# Supervised and Unsupervised Word Sense Disambiguation on Word Embedding Vectors of Unambiguous Synonyms

**Aleksander Wawer**
Institute of Computer Science
PAS
Jana Kazimierza 5
01-248 Warsaw, Poland
axw@ipipan.waw.pl

**Agnieszka Mykowiecka**
Institute of Computer Science
PAS
Jana Kazimierza 5
01-248 Warsaw, Poland
agn@ipipan.waw.pl

## Abstract

This paper compares two approaches to word sense disambiguation using word embeddings trained on unambiguous synonyms. The first one is an unsupervised method based on computing log probability from sequences of word embedding vectors, taking into account ambiguous word senses and guessing correct sense from context. The second method is supervised. We use a multilayer neural network model to learn a context-sensitive transformation that maps an input vector of ambiguous word into an output vector representing its sense. We evaluate both methods on corpora with manual annotations of word senses from the Polish wordnet.

## 1 Introduction

Ambiguity is one of the fundamental features of natural language, so every attempt to understand NL utterances has to include a disambiguation step. People usually do not even notice ambiguity because of the clarifying role of the context. A word *market* is ambiguous, and it is still such in the phrase *the fish market* while in a longer phrase like *the global fish market* it is unequivocal because of the word *global*, which cannot be used to describe physical place. Thus, distributional semantics methods seem to be a natural way to solve the word sense discrimination/disambiguation task (WSD). One of the first approaches to WSD was context-group sense discrimination (Schütze, 1998) in which sense representations were computed as groups of similar contexts. Since then, distributional semantic methods were utilized in very many ways in supervised, weekly supervised and unsupervised approaches.

Unsupervised WSD algorithms aim at resolving word ambiguity without the use of annotated corpora. There are two popular categories of knowledge-based algorithms. The first one originates from the Lesk (1986) algorithm, and exploit the number of common words in two sense definitions (glosses) to select the proper meaning in a context. Lesk algorithm relies on the set of dictionary entries and the information about the context in which the word occurs. In (Basile et al., 2014) the concept of overlap is replaced by similarity represented by a DSM model. The authors compute the overlap between the gloss of the meaning and the context as a similarity measure between their corresponding vector representations in a semantic space. A semantic space is a co-occurrences matrix M build by analysing the distribution of words in a large corpus, later reduced using Latent Semantic Analysis (Landauer and Dumais, 1997). The second group of algorithms comprises graph-based methods which use structure of semantic nets in which different types of word sense relations are represented and linked (e.g. WordNet, BabelNet). They used various graph-induced information, e.g. Page Rank algorithm (Mihalcea et al., 2004).

In this paper we present a method of word sense disambiguation, i.e. inferring an appropriate word sense from those listed in Polish wordnet, using word embeddings in both supervised and unsupervised approaches. The main tested idea is to calculate sense embeddings using unambiguous synonyms (elements of the same synsets) for a particular word sense. In section 2 we shortly present existing results for WSD for Polish as well as other works related to word embeddings for other languages, while section 3 presents annotated data we use for evaluation and supervised model training. Next sections describe the chosen method of calculating word sense embeddings, our unsuper-

*Proceedings of the 1st Workshop on Sense, Concept and Entity Representations and their Applications*, pages 120–125,
Valencia, Spain, April 4 2017. ©2017 Association for Computational Linguistics

vised and supervised WSD experiments and some comments on the results.

