First Steps in Using Word Senses as Contextual Features in Maxent Models for Machine Translation

Steven Neale, Luís Gomes and António Branco
Department of Informatics
Faculty of Sciences
University of Lisbon, Portugal
{steven.neale, luis.gomes, antonio.branco}@di.fc.ul.pt

Abstract

Despite the common assumption that word sense disambiguation (WSD) should help to improve lexical choice and improve the quality of the output of machine translation systems, how to successfully integrate word senses into such systems remains an unanswered question. While significant improvements have been reported using reformulated approaches to the disambiguation task itself – most notably in predicting translations of full phrases as opposed to the senses of single words – little improvement or encouragement has been gleaned from the incorporation of traditional WSD into machine translation.

In this paper, we present preliminary results that suggest that incorporating output from WSD as contextual features in a maxent-based translation model yields a slight improvement in the quality of machine translation and is potentially a step in the right direction, in contrast to other approaches to introducing word senses into a machine translation system which significantly impede its performance.

1 Introduction

Ambiguity is a common problem in language, caused by the phenomena of identical words having multiple, distinct meanings (Xiong and Zhang, 2014). To use a classic example, the word ‘bank’ could be interpreted in the sense of the financial institution or as the slope of land at the side of a river, depending on the context in which it is used. In natural language processing (NLP), word sense disambiguation (WSD) refers to the process of solving this problem by determining the ‘sense’ or meaning of a word when used in a particular context (Agirre and Edmonds, 2006).

In computational terms, WSD is a classification task, where the context in which a target word is used provides evidence that helps to determine which class of words – sense – it should be assigned to (Agirre and Edmonds, 2006). Most approaches to WSD in recent years have been ‘knowledge-based’, with those classes of words stored in lexical ontologies such as WordNet (Fellbaum, 1998), where the collective meanings of open-class words (nouns, verbs, adjectives and adverbs) are grouped together as ‘synsets’. For tasks such as machine translation, ambiguous terms are a major potential source of errors, as identical words with different meanings will normally have different target translations (Xiong and Zhang, 2014). Thus, it has long been assumed that in order for a machine translation system to be optimally successful, it must incorporate some kind of WSD component (Carpuat and Wu, 2005).

Most attempts to integrate WSD components into machine translation systems have met with mixed – and usually limited – success. Early attempts at ‘projecting’ word senses directly into a machine translation system (Carpuat and Wu, 2005) were followed by a complete reformulation of the disambiguation process as a multi-word ‘phrase sense’ disambiguation approach, yeilding some improvements in translation quality (Carpuat and Wu, 2007). More recently, a ‘word sense induction’ approach that assigns word senses without the need for predefined sense inventories (such as WordNets) has been explored (Xiong and Zhang, 2014), but the question of whether pure word senses from traditional, knowledge-based WSD approaches can be useful for machine translation still remains.

Proceedings of the 1st Deep Machine Translation Workshop (DMTW 2015), pages 64–72,
Praha, Czech Republic, 3–4 September 2015.

In this paper, we demonstrate that by including the output from WSD as a feature in a maximum entropy (maxent)-based translation model, small gains in machine translation from English to Portuguese can be obtained. The contribution of our work, albeit preliminary in nature, is in showing these gains, however small, to be possible without having to reformulate WSD or drastically alter the way disambiguation is performed – the features added to the transfer model are direct outputs of a state-of-the-art WSD algorithm, without any kind of intermediary conversion or reformulation of either the word senses or the algorithm that delivers them.

We first explore previous efforts to integrate word senses into machine translation (Section 2), before describing our own approaches to the problem (Section 3). Next, we present our evaluation of these approaches, comparing different methods of integrating the output from a WSD process into a machine translation system (Section 4). Finally, we discuss our findings (Section 5) before making our conclusions (Section 6).

2 Related Work

Early work from Carpaut and Wu (2005) presented empirical results that cast doubt on the common assumption that the disambiguation of word senses could help to improve the quality of machine translation systems. They demonstrated that many of the contextual features important to WSD algorithms are implicit in the language models that are trained to perform machine translation, making them WSD models in their own right (Carpuat and Wu, 2005). Despite acknowledging that dedicated WSD algorithms are usually based on rich semantic data and that this should enable better predictions of lexical choice to be made, they showed a machine translation system trained on complete parallel sentences (rather than isolated target words as in WSD) to yield higher BLEU scores than a system where WSD output was forced into the translation model (Carpuat and Wu, 2005).

