Estimating senses with sets of lexically related words for Polish word sense disambiguation

Szymon Rutkowski
Institute of Computer Science
Warsaw, Poland
szymon@szymonrutkowski.pl

Piotr Rychlik
Institute of Computer Science
Warsaw, Poland
rychlik@ipipan.waw.pl

Agnieszka Mykowiecka
Institute of Computer Science
Warsaw, Poland
agn@ipipan.waw.pl

Abstract

We propose a new algorithm for word sense disambiguation, exploiting data from a WordNet with many types of lexical relations, such as plWordNet for Polish. In this method, sense probabilities in context are approximated with a language model. To estimate the likelihood of a sense appearing amidst the word sequence, the token being disambiguated is substituted with words related lexically to the given sense or words appearing in its WordNet gloss. We test this approach on a set of sense-annotated Polish sentences with a number of neural language models. Our best setup achieves the accuracy score of 55.12% (72.02% when first senses are excluded), up from 51.77% of an existing PageRank-based method. While not exceeding the first (often meaning most frequent) sense baseline in the standard case, this encourages further research on combining WordNet data with neural models.

1 Introduction

Ambiguity is an inherent feature of natural languages in which there is no one-to-one relation between the vocabulary of very many words and the set of meanings which these words represent. Although there are more and more applications in which disambiguation step is not clearly distinguished, explicit identification in which sense a particular word is used in a given context remains important in many situations.

If we aim at selecting a specific sense from a given inventory like WordNet (A. Miller, 1995), this task is called Word Sense Disambiguation (WSD) and was commonly addressed in one of two ways. The first one treats the task as a standard word classification problem solved using any of the supervised learning techniques. The hard part of applying this approach is obtaining satisfactorily large annotated data sets for relatively big subset of senses, even if the annotation can be partially bootstrapped in a semi-supervised manner, for example using label propagation (Yuan et al., 2016). Manual labelling of data with word senses takes time, and agreement between annotators is usually not very high. Another problem is that a lot of text has to be processed to collect occurrences of several (or even more) senses of each word.

This is why the second approach to WSD seems to be more common. In this type of solutions, information included directly or indirectly in lexical databases, especially WordNet, is used either to generate additional features or as the only data source (in the algorithms based on analysis of knowledge graph structure). Recently, vector word representations and neural network architectures have started to be widely used. Our solution combines neural models trained on a large text corpus with information extracted from the plWordNet (Piasecki et al., 2009).

2 Related Work

The problem of resolving lexical ambiguity has a long and complicated history. This task is one of the oldest problems in computational linguistics and machine translation research, but its definition and role in natural language processing (NLP) community's efforts changed over time in many ways. Agirre and Edmonde (2006) wrote that although solutions of one specific version of the problem – an explicit task of resolving fine-grained and coarse-grained ambiguity to a fixed inventory of senses – showed, at the Senseval-3 conference (Mihalcea et al., 2004a), consistent and respectable accuracy levels, this success did not lead to better performance in real applications. They opined that WSD as a topic of study

seemed to diverge from research on NLP applications, "despite several efforts to investigate and demonstrate its utility".

The authors of the best solution at that time (Mihalcea et al., 2004b) reported an accuracy score of 0.65, which was at human levels according to inter-annotator agreement. Their method requires constructing a graph with all senses of words that are present in the text. A PageRank-like algorithm is applied to this graph for choosing the most salient senses, combined with the Lesk algorithm (Lesk, 1986) and most frequent senses heuristics. Although this system achieved the best result, accuracy of 0.65 was not satisfactory for industrial NLP applications and, with no direct enhancement in view, research on the WSD task was receiving waning interest. But it has not ceased entirely, because consistency of human annotators at the level of 0.65 suggested that either the instructions or the sense inventory itself are not consistent enough, as in everyday language use we usually do not observe such big discrepancies.

Many researchers explored different measures for graph connectivity which might be useful for the WSD task (Navigli and Lapata, 2007). In the SemEval-2013 Task 12, linked data for different languages were also used for this purpose (Navigli et al., 2013; Panchenko et al., 2017). With the increasing popularity of distributional semantic approach, many experiments exploiting word embeddings as an additional or the only source of information were performed (Iacobacci et al., 2016; O et al., 2018).

