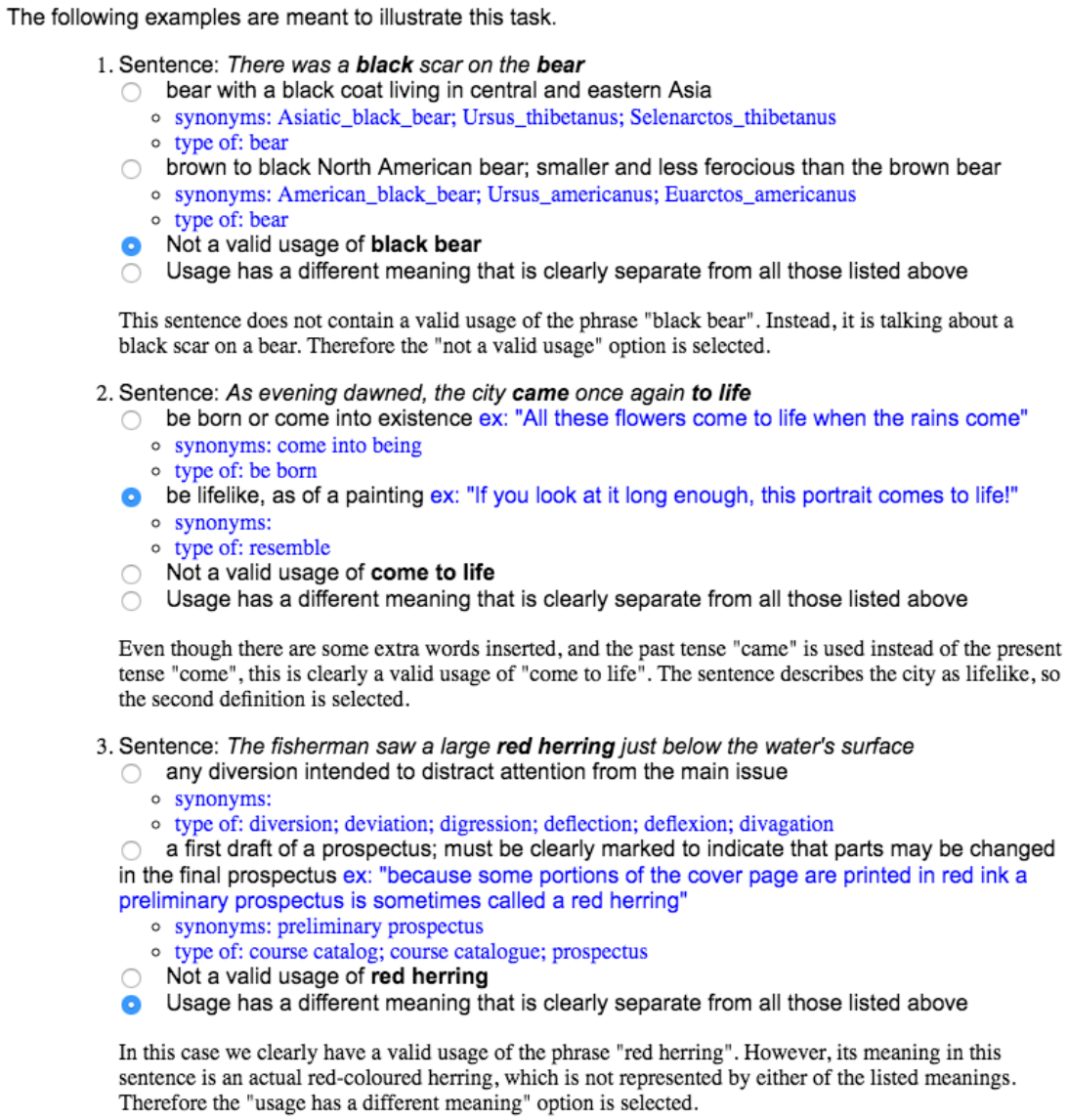

Figure A.4: Examples provided to AMT workers for the annotation of GoldMWE. These were provided at the start of each batch after the detailed instructions in order to help define the annotation task by example.