Based on these outcomes, a reformulated disambiguation process was proposed, with multi-word phrases the target as opposed to single words (Carpuat and Wu, 2007). Leveraging the fact that machine translation models are trained using contextual features from full sentences already, this 'phrase sense disambiguation' approach was designed to "generalize WSD to multi-word targets" and to incorporate the "crucial assumptions" that underly the sentence-based translation models into the sense disambiguation process as well (Carpuat and Wu, 2007). Across a number of evaluation metrics for machine translation, the phrase sense disambiguation approach was found to yield improved transation quality, suggesting that the sentence-based translation models used by machine translation systems can benefit from the addition of phrase-based (rather than word-based) sense disambiguation (Carpuat and Wu, 2007).

Further attempts to reformulate WSD into a more phrase-based concept followed. Chanel et al (2007) described having successfully integrated WSD into a machine translation system to obtain significantly improved results, but actually create their 'senses' by extracting English translations from full phrases in Chinese and using them as proposed translations . Inspired by traditional approaches to WSD, Giménez and Màrquez (2007) also advocated the move from 'word translation' to 'phrase translation', describing how lists of possible translations of a single source phrase can help to predict the correct translations of complete phrases in a given target language.

Recently, a renewed interest in exploring whether traditional, single word-based WSD can be useful for machine translation has emerged. Xiong and Zhang (2014) use the related technique of 'word sense induction' (WSI) to investigate whether or not pure word senses can be integrated into machine translation in such a way as to yield improvements in translation quality, being successful in their approach to predicting the senses of target words (rather than predicting their translations, as with the phrase-based approaches to disambiguation) (Xiong and Zhang, 2014). However, WSI automatically induces senses of words by clustering them together using their neighbouring words as context, *without* the need for a predefined sense inventory as in traditional WSD (Xiong and Zhang, 2014). The question still remains – how can word senses disambiguated using the rich semantic ontologies (such as WordNet) on which traditional WSD is based be successfully integrated into machine translation systems?

3 Description

This section outlines our implementation of WSD as part of a machine translation process, including descriptions of the graph-based algorithm we use to perform the WSD, the machine translation system and framework into which we implement it, and the two approaches we have taken to making use of the information output by the WSD process: 1) forcing information into the input sentences (directly affecting the alignment of words *before* the translation model is trained), and 2) including information as features in a maxent-based translation model (which does *not* affect word alignment but rather directly influences the training of the translation model).

3.1 WSD algorithm - *UKB*

To perform WSD we use *UKB*, a collection of tools and algorithms for performing graph-based WSD over a pre-existing knowledge base (Agirre and Soroa, 2009; Agirre et al., 2014). Graph-based WSD, as pioneered by a number of researchers (Navigli and Velardi, 2005; Mihalcea, 2005; Sinha and Mihalcea, 2007; Navigli and Lapata, 2007; Agirre and Soroa, 2008), allows knowledge bases such as WordNets to be represented as weighted graphs, where word senses correspond to nodes and the relationships or dependencies between pairs of senses correspond to the edges between nodes. The strength of the edge between two nodes, corresponding to the relationship or dependency between two synsets, can then be calculated using semantic similarity measures such as the Lesk algorithm (Lesk, 1986).

UKB uses graph-based representations of knowledge bases to choose the most likely sense of a word in a given context, based on the dependencies between nodes in the graph (Agirre and Soroa, 2009). Nodes (senses) 'recommend' each other based on their own importance – with the importance of any given node being higher or lower depending on the importance of other nodes which recommend it – and then follow a 'random walk' over the rest of the graph based on the importance of the nodes to whose edges they are attached (Mihalcea, 2005; Agirre and Soroa, 2009). The final probability of a random walk from the target word's node ending on any other node in the graph determines the most appropriate (probable) sense of the target word.

