

Word Sense Disambiguation Based on Constrained Random Walks in Linked Semantic Networks

Arkadiusz Janz, Maciej Piasecki

Wrocław University of Science and Technology, Poland

{arkadiusz.janz|maciej.piasecki}@pwr.edu.pl

Abstract

Word Sense Disambiguation remains a challenging NLP task. Due to the lack of annotated training data, especially for rare senses, the supervised approaches are usually designed for specific subdomains limited to a narrow subset of identified senses. Recent advances in this area have shown that knowledge-based approaches are more scalable and obtain more promising results in all-words WSD scenarios. In this work we present a faster WSD algorithm based on the Monte Carlo approximation of sense probabilities given a context using constrained random walks over linked semantic networks. We show that the local semantic relatedness is mostly sufficient to successfully identify correct senses when an extensive knowledge base and a proper weighting scheme are used. The proposed methods are evaluated on English (SenseEval, SemEval) and Polish (Składnica, KPWr) datasets.

1 Introduction

Semantic analysis is the process of understanding the underlying meaning in text. Building a rich semantic representation is a crucial element of modern Natural Language Processing NLP. It allows us to build comprehensive text representations in order to analyse the underlying text meaning more effectively. Semantic information helps us to resolve the issue of textual ambiguity. The main aim of semantic analysis is to differentiate texts which use the same vocabulary yet present different ideas about the same topic. One of the most important building blocks of semantic analysis is *word sense disambiguation* (WSD). WSD aims at solving the problem of lexical ambiguity, i.e. two

or more meanings being represented by one word e.g. English words *space*, *shape*, or Polish words *tło* (background color, a context or an incidental music) *wnętrze* (a room, a soul or a set). This lexical ambiguity is still an open issue for semantic analysis due to the lack of required training data and the computational power required to process the growing number of possible classes. The task of word sense disambiguation is usually solved by mapping words onto their senses given a particular sense inventory.

WSD can be performed in a supervised way, i.e. on the basis of the annotated lexical meanings in training texts, or in an unsupervised way, i.e. sense induction from texts.

Supervised machine learning approaches can achieve a very good performance when trained on a large training data annotated with good inter-annotator agreement and well designed features.

However, it is difficult to acquire large training data for WSD because training data is usually manually annotated and manual annotation is labour-intensive and thus costly¹. Semi-automatic approaches on the other hand often result in lower quality data blurred with statistical noise.

Thus, the best performing supervised approaches to WSD task are usually limited to a specific narrow subset of training vocabulary. Moreover, supervised approaches are strongly connected with the underlying domain of the training data. Sense induction methods require only large amounts of text, but representative in relation to all word senses. However, senses induced automatically are mostly difficult to be matched against dictionaries or other word sense inventories created manually. Thus, semi-supervised methods based on lexical knowledge bases describing word senses, e.g. wordnets, offer a poten-

¹It is barely possible to scale it up to a couple of dozens of thousands words well described according to all its senses

tially good compromise. Such methods can cover, at least potentially, all word senses described in a lexical knowledge base. Very large wordnets describing more than 100,000 words (lemmas) and their senses by hundreds of lexico-semantic relation instances have been constructed for several languages, including the English and Polish language. A large and dense network of relations seems to be a good basis for mapping word occurrences in texts onto their senses.

An enormous amount of semantic data appeared on the Web in recent years. With the growth of resources joining the Linked Open Data collection the available semantic information becomes a very important knowledge source for WSD tasks.

Semi-supervised methods are very often based on the idea of mapping texts onto the wordnet graph of lexico-semantic relations as the initial activation of the graph nodes. Next a recursive spreading activation method, mostly based on PageRank (Page et al., 1998) is applied. However, as seen in the following paragraphs, such methods raise problems with efficiency.

In this paper we explore Monte Carlo methods based on a random walk algorithm for the task of Word Sense Disambiguation. We show that the lexical knowledge base with its expansions and the way they are exploited has a strong impact on WSD performance and the proposed method allows for the efficient utilisation of large lexical knowledge bases. The presented solutions are evaluated on popular Polish and English benchmark datasets.

2 Related Work

The first group of approaches is based on supervised machine learning algorithms. The researchers have adapted many different ML methods for WSD task. One can find the classical NLP approaches based on feature engineering and popular classification algorithms e.g. decision trees, decision lists, Naive Bayes solutions, kNN, adaptive boosting, as well as more modern approaches using Deep Learning with LSTM, biLSTM and more sophisticated variants of neural architectures. The main issue is that most of the words are strongly imbalanced in terms of their sense distribution, thus, due to the lack of required training data, the supervised approaches present a lower recall in all-words WSD setting.