A.2 AMT Control Sentence Lists

We now list the control sentences used in our AMT annotation experiments. In Table A.1 we list the control sentences used for our simplex lemmas to create GoldSemCor, and in Table A.2 we list the control sentences used for our MWE lemmas to create GoldMWE. In both cases, for each lemma we list the set the lemma belongs to, the control sentences, and the "correct" sense(s) for each control sentence.

In the case of GoldSemCor lemmas, the set is based on the SemCor frequency of the lemma (see Section 5.3). On the other hand, in the case of GoldMWE lemmas the lemmas are partitioned into L_{int} and L_{diff}, based on whether they are in the high precision part of GoldMWE (see Section 5.4).

The numeric senses listed are a reference to the sense order in WordNet; for example, "#2"' refers to the second-listed sense in WordNet for a given lemma. Alternatively, in the case of MWE lemmas in GoldMWE, the "invalid" sense means that the sentence is a negative example; that is, the sentence is considered not a valid usage of the given MWE. For both tables of control sentences, if multiple senses are listed it means that we decided — sometimes based on annotator response, and sometimes ourselves — that the control sentence was ambiguous, and multiple senses could apply. These ambiguous control sentences were excluded when training MACE, as detailed in Section 5.3 and Section 5.4.

Lemma	Set	Control Sentence	Sense(s)
animation	$L^{(1)}_{gsc}$	She spoke with **animation** about her trip.	#3, #4, #6
		The Web site has hundreds of **animations** you can download.	#5
bowler	$L^{(1)}_{gsc}$	The man was dressed in a sharp suit and a **bowler** hat.	#3
		He is currently the best performing **bowler** on the Australian cricket team.	#1
crossover	$L^{(1)}_{gsc}$	A rock musician's **crossovers** into jazz and soul music.	#3
		What happens if they **crossover** and vote in the Democratic primary?	#2
fin	$L^{(1)}_{gsc}$	The large muscles of the body actually do most of the work, but the **fins** help with balance and turning.	#6
		We consistently talk to snorkelers who have wide feet; men specifically who have problems finding **fins** that will fit them.	#4
flora	$L^{(1)}_{gsc}$	One of Australia's greatest treasures is her **flora** — a staggering 24,000 species of native plants have been identified compared to England's 1700 native plants.	#1
		Mr Steven's garden contained a wide variety of native and introduced **flora**.	#2
format	$L^{(1)}_{gsc}$	The data **format** is widely used, as it facilitates very efficient processing under most scenarios.	#1
		The magazine is especially known for its easy to read **format**, and weekly satire columns.	#2
lodge	$L^{(1)}_{gsc}$	But otherwise, traditional Native American houses like these are usually only built for ritual or ceremonial purposes, such as a sweat **lodge** or tribal meeting hall.	#5
		After a hard day of walking, the pilgrims decided to search the local village for a **lodge** to stay for the night.	#6
metabolism	$L^{(1)}_{gsc}$	There are four stages in the **metabolism** of butterflies and moths: egg, larva, pupa, and adult.	#1
		If we eat and drink more kilojoules than we need for our **metabolism** and exercise, we store it mostly as fat.	#2
propulsion	$L^{(1)}_{gsc}$	Sailboats use wind as their source of **propulsion**.	#1
		In an act of **propulsion**, Andy hurled the ball over the wall.	#2
punt	$L^{(1)}_{gsc}$	This little 300 **punt** is light and economical, get into the boating lifestyle on a budget.	#2
		The Aussie **punt** is when we drop the nose and kick the point of the football.	#3
blend	$L^{(2)}_{gsc}$	The name Microsoft is a **blend** of the words microcomputer and software.	#2
		A refreshing **blend** of ice, coffee and milk that can be sweetened or flavored.	#1, #3
conjecture	$L^{(2)}_{gsc}$	This hypothesis was dismissed on the grounds that is was only **conjecture**.	#1
		Recent work has involved examining the cognitive processes involved with **conjecture**.	#3
cream	$L^{(2)}_{gsc}$	Coffee can be prepared with or without **cream**.	#2