We choose to use UKB in our work for two reasons:

- UKB includes tools for automatically creating graph-based representations of knowledge bases in WordNet-style formats.
- The algorithm used by UKB for performing WSD over the graph itself has been consistently shown to produce results in line with or above the state-of-the-art (Agirre and Soroa, 2009; Agirre et al., 2014).

For the purpose of our work, we are thus able to perform highly-efficient WSD over an accurate graph-based representation of our chosen knowledge base (WordNet), meaning that any differences in the results of our integration of disambiguated output into the machine translation system can be confidently attributed to the integration process, rather than to the quality of the WSD output itself.

3.2 Machine Translation system - *TectoMT*

The machine translation system used in our work is *TectoMT*, a multi-purpose open source NLP framework that allows different software modules and tools to be integrated with each other (Popel and Žabokrtský, 2010)[1]. The framework is based on individual modules (known as 'blocks') that allow new or existing tools to be created or 'wrapped' in such a way that they can be easily integrated at various stages in a larger pipeline. These blocks are re-usable in different contexts and combinations (known as 'scenarios') to perform a variety of NLP tasks and are designed to be language-independent where possible, reducing the amount of repeated, expensive and time-consuming extra work usually needed to integrate tools.

For machine translation, TectoMT breaks down the source language and reconstructs the target language according to four layers of representation: the word layer (raw text), the morphological layer, the

[1] The TectoMT framework is now being developed under the name *Treex*: `https://github.com/ufal/treex`

analytical layer (shallow-syntax) and the tectogrammatical layer (deep-syntax). Different combinations of blocks make up each of the three scenarios needed for machine translation – one for analysis (of the source language), one for transfer (of tectogrammatical nodes from source to target language) and one for synsthesis (of the target language).

3.3 Integrating WSD output into a TectoMT-based pipeline

The first step in integrating the output produced by the WSD process into the machine translation pipeline is to wrap the WSD process as a block that can be included in user-created scenarios using the TectoMT framework. This new block converts input sentences to a format suitable for the UKB algorithm, and then performs WSD on each sentence using a graph-based representation of our chosen knowledge base, WordNet. For each word disambiguated by UKB, the returned output consists of the 8-digit synset identifier of the appropriate sense in WordNet chosen at the end of the random walk over the graph.

The TectoMT WSD block then maps this output back onto the input sentence, either as the synset identifier returned by UKB, an 'unknown' tag ('UNK', given to UKB but not able to be disambiguated) or a 'not applicable' tag ('_', not open-class and not given to UKB). This mapped WSD output is encoded into the analytical layer of each word in TectoMT as an attribute of the given word. Once words in the analytical layer have been assigned word senses as attributes, there are two ways with which we have experimented making use of this information for training actual translation models:

3.3.1 Forcing synset identifiers into input sentences prior to creating translation models

Forcing the synset identifiers produced by the WSD process onto the input sentences prior to creating translation models is achieved by taking the synset identifier from the WSD attribute stored in the analytical layer for a given word and using it in place of the original lemma. During the training of transfer models, when alignments are made between sentences from parallel corpora in the source and target languages, it should be the case that the forced synset identifiers help to create more accurate alignments between pairs of words based on their meanings, rather than solely their lexical form[2]. In this paper, we investigate two possible ways to force a synset identifier onto the lemma:

- Replacing the lemma with the synset identifier (e.g. 'word' becomes '01234567')
- Appending the synset identifier to the lemma (e.g. 'word' becomes 'word_01234567')

If we consider a link between the English word 'table' and the Portuguese word 'mesa', we may find that this alignment is made when 'table' should have been interpreted as a table of results, not in the sense of the piece of furniture which would correspond to 'mesa'. Replacing the lemma 'table' with the synset identifier for table in the sense of the piece of furniture should ensure a more accurate alignment between the appropriate sense of the word table and the Portuguese word 'mesa'. Appending the synset identifier to the lemma is an extension of this technique which we hypothesized might avoid potential problems concerning lexical choice.