## 2 Existing Work

### 2.1 Polish WSD

There was very little research done in WSD for Polish. The first one from the few more visible attempts comprise a small supervised experiment with WSD in which machine learning techniques and a set of a priori defined features were used, (Kobyliński, 2012). Next, in (Kobyliński and Kopeć, 2012), extended Lesk knowledge-based approach and corpus-based similarity functions were used to improve previous results. These experiments were conducted on the corpora annotated with the specially designed set of senses. The first one contained general texts with 106 polysemous words manually annotated with 2.85 sense definitions per word on average. The second, smaller, WikiEcono corpus (http://zil.ipipan.waw.pl/plWikiEcono) was annotated by another set of senses for 52 polysemous words. It contains 3.62 sense definitions per word on average. The most recent work on WSD for Polish (Kędzia et al., 2015) utilizes graph-based approaches of (Mihalcea et al., 2004) and (Agirre et al., 2014). This method uses both plWordnet and SUMO ontology and was tested on KPWr data set (Broda et al., 2012) annotated with plWordnet senses — the same data set which we use in our experiments. The highest precision of 0.58 was achieved for nouns. The results obtained by different WSD approaches are very hard to compare because of different set of senses and test data used and big differences in results obtained by the same system on different data. (Tripodi and Pelillo, 2017) reports the results obtained by the best systems for English at the level of 0.51-0.85% depending on the approach (supervised or unsupervised) and the data set. The only system for Polish to which to some extend we can compare our approach is (Kędzia et al., 2015).

### 2.2 WSD and Word Embeddings

The problem of WSD has been approached from various perspectives in the context of word embeddings.

Popular approach is to generate multiple embeddings per word type, often using unsupervised automatic methods. For example, (Reisinger and Mooney, 2010; Huang et al., 2012) cluster contexts of each word to learn senses for each word, then re-label them with clustered sense for learning embeddings. (Neelakantan et al., 2014) introduce flexible number of senses: they extend sense cluster list when a new sense is encountered by a model.

(Iacobacci et al., 2015) use an existing WSD algorithm to automatically generate large sense-annotated corpora to train sense-level embeddings. (Taghipou and Ng, 2015) prepare POS-specific embeddings by applying a neural network with trainable embedding layer. They use those embeddings to extend feature space of a supervised WSD tool named IMS.

In (Bhingardive et al., 2015), the authors propose to exploit word embeddings in an unsupervised method for most frequent sense detection from the untagged corpora. Like in our work, the paper explores creation of sense embeddings with the use of WordNet. As the authors put it, sense embeddings are obtained by taking the average of word embeddings of each word in the sense-bag. The sense-bag for each sense of a word is obtained by extracting the context words from the WordNet such as synset members (S), content words in the gloss (G), content words in the example sentence (E), synset members of the hypernymy-hyponymy synsets (HS), and so on.

## 3 Word-Sense Annotated Treebank

The main obstacle in elaborating WSD method for Polish is lack of semantically annotated resources which can be applied for training and evaluation. In our experiment we used an existing one which use wordnet senses – semantic annotation (Hajnicz, 2014) of Składnica (Woliński et al., 2011). The set is a rather small but carefully prepared resource and contains constituency parse trees for Polish sentences. The adapted version of Składnica (0.5) contains 8241 manually validated trees. Sentence tokens are annotated with fine-grained semantic types represented by Polish wordnet synsets from plWordnet 2.0 plWordnet, Piasecki et al., 2009, http://plwordnet.pwr.wroc.pl/wordnet/). The set contains lexical units of three open parts of speech: adjectives, nouns and verbs. Therefore, only tokens belonging to these POS are annotated (as well as abbreviations and acronyms). Składnica contains about 50K nouns, verbs and adjectives for annotation, and 17410 of them belonging to 2785 (34%) sentences has been already annotated. For 2072 tokens (12%), the lexical unit appropriate in the context has not been found in plWordnet.

## 4 Obtaining Sense Embeddings

In this section we describe the method of obtaining sense-level word embeddings. Unlike most of the approaches described in Section 2.2, our method is applied to manually sense-labeled corpora.

In Wordnet, words either occur in multiple synsets (are therefore ambiguous and subject of WSD), or in one synset (are unambiguous). Our approach is to focus on synsets that contain both ambiguous and unambiguous words. In Skadnica 2.0 (Polish WordNet) we found 28766 synsets matching these criteria and therefore potentially suitable for our experiments.

Let us consider a synset containing following words: 'blemish', 'deface', 'disfigure'. Word 'blemish' appears also in other synsets (is ambiguous) while words 'deface' and 'disfigure' are specific for this synset and do not appear in any other synset (are unambiguous).