		Recently an experimental anti-ageing skin **cream** has been trailed by pharmaceutical companies.	#3
designer	$L^{(2)}_{gsc}$	Interior **designers** are often employed to decorate their homes.	#1
		The following graphic **designers** were involved in creating the show's artwork:	#3
heroine	$L^{(2)}_{gsc}$	The book's main **heroine** was widely regarded as a one-dimensional character.	#1
		As a result of her brave actions she was declared a **heroine** by the President of the United States.	#2
hobby	$L^{(2)}_{gsc}$	The king's **hobby** had injured its wing during the last hunt.	#3
		Hunting is a common **hobby** for those living in rural communities.	#1
jewel	$L^{(2)}_{gsc}$	The crown was set with a variety of brightly colored **jewels**.	#1
		To her father, she was a real **jewel**.	#2
lease	$L^{(2)}_{gsc}$	The **lease** must be signed by the lawyers of both parties before the contract is enforceable.	#2
		The **lease** is 5 years.	#3
poster	$L^{(2)}_{gsc}$	The activist **posters** regularly placed pieces of propaganda on walls around the university.	#2
		Missing-pet **posters** are a common sight on poles and trees around the city.	#1
zombie	$L^{(2)}_{gsc}$	The **zombie** is a high-octane cocktail full of delicious fruit juices, that make you feel like you're downing sophisticated candy.	#5
		The soldiers were believed to have been brought back to life to create an army of **zombies**.	#1
anatomy	$L^{(3)}_{gsc}$	He was a researcher in human **anatomy**.	#1
		The **anatomy** of a movie trailer.	#3
associate	$L^{(3)}_{gsc}$	He worked as an **associate** at the bank for three years before being promoted.	#3
		He was a close **associate** of the mayor.	#1, #2
graduate	$L^{(3)}_{gsc}$	Too many law **graduates** and not enough jobs.	#1
		The meniscus is used to measure the volume of a liquid in a container, such as a **graduate**.	#2
lane	$L^{(3)}_{gsc}$	The cafe is located on a small **lane** in between the town hall and the public library.	#1
		Swimming pools are often divided into separate **lanes** for swimmers of different speeds.	#2
orange	$L^{(3)}_{gsc}$	Nutrients contained in **oranges** are plentiful and diverse.	#1, #3
		Orange is the color of the Dutch Royal Family.	#2
original	$L^{(3)}_{gsc}$	A Van Gogh **original** nearly sold for $80.	#1
		Several schematic copies were created from the **original**.	#2
symphony	$L^{(3)}_{gsc}$	Beethoven's ninth **symphony** is arguably the single piece that inspired the methodology of musical analysis.	#1
		The university **symphony** is playing in the concert hall after noon.	#2