For example, it might be that in some situations two words such as 'table' and 'desk' in English might belong to the same synset, but correspond to different words ('mesa' and 'secretária' respectively) in Portuguese. By replacing the English words by the synset identifier and aligning that with the Portuguese words, we are essentially assigning the main lemma of the synset (e.g. 'table') to both Portuguese words, which while being better than assigning the wrong sense of table altogether, is not quite as accurate as aligning 'desk' to 'secretária'. Hence, by appending the synset identifiers to the original lemmas (e.g. aligning 'table_01234567' to 'mesa' and 'desk_01234567' to 'secretária'), we are hopefully able to constrain alignments to the correct sense of source language words without introducing problems relating to lexical choice.

[2] For the work described in this paper, we make no assumption about the number of synset identifiers found in the training corpus before using them to align words. This may be an interesting caveat to explore in future work.

3.3.2 Including synset identifiers as features of a maxent-based translation model

TectoMT leverages the alignments it finds between the words in pairs of sentences from a parallel corpus to create and train maxent-based translation models, which are used later to perform machine translation tasks. Maximum entropy (maxent) classifiers, which are used when the conditional independence of a set of 'features' cannot be assumed, are common in NLP, where features such as neighbouring words usually provide context and are therefore not independent. In TectoMT, for each word in the source language that has more than one possible translation in the target language a maxent model exists to determine the probability of any of those translations being correct based on contextual features such as neighbouring words – words with only one translation have no ambiguity, and hence no need of a maxent model. For statistical machine translation systems, previous research suggests that maxent-based translation models are an effective way of leveraging the context provided by the neighbouring words of source sentences (Ittycheriah and Roukos, 2007; Bangalore et al., 2007).

In order for maxent models to be created, analysis must have been performed on both the source and target languages, in order for the models to be trained based on aligned parallel treebanks of sentences represented as tectogrammatical (deep-syntax) trees. The maxent model for each word is trained using a list of 'samples', which are themselves vectors between contextual features in the source language'node' (the tectogrammatical representation of the given word) and an output label (e.g. the lemma of the given word). Contextual features might include information (such as lemmas) from neighbouring nodes in the tectogrammatical tree (such as parent or sibling nodes), which help to provide the context in which a particular word was used.

The maxent model learns, using this information, to output the correct label (target language lemma) given a particular vector of source language contextual features (e.g. a sentence that we want to translate). With the output from the WSD process already stored as an attribute of the analytical layer by the WSD block that we added to the TectoMT framework (and hence propagated to the tectogrammatical layer), synset identifiers can also be added as source language contextual features of words. Thus, the maxent model can in theory constrain the expected probability of a possible translation as determined by the neighbouring words in context to the particular sense in which a given word was used.

4 Evaluation

This section describes our evaluation of how the results of translation from English to Portuguese using our baseline TectoMT-based machine translation system are affected by our two approaches to including information from WSD in the process:

- Forcing synset identifiers into input sentences prior to creating translation models:
  - By replacing lemmas with synset identifiers
  - By appending synset identifiers to lemmas
- Adding synset identifers as features in a maxent-based translation model:
  - As features of single nodes (words)
  - As features of single nodes plus their parent nodes
  - As features of single nodes plus their sibling (to the left and right) nodes
  - As features of single nodes plus their parent *and* sibling nodes

4.1 Experimental System Setup

In order to run the evaluation, we introduce different combinations of interchangable blocks to the analysis scenario in TectoMT, in order that WSD is performed and that its output (synset identifiers) can be propagated from the analytical to the tectogrammatical layer, and thus included in the eventual translation model. As described in section 3.3.2, aligned parallel treebanks of sentences are needed in order for maxent models to be created for target words, and so analysis scenarios are set up for both the source language (English) and the target language (Portuguese). WSD, however, is only included on the source language side (English).

Method	BLEU
Baseline	21.67
Replacing Synsets	20.46
Appending Synsets	19.86
Synset as Feature	**21.69**

Table 1: A comparison of incorporating WSD into a machine translation system by 1) forcing synset identifiers into input sentences (replacing lemmas or appending synsets to lemmas) or 2) adding synsets identifiers to a maxent model as features

Feature Types (Synset of ...)	BLEU
None (Baseline)	21.67
Single Node	**21.69**
+ Parent	21.61
+ Siblings	**21.68**
+ Parent & Siblings	21.62

Table 2: A comparison of different types of features that can be added to a maxent model, including the synset identifiers of 1) single nodes, 2) single nodes plus parent nodes, 3) single nodes plus sibling nodes, and 4) single nodes plus parent *and* sibling nodes

For both approaches, the WSD block is used to run the graph-based UKB algorithm (desribed in section 3.1) over the source sentences in English. In order to use the algorithm, we create the required dictionary files and corresponding graph from version 3.0 of the Princeton English WordNet (Fellbaum, 1998), comprising approximately 117,000 synsets. The 8 digit identifiers of any of these synsets can be assigned by the algorithm to given words in an input text, based on the context provided by their surrounding open class words.