We assume that embeddings specific to a sense or synset can be approximated by unambiguous part of the synset. While some researchers such as (Bhingardive et al., 2015) take average embeddings of all synset-specific words, even using glosses and hyperonymy, we use unambiguous words to generate word2vec embedding vector of a sense.

During training, each occurrence of unambiguous word in corpus is substituted for a synset identifier. As in the provided example, each occurrence of 'deface' and 'disfigure' would be replaced by its sense identifier, the same for both unambiguous words. We'll later use these sense vectors to distinguish between senses of ambiguous 'blemish' given their contexts.

We train word2vec vectors using substitution mechanism described above on a dump of all Polish language Wikipedia and 300-million subset of the National Corpus of Polish (Przepiórkowski et al., 2012). The embedding size is set to 100, all other word2vec parameters have the default value as in (Řehůřek and Sojka, 2010). The model is based on lemmatized (base word forms) so only the occurrences of forms with identical lemmas are taken into account.

## 5 Unsupervised Word Sense Recognition

In this section we are proposing a simple unsupervised approach to WSD. The key idea is to use word embeddings in probabilistic interpretation and application comparable to language modeling, however without building any additional models or parameter-rich systems. The method is derived from (Taddy, 2015), where it was used with a bayesian classifier and vector embedding inversion to classify documents.

(Mikolov et al., 2013) describe two alternative methods of generating word embeddings: the skip-gram, which represents conditional probability for a word's context (surrounding words) and CBOW, which targets the conditional probability for each word given its context. None of these corresponds to a likelihood model, but as (Taddy, 2015) note they can be interpreted as components in a composite likelihood approximation. Let w = $[w_1 \dots w_T]$ denote an ordered vector of words. The skip-gram in (Mikolov et al., 2013) yields the pairwise composite log likelihood:

$$logp_{\mathcal{V}}(w) = \sum_{j=1}^{T} \sum_{i=1}^{T} \mathbb{1}_{[1 \leq |k-j| \leq b]} logp_{\mathcal{V}}(w_k|w_j) \tag{1}$$

We use the above formula to compute probability of a sentence. Unambiguous words are represented as their word2vec representations derived directly from corpus. In case of ambiguous words, we substitute them for each possible sense vector (generated from unambiguous parts of synsets, as has been previously described). Therefore, for an ambiguous word to be disambiguated, we generate as many variants of a sentence as there are its senses, and compute each variant's likelihood using formula 1. Ambiguous words which occur in the context are omitted (although we might also replace them with an averaged vector representing all their meanings). Finally, we select the most probable variant.

Because the method involves no model training, we evaluate it directly over the whole data set without dividing it into train and test sets for cross-validation.

## 6 Supervised Word Sense Recognition

In the supervised approach, we train neural network models to predict word senses. In our experiment, neural network model acts as a regression function F transforming word embeddings provided at input into sense (synset identifiers) vectors.

As the network architecture we selected LSTM (Hochreiter and Schmidhuber, 1997). Neural network model consists of one LSTM layer followed by a dense (perceptron) layer at the output. We train the network using mean standard error loss function.

Input data consists of the sequences of five word2vec embeddings: of two words that make left and right symmetric contexts of each input word to be disambiguated, and the word itself represented by the average vector of vectors representing all its senses. Ambiguous words for which there are no embeddings are represented by zero vectors (padded). Zero vectors are also added if the context is too short. This data is used to train LSTM model (Keras 1.0.1 `https://keras.io/`) linked with the subsequent dense layer with sigmoid activation function.

At the final step, we transform the output into synsets rather than vectors. We select the most appropriate sense from a set of possible sense inventory, taking into account continuous output structure. In this step, neural network output layer (which is a vector of the same size as input embeddings, but transformed) is compared with each possible sense vector. To compare vectors, we use cosine similarity measure, defined between any two vectors.

We compute cosine similarity between neural network output vector $nnv$ and each sense from possible sense inventory S, and select the sense with the maximum cosine similarity towards $nnv$.

To test each neural network set-up we use 30-fold cross-validation.