treason	$L^{(3)}_{gsc}$	He was sentenced to death later that year for high **treason**.	#1
		He considered his best friend's betrayal as an act of **treason**.	#2, #3
uncle	$L^{(3)}_{gsc}$	He had over a dozen aunts and **uncles**.	#1
		He served as an **uncle** to the new employees at the company, helping them settle in to work.	#2
variable	$L^{(3)}_{gsc}$	The main **variable** is the teacher.	#1
		The equation contains three **variables** in it; x, y, and t.	#4
base	$L^{(4)}_{gsc}$	They returned to the main military **base** in Paris.	#1, #14
		They set up camp at the **base** of Mt Everest.	#4
belt	$L^{(4)}_{gsc}$	The lawn formed a green **belt** around the manor.	#5
		The man's **belt** did not match his black dress shoes.	#2
canvas	$L^{(4)}_{gsc}$	The tent was made from a thick, green **canvas**.	#1, #4
		The signature wrestling move involves slamming the opponent face-down onto the **canvas**.	#6
engagement	$L^{(4)}_{gsc}$	It is normal for couples to hold an **engagement** party before getting married.	#3
		In the first **engagement** alone, both sides lost thousands of troops.	#1
occupation	$L^{(4)}_{gsc}$	The territory is currently under military **occupation** by Russia.	#2, #5
		His **occupation** with his smartphone was so great that he missed his train stop.	#3
pot	$L^{(4)}_{gsc}$	Hot summer days can leave **pot** plants looking worse for wear with wilted or curling leaves and dehydrated soil, here's how to revive them.	#4
		At the end of the game of poker, he won the entire **pot**.	#6
survey	$L^{(4)}_{gsc}$	A **survey** was conducted on Americans that found 1 in 4 think the sun goes around the Earth.	#1
		He tried to get a better **survey** of the munitions camp through his binoculars.	#3
technology	$L^{(4)}_{gsc}$	The company is developing new **technology** to automatically assemble buildings.	#1
		He served for 10 years as a professor in **technology**.	#2
tension	$L^{(4)}_{gsc}$	The book was filled with a **tension** between its desire to be exciting and action packed, and its political message.	#3
		There was an openly hostile **tension** in the air as the rival gang members faced each other.	#5
victim	$L^{(4)}_{gsc}$	Reports say there were 567 **victims** from the cyclone.	#1
		Many people are taken in by fraudsters — criminals that use clever tactics and tricks to try and manipulate their **victims**.	#2
integration	$L^{(5)}_{gsc}$	Successful **integration** of different racial and religious groups into the United States community is considered an essential component of immigration policy.	#1
		Mathematical **integration** is first taught to students in late high school, in calculus courses.	#3
number	$L^{(5)}_{gsc}$	Every whole **number** is either even or odd.	#2

		Afterwards, she nervously asked him for his telephone **number**.	#4
opinion	$L_{gsc}^{(5)}$	Public **opinion** holds that a traditional family is the ideal environment to raise children.	#3
		Her privately-held **opinion** was that the monarchy should be overthrown.	#1, #6
parent	$L_{gsc}^{(5)}$	While her **parents** were away on vacation, she was looked after by her uncle.	#1
		This rare tree was used as a **parent** to grow thousands of clone trees.	#2
payment	$L_{gsc}^{(5)}$	She received a **payment** of $100 for services rendered.	#1, #2
		As **payment** for his crimes, he was sentenced to death by guillotine.	#3
pleasure	$L_{gsc}^{(5)}$	He felt immense pleasure at seeing his children again.	#1, #2
		The directors serve at the **pleasure** of the shareholders.	#3
result	$L_{gsc}^{(5)}$	Lightning occurs as a **result** of an accumulation of electrostatic charge in the air.	#1
		The **results** of the blood test are sent directly to the specialist physician.	#3
speech	$L_{gsc}^{(5)}$	He delivered the first **speech** at the opening ceremony.	#1
		There is ongoing research into whether other great apes are capable of **speech**.	#2, #4, #8
step	$L_{gsc}^{(5)}$	The world's longest staircase is located on the face of Mt. Niesen in Switzerland, and contains 11,674 **steps**.	#4
		He took ten **steps** towards the door before collapsing.	#3
theory	$L_{gsc}^{(5)}$	The **theory** of gravity has allowed scientists to understand the motions of stars and planets.	#1
		The police officer acted on the **theory** that at least one of them must have been telling the truth.	#3

Table A.1: List of all control sentences used in creation of the GoldSemCor gold-standard dataset, from Section 5.3. For each lemma we list which partition of L_{gsc} the lemma belongs to, along with both control sentences and the sense(s) for each control sentence. Each sense listed is based on the order of the lemma's senses in WordNet.