While the evidence from research on the WSD task for English appears contradictory, it should be instructive to see how approaches perform on data in different languages with their unique problems and qualities. For Polish, relatively little was investigated on this subject, but some results were published. Leaving out very early experiments which constrained themselves to a purpose-built set of senses for a group of selected words, we should mention (Kędzia et al., 2015) who employed the graph-based method proposed by (Mihalcea et al., 2004b) and (Agirre et al., 2014), utilizing data from plWordNet integrated by the authors with existing SUMO ontology.

Recently, (Wawer and Mykowiecka, 2017) proposed an approach where probability of senses in context is assessed by replacing the disambiguated word with unambiguous members of their synsets. This method, while obviously limited to cases where such unambiguous words can be found in the token's synsets, produced promising results when tested on data from (Hajnicz, 2014). The general idea of estimating context probability with replacements from a WordNet is similar to the one presented in this paper, but we argue that it can be exploited more fully using lexical relations.

3 Test Data Description

Our test data consists of 1000 sentences selected from the manually annotated part of the NKJP (National Corpus of Polish) (Przepiórkowski et al., 2012). The sentences were chosen randomly, but we excluded transcribed speech and internet sources. We collected 24,535 tokens of 9,741 token types in total. All nouns, verbs, adjectives and adverbs were manually annotated with plWordNet 3.1 senses by appropriately trained linguists.

As the annotation process is very time consuming, only a part of the data was annotated by both of them and they agreed on 83% of tokens. This is comparable to the measures of inter-annotator agreement in Senseval competitions (Green et al., 2017). In Senseval-1, the 80% agreement was eventually achieved by allowing for discussion and revisions of ambiguities in lexical entries before final tagging. In Senseval-2, the agreement on verb annotation was initially 71%, but after grouping some senses into more coarse-grained ones it rose to 82%.

4 Method

Our approach is conceptually rather simple: for every ambiguous word (w), we would like to select the sense (s^*) with the highest probability given the form and context (c) of the word:

$$s^* = \arg\max_s P(s|w,c) \tag{1}$$

However, since there is no clear way to obtain $P(s)$ directly, we approximate it with some set R_s of word forms related to the sense in question. One way of combining the evidence from members of R_s is to average their probabilities in the context:

$$s^* = \arg\max_s \frac{\sum_{r \in R_s} P(r|w,c)}{|R_s|} \tag{2}$$

We also test the variant where the highest probability estimated for a related word is taken to represent the whole sense:

$$s^* = \arg\max_s \max_{r \in R_s} P(r|w, c) \quad (3)$$

Once r is an explicitly designated word form or lemma, a language model capable of predicting probability of word sequences can be used to predict $P(r|w, c)$. Note that we only have to decide whether the word is likely to occur in the context or not; there is no need for a full distribution of words that could occur there otherwise. Thus, following word2vec's negative sampling method (Mikolov et al., 2013), we train our language model only to discriminate between true and "garbled" fragments of text. Specifically, we obtain negative samples for training from positive (real) ones by shuffling the order of words and replacing some of them with random entries from vocabulary.

We define the set of related words (neighbours) as follows, using relations between lexical units, i.e. senses, and synsets in plWordNet. For relations among lexical units, we include lemmas of the related units. For relations between synsets, we include lemmas of all lexical units belonging to the related synsets. Also words from the same synset as the lexical unit in question are taken into account. Finally, words from the lexical unit description (gloss) are also treated as neighbours.

Intuitively, swapping the ambiguous word for related terms, such as hyponyms or hypernyms, is a method similar to heuristics that a human could use. To give an English example, to disambiguate the word *jam* in the phrase *I brought you some apple jam*, one might try to substitute some synonyms, and estimate how much sense they would make semantically in the context: *I brought you some apple preserve*, *I brought you some apple mess*, *I brought you some apple session*, etc. The ones that have the highest probability of occurring would tend to be those which are related to true sense of the original word.

Since it is possible for a sense to not yield any neighbours, because of having no relevant relations, we use the probability of the original context (that is, the one containing the word being disambiguated) as the baseline probability for all senses. Only when a sense does have some other words related to it, the baseline is replaced with either the average or the maximum of their estimated probabilities.