For the adding synset identifiers as features in a maxent-based translation model approach, the inclusion of the WSD block in the scenario is all that is needed – the synset identifiers it returns are included in the analytical layer of each word, and from there propagated to the tectogrammatical layer and, finally, the maxent model where they are called upon as features. For the forcing synset identifiers into input sentences approach, two additional (interchangable) blocks are included in the scenario: 1) a block for replacing a given lemma in the input sentence with the synset identifier returned by the WSD, and 2) a block for appending the synset identifier returned by the WSD to a given lemma in the input sentence.

4.2 Training Corpus

Transfer models are trained over a small, in-domain corpus. The corpus primarily consists of 2000 sentences of questions and answers from a chat-based technology help service (1000 questions and 1000 answers). These sentences are sourced from a real-world company who employ human technicians to provide technical assistance to their customers (technology users) through a chat interface. These 2000 sentences are supported by a number of aligned terms sourced from localized terminology data from Microsoft (13,000 terms) and LibreOffice (995 terms), making the total size of our in-domain corpus approximately 16,000 paired segments (of which 2000 are full sentences and approximately 14,000 are paired terms). No development set or tuning steps are needed in the TectoMT-based pipeline.

4.3 Results of Including WSD Output in Machine Translation

By interchanging the different blocks incorporated into the analysis scenario of TectoMT to train different translation models for evaluation, we can compare our two chosen approaches to including the output from WSD in a machine translation system: 1) forcing synset identifiers into the input sentences prior to creating translation models, and 2) adding synset identifers as features in a maxent-based translation model. For all evaluations, we analyse the different translation models using a test corpus of 1000 full answers to questions asked by people seeking assistance in resolving problems using technology, as per the domain of the training corpus described in section 4.2.

Table 1 shows that when translating these 1000 sentences from English to Portuguese using a baseline TectoMT system (without WSD), we achieve a BLEU score of 21.67. Using the first approach (forcing sysnet identifers into the input sentences prior to creating translation models), the scores we obtain are significantly lower than the baseline (at a 0.05 level of significance) – 20.46 when we replace lemmas with synset identifiers, and 19.86 when appending the sysnet identifier to the lemma. Using the second approach (adding synset identifiers as features in a maxent-based translation model) we obtain a BLEU score of 21.69, *very* slightly above the baseline.

Table 2 shows our experimentation with adding different types of features into the maxent-model for a given word: 1) synset identifiers from single nodes (the standard method, as used to obtain the score in Table 1), 2) synset identifiers from single nodes plus the parent node in the tectogrammatical tree, 3) synset identifiers from single nodes plus the sibling (left and right) nodes, and 4) synset identifiers from single nodes plus the parent *and* sibling nodes. With a baseline BLEU score of 21.67 and a slightly improved score of 21.69 when including the synset identifiers of single nodes, as before, the table demonstrates that adding the synset identifiers of sibling nodes yields a BLEU score of 21.68, slightly above the baseline but slightly below single nodes only, while adding parent nodes alone or parent *and* sibling nodes yields BLEU scores of 21.61 and 21.62 respectively, significantly and almost significantly lower than the baseline (at a 0.05 level of significance).

5 Discussion

In addition to showing that adding synset identifiers as features in a maxent-based translation model yields a BLEU score very slightly above our baseline TectoMT-based machine translation system – suggesting that with some further tweaking output from WSD *can* be useful for machine translation, without the need for any kind of intermediary reformulation or conversion – there are some interesting outcomes from our evaluation. Namely, we found it surprising that:

- Using the first approach (forcing synset identifiers into the input sentences prior to creating the tranlation models), appending synset identifiers to lemmas yielded *worse* results than replacing lemmas with synset identifers.