Lemma	Set	Control Sentence	Sense(s)
batting average	L_{diff}	Online Calculator for calculating the **batting average** of a baseball player.	#1
		The stock trader increased her **batting average** to 9 in 10 successful trades.	#2
		His skill in **batting** was **average** at best.	invalid
black and white	L_{diff}	Back then, all photographs were **black and white**.	#2
		This kind of written communication was sometimes referred to as **black and white**.	#1
		Many studies have established an earnings gap between **black** workers **and white** workers.	invalid
cabinet minister	L_{diff}	He was a **cabinet minister** in the 23rd government.	#1
		As a senior minister and member of the cabinet, her job title was **cabinet minister**.	#2
		She was hurriedly searching for a clean dress from her **cabinet** while the **minister** waited outside.	invalid
carrying out	L_{diff}	The award was awarded for the most artful **carrying out** of the play.	#2
		The **carrying out** of the commander's orders was their first priority.	#1
		Most of the mess was created while **carrying** the uprooted tree **out** of the property.	invalid
case study	L_{diff}	After the bank collapsed, a detailed **case study** was undertaken to find the cause.	#1
		Most **case studies** done on the drug have found more reliable results for younger patients.	#2
		It is recommended that students study high school maths in **case** they decide to **study** advanced science or engineering.	invalid
community service	L_{diff}	She had regularly volunteered to perform **community service** at the local aged care home since a young age.	#1
		As part of his punishment, he was sentenced to 6 months of **community service**.	#2
		The local **community** centre offers various **services** to residents of the suburb.	invalid
dance music	L_{diff}	Back then, **dance music** referred to music used for ballroom dancing.	#1
		Electronic **dance music** was booming, and no company symbolized the boom more than SFX.	#2
		The **dance** instructor muted the **music** until the protestors left.	invalid
day school	L_{diff}	A **day school** - as opposed to a boarding school - is an institution where children (or high-school age adolescents) are given educational instruction during the day, after which the students return to their homes.	#3

		Unlike night schools, **day schools** do their teaching during the day.	#2
		Students are angry at the school's plans to scrap the four **day school** week.	invalid
end of the world	L_{diff}	Various groups of Christians believe that the Last Judgement - i.e. the **end of the world** - is going to occur in the next century.	#1
		Various ancient tribes prophesied a possible **end of the world**.	#2
		At the **end of the** promenade is a **world** class restaurant.	invalid
fairy tale	L_{diff}	Reading **fairy tales** to children at a young age has been shown to improve educational outcomes.	#1
		He called in sick to work, but was fired because his boss thought his story was a **fairy tale**.	#2
		The **fairy** told a **tale** about her home kingdom.	invalid
first base	L_{diff}	**First base** is often considered an offensive position to play.	#2
		The government only reached **first base** on their plans to restructure the tax department.	#3
		He had **first** arrived at the **base** two years before.	invalid
first lady	L_{diff}	She was the **first lady** of Russian dance.	#1
		The president's wife is known as the **first lady**.	#2
		Food is to be served **first** to the **lady**, then to the gentleman.	invalid
gold rush	L_{diff}	The recent explosion in computing speed has created a **gold rush** within the IT sector.	#1
		Alluvial gold was discovered along the banks of the Bendigo Creek in 1851 and resulted in a major **gold rush**.	#2
		They stormed the bank to collect their **gold** in a mad **rush**, fearful that it would all be gone.	invalid
golden age	L_{diff}	The period of the Roman Empire is often questionably viewed as a kind of **golden age** for Europe.	#2
		The **golden age** of advertising - the 60s.	#1
		The longest living **golden** eagle lived to **age** 46.	invalid
golf club	L_{diff}	He was a member of the same **golf club** for over 20 years.	#1
		She had purchased a new set of **golf clubs** in anticipation of the big game on Saturday.	#2
		After a long game of **golf** they went out **clubbing**.	invalid
grammar school	L_{diff}	She spent the early years of her education, up to the age of 12, at the local **grammar school**.	#2
		The majority of students at the university program completed their education at an Anglican **grammar school**, most of whom had studied at least one of Latin or Classics.	#1
		It is believed by some experts that education in **grammar** at primary **school** is currently lacking.	invalid