## 7 Results

In this section we put summary of the results obtained on our test set, as well as two baseline results. The corpus consisted of 2785 sentences and 303 occurences of annotated ambiguous words which could be disambiguated by our algorithms, i.e. there were unambiguous equivalents of its senses and there were appropiate word embeddings for at least one of the other senses of this word. There were 5571 occurences of words which occurred only in one sense.

Table 1 presents precision of both tested methods computed over the Skladnica dataset. The set contains 344 occurrences of ambiguous words which were eligible for our method. For the unsupervised approach we tested a window of 5 and 10 words around the analyzed word.

The ambiguous words from the sentence other than the one being disambiguated at the moment are either omitted or represented as a vector representing all their occurrences. The *uniq* variant omit all other ambiguous words from the sentence while in the *all* variant we use not disambiguated representation of these words.

| **Method** | Settings | Precision |
|---|---|---|
| random baseline | N/A | 0.47 |
| MFS baseline | N/A | 0.73 |
| pagerank | N/A | 0.52 |
| unsupervised | 5 word, all | 0.507 |
| | 5 word, uniq | 0.507 |
| | 10 word, uniq | 0.529 |
| | 10 word, all | 0.513 |
| supervised | 750 epochs | 0.673 |
| | 1000 epochs | 0.680 |
| | 2000 epochs | 0.690 |
| | 4000 epochs | 0.667 |

Table 1: Precision of word-sense disambiguation methods for Polish.

In the supervised approach the best results were obtained for 2000 epochs but they did not differ much from these obtained after 1000 epochs.
For comparison, we include two baseline values:

- random baseline select random sense from uniform random probability distribution,
- MFS baseline use most frequent sense as computed from the same corpus (There is no other available sense frequency data for Polish, that could be obtained from manually annotated sources.)

The table also includes results computed using pagerank WSD algorithm developed at the PWR (Kędzia et al., 2015). These results were obtained for all the ambiguous words occurring within the sample, so cannot be directly compared to our results.

As the results indicate, unsupervised method performs at the level of random sense selection.

Below there are two examples of the analyzed sentences.

- *lęk przed nicością łączy się z doświadczeniem pustki* 'fear of nothingness combines with the experience of emptiness': in this sentence, Polish ambiguous words 'nothingness' and 'emptiness' were resolved correctly while an ambiguous words 'experience' does not have unambiguous equivalents.
- *na tym nie kończą się problemy* 'that does not stop problems': in this example ambiguous word 'problem' was not resolved correctly, but this case is difficult also for humans.

The low quality of the results might be the effect of a relatively short context available as the analysed text is not continuous.

It might have also pointed out to the difficulty of the test set. Senses in plWodnet are very numerous and hard to differentiate even for human. But the results of the supervised method falsify this assumption.

Our supervised approach gave much better results although they are also not very good as the amount of annotated data is rather small. In this approach more epochs resulted in a slight model over-fitting.

## 8 Conclusions

Our work introduced two methods of word sense disambiguation based on word embeddings, supervised and unsupervised. The first approach assumes probabilistic interpretation of embeddings and computes log probability from sequences of word embedding vectors. In place of ambiguous word we put embeddings specific for each possible sense and evaluate the likelihood of thus obtained sentences. Finally we select the most probable sentence. The second supervised method is based on a neural network trained to learn a context-sensitive transformation that maps an input vector of ambiguous word into an output vector representing its sense. We compared the performance of both methods on corpora with manual annotations of word senses from the Polish wordnet (plWordnet). The results show the low quality of the unsupervised method and suggest the superiority of the supervised version in comparison to the pagerank method on the set of words which were eligible for our approach. Although the baseline in which just the most frequent sense is chosen is still a little better, this is probably due to a very limited training set available for Polish.

## Acknowledgments

The paper is partially supported by the Polish National Science Centre project *Compositional distributional semantic models for identification, discrimination and disambiguation of senses in Polish texts* number 2014/15/B/ST6/05186.

## References

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.