Estimates for all senses, computed separately, in practice rarely sum to one. We normalise them before making the decision, although this does not influence the final verdict of the model. If many senses have the same, highest estimated probability, we choose from among them at random.

5 Experiments

As there are many types of relations (over 40 in plWordNet 3.1, not counting subtypes), we selected some of them that seemed particularly useful for our task, and grouped them into three subsets. The first one includes synonymy (including belonging to the same synset), hypernymy and hyponymy, the second includes also antonyms, and the third one, apart from everything from the first subset, incorporates various types of meronymy and other relation types that seem to connect to words that would be adequate replacements for their neighbours in the sentence. For example, in plWordNet there is a number of relations connecting verbs that presuppose or imply each other, or adjectives that differ by magnitude of the quality that they describe.

We test [1] how accurate are predictions based on (1, 2, 3) those three subsets, (4) combination of all of them, (5) on words from glosses only, (6) on words in glosses and all words obtained from relation subsets.

The basic context probability estimator, serving as the core of our system, is an LSTM (Hochreiter and Schmidhuber, 1997) network, taking nine word vectors as its input, with the disambiguated word position in the middle. The hidden size of an LSTM cell is as little as 9 – we have tried bigger values, such as 64 and 128, but they performed worse.

The last output of the LSTM is squashed with sigmoid function and interpreted as probability. Previously published set of word embeddings (Mykowiecka et al., 2017) was used for vectorising sentences. We used 300-dimensional vectors from a word2vec model, trained using continuous bags-of-words and negative sampling on lemmatised corpus consisting of NKJP and the Polish Wikipedia. As an alternative, we also tested vectorising contexts with ELMo embeddings (Peters et al., 2018), using the ELMoForManyLangs package (Che et al., 2018; Fares et al., 2017). Both LSTMs were trained on the manually annotated, balanced portion of NKJP.

[1] The source code is available at `zil.ipipan.waw.pl/CoDeS`.

Neighbour subset	RNN/avg	Gensim/avg		ELMo/avg	RNN/max	Gensim/max		ELMo/max
Relations 1	42.94%	41.36%	40.28%	43.45%	43.90%	40.36%	39.53%	43.75%
Relations 1+2	44.70%	43.06%	42.53%	44.99%	43.89%	40.73%	39.68%	43.52%
Relations 1+3	45.58%	44.68%	44.77%	46.04%	44.37%	40.34%	40.62%	44.00%
Relations 1+2+3	53.93%	50.83%	54.00%	54.08%	54.92%	50.57%	54.97%	55.08%
Glosses	43.93%	43.37%	43.90%	44.18%	44.70%	42.85%	44.80%	42.85%
Glosses + Rels	53.88%	50.88%	54.09%	54.01%	**55.12%**	50.52%	54.89%	55.08%

Table 1: Prediction accuracy measured for all ambiguous cases in our corpus: 'RNN' – basic model, 'Gensim' – Gensim implementation of sequence likelihood (for nine word window and full sentence case), 'ELMo' – RNN with ELMo embeddings instead of word vectors; 'avg' – taking the average probability of all neighbours, 'max' – taking the maximal value.

Neighbour subset	RNN/avg	Gensim/avg		ELMo/avg	RNN/max	Gensim/max		ELMo/max
Relations 1	55.51%	54.16%	52.97%	55.69%	56.70%	53.26%	51.45%	55.75%
Relations 1+2	57.73%	56.23%	55.72%	57.95%	56.68%	53.65%	51.85%	55.64%
Relations 1+3	58.40%	59.63%	57.23%	58.94%	57.58%	53.27%	52.84%	56.40%
Relations 1+2+3	70.01%	66.94%	69.99%	70.22%	71.77%	65.35%	71.82%	72.02%
Glosses	56.58%	57.23%	56.33%	57.01%	56.93%	56.53%	57.85%	58.38%
Glosses + Rels	70.60%	66.97%	70.05%	70.12%	**72.02%**	65.29%	71.61%	71.74%

Table 2: Prediction accuracy measured for cases where the first sense was not the correct one.