music hall	L_{diff}	The show was held every Saturday at the **music hall** in Chicago.	#1
		The group performed a **music hall**, which was a popular style of performance at the time.	#2
		There was **music** playing in the **hall**.	invalid
open door	L_{diff}	Voters were opposed to the **open door** policy on international trade.	#1
		As part of an initiative to increase transparency, the department was forced to maintain an **open door** to journalists.	#2
		The abandoned house had **open** windows and no **doors**.	invalid
real time	L_{diff}	Internet technologies allow the odds for sporting events to be updated in **real time**.	#1
		Subsequent advances in search engine engineering allowed a massive reduction in the **real time** for search queries to be processed.	#2
		I believe I made it **real** clear that this **time** failure will not be tolerated.	invalid
record album	L_{diff}	The band's first **record album** was an international sensation.	#1
		Back then, they used to store their phonograph records in a **record album**.	#2
		Later that year, they broke the **record** for best selling **album** of all time.	invalid
street name	L_{diff}	The **street name** was changed from Croydon Rd to Norfolk Rd.	#4
		Some of the **street names**, slang terms and nicknames given to cocaine during the height of its popularity have become part of the American lexicon.	#2
		The man is known to live on the same **street**, but his **name** is unknown.	invalid
track record	L_{diff}	Only companies with a strong **track record** are hired for tier 1 construction projects.	#1
		Usain Bolt bounced back from a rare defeat, setting a **track record** at the Bislett Games on Thursday in his first 200-meter race of the season.	#2
		He was on **track** to release another **record** by the year 2000.	invalid
training school	L_{diff}	Many students attend the **training school** after high school, in order to learn skills such as carpentry.	#1
		When youth require detention, the program ensures that those youth exit the **training school** with the education, skills and supports to reduce the likelihood of recidivism.	#2
		Every Friday he went to football **training** after **school**.	invalid

turning point	L_{diff}	the agreement was a **turning point** in the history of both nations.	#1
		Turn left at the next **turning point**, in order to enter Lennox St.	#2
		There was an excellent article on John Ford, **turning** on the **point** that anyone who admired Ford's later works must have only a very imperfect appreciation of the earlier ones.	invalid
box office	L_{int}	It was the most successful movie that year in terms of **box office** results.	#1
		Tickets can be purchased at the **box office** on the level above the cinema.	#2
		A new delivery of cardboard **boxes** arrived at the **office** yesterday.	invalid
common law	L_{int}	In the context of civil law systems, **common law** is often used to refer to laws established by precedent.	#1
		England follows a **common law** system, which means that legal cases are primarily decided based on the outcomes of past cases, rather than statutory law.	#2
		It is **common** to ignore the **law** in these kinds of cases.	invalid
concentration camp	L_{int}	After the war, the enemy soldiers were held in **concentration camps** for over 6 years.	#1
		Their living situation, which involved nine people sleeping in two bedrooms, could be easily categorized as a **concentration camp**.	#2
		The **concentration** of boot **camp** facilities was increased again that year.	invalid
field goal	L_{int}	He didn't score any **field goals** until the last game of the football season.	#1
		It is commonly believed that taller basketball players have a distinct advantage in scoring **field goals**.	#2
		After her recent success in track and **field** events, her **goal** was to join the number one division.	invalid
first class	L_{int}	The airline only recently started offering **first class**.	#3
		First class is a fast, affordable way to send envelopes and lightweight packages.	#2
		He graduated **first** in his **class** at Harvard.	invalid
free agent	L_{int}	Since he ripped up his contract with the Manchester United Football Club, he was a **free agent**.	#1
		After joining the hippie commune she started living as a **free agent**.	#2
		They are **free** to hire an **agent** to act on their behalf.	invalid
health care	L_{int}	The government's **health care** policy provides social insurance for the sick and needy in society.	#1