- Using the second approach (adding synset identifiers as features in a maxent-based translation model), adding the synset identifiers of the parent nodes as extra features in the maxent model *decreases* the BLEU score.

A possible explanation for the weaker results obtained in general using the first approach is that maxent models, as their description in section 3.3.2 demonstrates, already include lemmas from neighboring nodes as contextual features, in much the same way as graph-based WSD algorithms such as UKB rely on the open class words surrounding a given target word as context. The maxent model could be seen as repeating a very similar task, and while it may not be as wholly dedicated to it as a WSD-specific algorithm, we may find that the maxent models used in machine translation are "sufficiently accurate" so that the output from WSD is only able to improve on the lexical choice offered by the maxent model in a "relatively small proportion of cases" (Carpuat and Wu, 2005).

Taken in this context, and assuming as proposed by Carpaut and Wu (2005) that machine translation is excessively dependent on the language models it trains, it could be the case that forcing synset identifiers into the input sentences prior to creating translation models only introduces excessive data that cannot really be put to any efficient use. This might also explain how appending synset identifiers to lemmas yielded even lower results than replacing lemmas with synset identifiers – while the case made in section 3.3.1 for appending the synset identifiers in order to preserve lexical choice seems persuasive, it may in fact be that as well as introducing a redundant synset identifier that cannot be put to much use, this renders the lemma itself redundant (by way of being intrinsically tied to that identifier), thus increasing the sparsity of the input sentences.

The second surprising outcome of our evaluation was the discovery that while adding the synset identifiers of nodes as features in a maxent model yields a slight improvement over the baseline BLEU score, adding synset identifiers from parent nodes as well can have a significantly adverse effect on results (the inclusion of sibling nodes seems to 'limit the damage' to a very small degree). This seems counterintuitive – introducing the output of WSD as a feature in the maxent model seems to yield an improvement, as Xiong and Zhang (2014) also found when creating a sense-based translation model based on their reforumulated word sense induction approach, and one would expect that providing a maxent model with more features would introduce more useful constraints.

As a possible explanation for this outcome, we consider that not all of the open class words UKB tries to disambiguate will be assigned an appropriate synset identifier – a particular word may not have had an entry in WordNet to begin with, or in a very small number of cases the synset identifier assigned by the algorithm may not have been the correct one. For parent and sibling nodes – parent nodes in particular – this inevitably means that for a given node whose synset identifier *is* included in the maxent model, it might often be the case that its parent (and to a lesser extent sibling) nodes in the tectogrammatical tree do *not* have synset identifiers of their own – we are probably not adding many synset identifiers anyway by choosing to include the extra information from these nodes. We might also consider that if multiple additional synset identifiers are all very different from each other, they might act as conflicting rather than constraining information, thus increasing the overall redundancy or sparsity of the data included in the maxent model.

6 Conclusions

We have presented preliminary findings that suggest that it *is* possible to improve machine translation results by incorporating information about word senses, making direct use of the output of WSD tools and *without* the need for any kind of intermediary reformulation or conversion of either the WSD tool itself or its output. By including the output from WSD as features in a maxent-based translation model, we obtain slightly higher BLEU scores than with a baseline version of the system running without these added features (translating from English to Portuguese), indicating that these features can increase the likelihood of pairings between words and phrases occuring in the translation model.

While the improvement we report is not statistically significant, we find any improvement at all to be in contrast to other approaches we experimented with – replacing synset identifers with lemmas, appending synset identifiers to lemmas, and including the synset identifiers of the parent nodes of words as features in the maxent-based translation model – all of which produce results significantly below our baseline machine translation system. While these results seem counterintuitive – more information should provide more constraints on the probabilities of alignments and pairings between words being made – we interpret them as showing that the extra data we introduce to the translation model with these approaches has resulted in too much sparsity, rather than constraint. It would be interesting in future work to explore whether a paraphrasing (Marton et al., 2009) or synonym-based approach as opposed to a strictly word sense-based approach might yield different outcomes.