These setups were compared with an existing hierarchical softmax model that was trained on full, unbalanced version of NKJP and Polish Wikipedia corpus. It exists in Gensim (Řehůřek and Sojka, 2010) format, which allows for scoring probabilities (or more precisely, log likelihoods) of entire sentences, which can be also applied to sentence fragments. As explained in (Taddy, 2015), log likelihood of a sentence $S = [w_1, w_2, \ldots, w_n]$ is defined as the pairwise composite log likelihood:

$$\mathcal{L}(S) = \sum_{i=1}^{|S|} \sum_{j=1}^{|S|} \ell(w_i, w_j),$$

where

$$\ell(w_i, w_j) = \begin{cases} \log P(w_i|w_j) & \text{if} \quad 1 \leq |j-i| \leq b \\ 0 & \text{otherwise} \end{cases}$$

With the skipgram variant of word2vec model which was used here, $P(w_i|w_j)$ denotes the conditional probability of a context word w_i for a target word w_j. The number b is the window size used in model training. In our case, it is 5, so the whole window contains 11 words.

The Gensim implementation uses a shallower regular word2vec architecture instead of recurrent networks. It is also, in contrast to the RNN, not intrinsically aware of word order.

Results in Table 1 show, for all models, a sharp improvement of quality when all types of relations are considered, as opposed to smaller subsets. It seems that regardless of whether neighbour words make sense as replacements for the word being disambiguated, their semantic relatedness to the context facilitates recognition of the correct sense. On the other hand, glosses appear to work relatively poorly as a source of neighbours for our solution. This may be partially explained by the lack of consistent formatting of glosses in Polish WordNet, where definitions, examples and other metadata are mixed in a couple of ways in one field of the database.

For almost all methods, the approach of taking the maximum probability instead of the average yielded better results. The only exceptions are some weaker versions of Gensim and ELMo approaches. We hypothesise that neighbours that seem the most likely in given context may indeed reflect the best whether the sense that they represent is appropriate. A possible counterargument would point towards negligible improvements caused by this change to the approach based entirely on words from glosses. Although one would think that ignoring junk words from metadata would markedly raise chances of the true sense, this appears not to be the case.

It should be noted that these results, unfortunately, are still lower than the baseline of 59.77% cases where the correct sense is the first variant in Polish WordNet (which often, but not always, happens to be the most frequent one in Polish language). It is a known issue in development of WSD solutions, and for our data this result is even

higher than MFS (Most Frequent Sense) accuracy cited for English in (Agirre and Edmonds, 2006), i.e. 46.4%. However, most measurements exceed the lower baseline of assigning sense annotations at random (45.08% accuracy).

Among all the models of context probability evaluation, the basic word vector-based LSTM performed the best. Its superiority over ELMo seems to be linked to operating on lemmas, instead of forms, as the pretrained ELMo embedder. Due to rich morphology of Polish, information in a corpus is markedly easier to generalise if the inflections are abstracted away. Our preliminary tests with training a form-based LSTM operating on word vectors confirmed this hypothesis by degrading maximum accuracy, although it still fared better than ELMo on smaller relation subsets.

It is true that any RNN shows an improvement over the Gensim non-recursive solution, which is unaware of word order. We additionally ran more relaxed tests where this model was allowed to see whole sentences (as the Gensim package interface suggests to do), and even then it was not able to reach the level of RNNs.

We also present results obtained on non-first variant cases only, in Table 2. It appears that our algorithm is capable of relatively precise treatment of less frequent senses, even though it has issues with separating them from the dominant ones. Here we still observe the superiority of LSTM based on word vectors with taking the maximum probability.

We compared our results with the only one other general purpose method for solving Polish WSD task described in (Kędzia et al., 2015). We carried out the test on our test set using two taggers: WCRFT2 (Radziszewski and Warzocha, 2014) and MorphoDiTa (Straka and Straková, 2014). In both cases, we have achieved accuracy of around 51% (more precisely, 51.05% for WCRFT2 and 51.77% for MorphoDiTa). All versions of our algorithm surpassed these scores, as long as they considered all the subsets of plWordNet relations.

6 Conclusions

We present a new method of disambiguating senses in Polish texts using lexical relations from the plWordNet database. We test various relation subsets and approaches to modeling probability of contexts.