		The doctor provided top quality **health care** to his patients.	#2
		Although the car's engine was not in good **health**, nobody **cared**.	invalid
heavy metal	L_{int}	Mercury is classified as a **heavy metal**.	#1
		He started his career as a member of a local **heavy metal** band.	#2
		The wooden table is less **heavy** than the **metal** one.	invalid
home run	L_{int}	In his first professional baseball game, he scored two **home runs**.	#1
		The movie's box office success was a **home run**.	#2
		In order to increase her fitness, every day she left **home** and **ran** around the lake.	invalid
military service	L_{int}	In feudal, medieval England, **military service** was often the basis of land tenure.	#2
		The available manpower of the country's **military service** is over 100 million individuals.	#1
		The **military** mostly obtained administrative **services** from other departments.	invalid
number one	L_{int}	His selfish life philosophy was based around taking care of **number one**.	#1
		He is often considered the **number one** best chess player of all time.	#2
		She also lost a **number** of limited edition **one** dollar coins in the flood.	invalid
old man	L_{int}	He was believed to be an **old man**, possibly over 80 years old.	#1
		Common wormwood, or **old man**, is a species of flowering plants in the sunflower family.	#4
		The **old** dog and young **man** were out for a walk.	invalid
point of view	L_{int}	History is often taught from the **point of view** of the conquerers.	#1
		From her new **point of view**, the tree looked much taller than it did from further away.	#2
		Far above the **point of** impact the **view** of the crater is breathtaking.	invalid
post office	L_{int}	He had worked for several years as a high ranking administrator of the US **Post Office**.	#2
		The pub is located 50m from the local **post office**.	#1
		The **post** is visible from the **office** window.	invalid
public school	L_{int}	Conservatives are unhappy about the amount of taxpayer money going towards government run **public schools**.	#1
		Unlike in other countries, in Great Britain **public school** refers to a kind of private, independently run secondary school.	#2
		He did not like to be seen in **public** in his **school** uniform.	invalid

public service	L_{int}	Her research in medical science was awarded as an extraordinary act of **public service**.	#1
		Over half of the workforce is employed in the **public service**.	#2
		It was first floated on the stock market as a **public** financial **services** company.	invalid
railway line	L_{int}	In this context, the **railway line** refers to a bundle of railway routes bundled together as a corporation (e.g. a UK train operating company).	#1
		The old **railway lines** in the desert have been showing signs of rust, and are in need of repair.	#2
		She was at the **railway** station standing in **line** to board the train.	invalid
roller coaster	L_{int}	The constant change of pace in the movie provided an emotional **roller coaster** for viewers.	#1
		As of the summer 2009, Walt Disney World has five **roller coaster** rides.	#2
		Skateboards, Roller Skates, **Roller** Blades, **Coasters** and Go-Carts.	invalid
task force	L_{int}	The military created a special **task force** to recapture the capital from the rebels.	#1
		The program was launched by a school **task force** created to counter playground bullying.	#2
		The first **task** was to **force** the opposition to agree to the new tax plan.	invalid
theme song	L_{int}	The radio station changed their **theme song** after continuous listener complaints.	#1
		He was hired to compose the **theme song** for every movie in the series.	#2
		For each chosen **theme**, three matching **songs** were decided.	invalid
third party	L_{int}	It is recommended that these services are performed by a **third party**, so there is no conflict of interests between landlord and tenant.	#1
		The Greens have recently become a major **third party** in Australian politics.	#2
		She was the **third** guest at the **party** to arrive.	invalid
young man	L_{int}	Most of the workers at the factory are **young men**.	#1
		She was out for dinner that night with her **young man**.	#2
		The plot is based around a **young** woman and a **man** of middle age.	invalid

Table A.2: List of all control sentences used in creation of the GoldMWE gold-standard dataset, from Section 5.4. For each MWE lemma we list which partition of L_{union} the lemma belongs to, along with all three control sentences and the sense(s) for each control sentence. Each sense listed is based on the order of the lemma's senses in WordNet, and "invalid" is listed if the sentence is a negative example (not a valid MWE usage).