Pierpaolo Basile, Annalina Caputo, and Giovanni Semeraro. 2014. An enhanced lesk word sense disambiguation algorithm through a distributional semantic model. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics*, Dublin, Irleand. Association for Computational Linguistics.

Sudha Bhingardive, Dhirendra Singh, Rudra Murthy, Hanumant Redkar, and Pushpak Bhattacharyya. 2015. Unsupervised most frequent sense detection using word embeddings. In *DENVER*.

Bartosz Broda, Michał Marcinczuk, Marek Maziarz, Adam Radziszewski, and Adam Wardynski. 2012. KPWr: Towards a free corpus of polish. *Proceedings of LREC'12*.

Elżbieta Hajnicz. 2014. Lexico-semantic annotation of *składnica* treebank by means of PLWN lexical units. In Heili Orav, Christiane Fellbaum, and Piek Vossen, editors, *Proceedings of the 7th International WordNet Conference (GWC 2014)*, pages 23–31, Tartu, Estonia. University of Tartu.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November.

Eric H. Huang, Richard Socher, Christopher D. Manning, and Andrew Y. Ng. 2012. Improving word representations via global context and multiple word prototypes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, pages 873–882, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ignacio Iacobacci, Mohammad Taher Pilehvar, and Roberto Navigli. 2015. Sensembed: Learning sense embeddings for word and relational similarity. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the*

*7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 95–105.

Łukasz Kobyliński and Mateusz Kopeć. 2012. Semantic similarity functions in word sense disambiguation. In Petrand Horák Sojka, Alešand Kopeček, and Karel Ivanand Pala, editors, *Text, Speech and Dialogue: 15th International Conference, TSD 2012, Brno, Czech Republic, September 3-7, 2012. Proceedings*, pages 31–38, Berlin, Heidelberg. Springer.

Łukasz Kobyliński. 2012. Mining class association rules for word sense disambiguation. In Pascal Bouvry, Mieczysław A. Kłopotek, Franck Leprevost, Małgorzata Marciniak, Agnieszka Mykowiecka, and Henryk Rybiński, editors, *Security and Intelligent Information Systems: International Joint Conference, SIIS 2011, Warsaw, Poland, June 13-14, 2011, Revised Selected Papers*, volume 7053 of *Lecture Notes in Computer Science*, pages 307–317, Berlin, Heidelberg. Springer.

Paweł Kędzia, Maciej Piasecki, and Marlena Orlińska. 2015. Word sense disambiguation based on large scale Polish CLARIN heterogeneous lexical resources. *Cognitive Studies| Études cognitives*, 15:269–292.

Rada Mihalcea, Paul Tarau, and Elizabeth Figa. 2004. Pagerank on semantic networks, with application to word sense disambiguation. In *Proceedings of the 20th International Conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 746–751.

Arvind Neelakantan, Jeevan Shankar, Alexandre Passos, and Andrew McCallum. 2014. Efficient nonparametric estimation of multiple embeddings per word in vector space. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1059–1069.

Adam Przepiórkowski, Mirosław Bańko, Rafał L. Górski, and Barbara Lewandowska-Tomaszczyk, editors. 2012. *Narodowy Korpus Języka Polskiego*. Wydawnictwo Naukowe PWN, Warsaw.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May. ELRA. `http://is.muni.cz/publication/884893/en`.

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

Hinrich Schütze. 1998. Automatic word sense discrimination. *Computational Linguistics*, 24 (1):97–123.

Matt Taddy. 2015. Document classification by inversion of distributed language representations. *CoRR*, abs/1504.07295.

Kaveh Taghipou and Hwee Tou Ng. 2015. Semi-supervised word sense disambiguation using word embeddings in general and specific domains. In *Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL*, page 314–323. Association for Computational Linguistics.

Rocco Tripodi and Marcello Pelillo. 2017. A game-theoretic approach to word sense disambiguation. *Computational Linguistics*.

Marcin Woliński, Katarzyna Głowińska, and Marek Świdziński. 2011. A preliminary version of Składnica—a treebank of Polish. In Zygmunt Vetulani, editor, *Proceedings of the 5th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*, pages 299–303, Poznań, Poland.