The WSD problem for Polish is still far from being solved. No published results were able to exceed 70% accuracy, which would move them closer to matching those published for English. It is worth pointing out, however, that our accuracy for cases where the first WordNet sense was excluded does approach this level of performance. Perhaps finding a way to distinguish the most typical contexts, where one can expect these most frequent senses to occur, can greatly help the overall usefulness of the system.

Judging from our findings, there is little to be gained by enhancing language models within the same framework of estimating sense likelihoods. The results do show potential in combining modern machine learning with creative use of existing knowledge bases, and should encourage further research in this direction.

Acknowledgments

This work was supported by the Polish National Science Centre project 2014/15/B/ST6/05186.

References

George A. Miller. 1995. Wordnet: A lexical database for english. *Communications of the ACM*, 38:39–, 11.

Eneco Agirre and Philip Edmonds, editors. 2006. *Word Sense Disambiguation: Algorithms and Applications*. Springer.

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.

Wanxiang Che, Yijia Liu, Yuxuan Wang, Bo Zheng, and Ting Liu. 2018. Towards better UD parsing: Deep contextualized word embeddings, ensemble, and treebank concatenation. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 55–64, Brussels, Belgium, October. Association for Computational Linguistics.

Murhaf Fares, Andrey Kutuzov, Stephan Oepen, and Erik Velldal. 2017. Word vectors, reuse, and replicability: Towards a community repository of large-text resources. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 271–276, Gothenburg, Sweden, May. Association for Computational Linguistics.

Meredith Green, Orin Hargraves, Claire Bonial, Jinying Chen, Lindsay Clark, and Martha Palmer. 2017. Verb/ontonotes-based sense annotation. In *Handbook on Linguistic Annotation*. Springer.

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.

Ignacio Iacobacci, Mohammad Taher Pilehvar, and Roberto Navigli. 2016. Embeddings for word sense disambiguation: An evaluation study. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 897–907.

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.

Michael E. Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the SIGDOC Conference*.

Rada Mihalcea, Timothy Chklovski, and Adam Kilgarriff. 2004a. The Senseval-3 English lexical sample task. In *Proceedings of SENSEVAL-3 Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 25–28. ACL.

Rada Mihalcea, Paul Tarau, and Elizabeth Figa. 2004b. 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, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Agnieszka Mykowiecka, Małgorzata Marciniak, and Piotr Rychlik. 2017. Testing word embeddings for Polish. *Cognitive Studies / Études Cognitives*, 17:1–19.

Roberto Navigli and Mirella Lapata. 2007. Graph connectivity measures for unsupervised word sense disambiguation. In *Proceedings of the IJCAI*, pages 1683–1688.

Roberto Navigli, David Jurgens, and Daniele Vannella. 2013. Semeval-2013 task 12: Multilingual word sense disambiguation. In *Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 222–231.

Dongsuk O, Sunjae Kwon, Kyungsun Kim, and Youngjoong Ko. 2018. Word sense disambiguation based on word similarity calculation using word vector representation from a knowledge-based graph. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*.

Alexander Panchenko, Stefano Faralli, Simone Paolo Ponzetto, and Chris Biemann. 2017. Using linked disambiguated distributional networks for word sense disambiguation. In *Proceedings of the 1st Workshop on Sense, Concept and Entity Representations and their Applications*, pages 72–78. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *CoRR*, abs/1802.05365.

Maciej Piasecki, Stanisław Szpakowicz, and Bartosz Broda. 2009. *A Wordnet from the Ground Up*. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław.

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

Adam Radziszewski and Radosław Warzocha. 2014. WCRFT2. CLARIN-PL digital repository.

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`.

Milan Straka and Jana Straková. 2014. MorphoDiTa: Morphological dictionary and tagger. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

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

Aleksander Wawer and Agnieszka Mykowiecka. 2017. Supervised and unsupervised word sense disambiguation on word embedding vectors of unambigous synonyms. In *Proceedings of the 1st Workshop on Sense, Concept and Entity Representations and their Applications*, pages 120–125. Association for Computational Linguistics.

Dayu Yuan, Julian Richardson, Ryan Doherty, Colin Evans, and Eric Altendorf. 2016. Semi-supervised word sense disambiguation with neural models.