While the work we report in this paper is in a preliminary state, the small improvement achieved by adding synset identifiers as features of single nodes in a maxent-based translation model does represent a step in the right direction, and merits further discussion and experimentation. The results reported here are based on a very controlled evaluation, trained on a small, in-domain corpus. We acknowledge that training on large, open domain corpora such as Europarl might produce different results, and aim to investigate this in the future. In addition, we also plan to explore how different types of word sense information and different approaches to WSD itself, as well as alternative machine translation evaluation metrics (possibly more semantically-oriented), might affect the gains we report using the 'senses as features' approach we describe here.

Acknowledgements

This work has been undertaken and funded as part of the EU project QTLeap (EC/FP7/610516) and the Portuguese project DP4LT (PTDC/EEI-SII/1940/2012).

References

Eneko Agirre and Philip Edmonds. 2006. *Word Sense Disambiguation: Algorithms and Applications*. Springer.

Eneko Agirre and Aitor Soroa. 2008. Using the Multilingual Central Repository for Graph-Based Word Sense Disambiguation. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, may. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

Eneko Agirre and Aitor Soroa. 2009. Personalizing PageRank for Word Sense Disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '09, pages 33–41, Athens, Greece. Association for Computational Linguistics.

Eneko Agirre, Oier López de Lacalle, and Aitor Soroa. 2014. Random Walks for Knowledge-based Word Sense Disambiguation. *Comput. Linguist.*, 40(1):57–84, March.

Srinivas Bangalore, Patrick Haffner, and Stephan Kanthak. 2007. Statistical machine translation through global lexical selection and sentence reconstruction. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, ACL 2007.

Marine Carpuat and Dekai Wu. 2005. Word Sense Disambiguation vs. Statistical Machine Translation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (ACL-05)*, pages 387–394.

Marine Carpuat and Dekai Wu. 2007. How Phrase Sense Disambiguation Outperforms Word Sense Disambiguation for Statistical Machine Translation. In *Proceedings of the 11th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI 2007)*.

Yee Seng Chan, Hwee Tou Ng, and David Chiang. 2007. Word Sense Disambiguation Improves Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL-07)*, pages 33–40, Prague, Czech Republic.

Christine Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Jesús Giménez and Lluís Màrquez. 2007. Context-Aware Discriminative Phrase Selection for Statistical Machine Translation. In *Proceedings of the 2nd Workshop on Statistical Machine Translation*, pages 159–166, Prague, Czech Republic.

Abraham Ittycheriah and Salim Roukos. 2007. Direct translation model 2. In *Proceedings of NAACL Human Language Technology Conference 2008*, NAACL HLT '07, pages 57–64.

Michael Lesk. 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*, SIGDOC '86, pages 24–26, New York, NY, USA. ACM.

Yuval Marton, Chris Callison-Burch, and Philip Resnik. 2009. Improved statistical machine translation using monolingually-derived paraphrases. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 381–390, Singapore, August. Association for Computational Linguistics.

Rada Mihalcea. 2005. Unsupervised Large-vocabulary Word Sense Disambiguation with Graph-based Algorithms for Sequence Data Labeling. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 411–418, Stroudsburg, PA, USA. Association for Computational Linguistics.

Roberto Navigli and Mirella Lapata. 2007. Graph Connectivity Measures for Unsupervised Word Sense Disambiguation. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 1683–1688, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Roberto Navigli and Paola Velardi. 2005. Structural Semantic Interconnections: A Knowledge-Based Approach to Word Sense Disambiguation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(7):1075–1086, July.

Martin Popel and Zdeněk Žabokrtský. 2010. TectoMT: Modular NLP Framework. In *Proceedings of the 7th International Conference on NLP*, IceTal '10, pages 293–304. Springer Berlin Heidelberg.

Ravi Sinha and Rada Mihalcea. 2007. Unsupervised Graph-basedWord Sense Disambiguation Using Measures of Word Semantic Similarity. In *Proceedings of the International Conference on Semantic Computing*, ICSC '07, pages 363–369, Washington, DC, USA. IEEE Computer Society.

Deyi Xiong and Min Zhang. 2014. A Sense-Based Translation Model for Statistical Machine Translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL-2014)*, pages 1459–1469, Baltimore MD, USA.