The knowledge-based solutions use the struc-

tural properties of existing sense inventories e.g. wordnets.

One of the first solutions to WSD task for Polish (Baś et al., 2008) was mainly focused on supervised approaches. The authors trained the classifiers on a relatively small dataset for a pre-selected, annotated vocabulary. Later works were focused mainly on WSD methods close to sense induction from text corpora, e.g. (Broda and Piasecki, 2011).

Weakly supervised WSD approaches are usually based on sense inventories and their semantic structure describing senses. The main assumption is that the words being a part of a text can be mapped onto their potential senses existing in a given sense inventory, mainly the synsets existing in a wordnet. The senses activate local subgraphs of wordnet’s semantic structure to activate meanings possibly being relevant to a given text fragment. As the initial activation is sparse a kind of spreading activation algorithm is next applied in order to recursively concentrate this information in some “hot” areas and identify word senses located in them or close to them. The identified synsets should be the most likely senses for words in the text. There are several parameters to set in this general scheme: the initial activation (coming from the text words), spreading activation algorithm (topology and relations) and identification of association between “hot” areas and senses to be selected. Various methods following this scheme were proposed.

Weakly supervised WSD methods are mostly based on the recursive PageRank algorithm (Page et al., 1998) for spreading activation. Mihalcea et al. (2004) proposed application of the original PageRank to WSD called Static PageRank.

PageRank algorithm (henceforth PR) is an iterative method for ranking nodes in the graph G . In WSD the nodes in G represent synsets and the edges of G correspond to wordnet relations (linking synsets, and in some wordnets also linking specific word senses).

(Agirre and Soroa, 2009; Agirre et al., 2014) proposed a modified version called Personalised PageRank (PPR) in which the values in \mathbf{v} , called personalised vector, depend on the textual context of the disambiguated word. The non-zero score values are assigned to those nodes which are contextually supported. In PPR all words from the context are disambiguated at once. The \mathbf{v} values

are equal to:

$$v[i] = \frac{1}{\frac{CS}{NS(i)}}, \quad i = 1, 2, \dots, N \quad (1)$$

where CS is the number of different lemmas in the context, $NS(i)$ – the number of synsets sharing the same context lemma with the synset i .

In addition a modified version of PPR called *Personalised PageRank Word-to-Word* (PPR_W2W) was also presented by (Agirre and Soroa, 2009; Stevenson et al., 2012), in which a word to be disambiguated is excluded from the occurrence contexts, i.e. all synsets of this word have initial scores in \mathbf{v} set to zero. Thus, PPR_W2W cannot be run once for all ambiguous words in the context. The vector \mathbf{v} must be initialised individually for each ambiguous word in the context – this is a disadvantage of PPR_W2W. A potential advantage is the removal of the effect of mutual amplification of the closely connected senses of the word being disambiguated. All PR-based WSD algorithms showed good performance that is increasing with the enlargement and enrichment of the knowledge base. However, with larger graphs the time of processing increases non-linearly causing a significant drop in efficiency. The problem is especially visible in the case of PPR_W2W in which the algorithm must be restarted several times per context.

In (Kędzia et al., 2014), PR-based WSD algorithm for Polish was presented and run with the help of plWordNet 2.1. The graph consisted of synsets linked by edges representing a selected subset of the synset relations. Next several versions of the PR-based algorithms, namely Static PR, PPR and PPR_W2W was applied to Polish texts and plWordNet 2.2 in (Kędzia et al., 2015). The achieved precision (on *KPWr*) was in the range 42.79%-50.73% for nouns and 29.79%-32.94% for verbs. PPR_W2W produced the best results. Different variants of combining plWordNet with the *Suggested Upper Merged Ontology* (SUMO) (Pease, 2011) on the basis of the mapping constructed in (Kędzia and Piasecki, 2014). All three PR-based algorithm were evaluated. A slight improvement of the precision for nouns up to 50.89% for PPR_W2W could be observed when the two joined graphs were treated as one large graph.

In (Perschina et al., 2015) the authors presented a graph-based algorithm using random walks algo-

rithm for named entity disambiguation. The motivation for their work was the fact that PR-based methods mainly rely on global coherence, but the methods should utilise the local similarity more effectively.

3 Knowledge Base

The general sense inventory for all-words WSD is usually created on the basis of a wordnet. Wordnets can be presented as graphs with nodes representing word senses or synsets (but also with two types of nodes) and edges expressing the structure of the lexico-semantic relations described in a given wordnet. The methods based on lexical knowledge bases usually explore the lexico-semantic relations represented by a wordnet to disambiguate the words given the context.

Most wordnet relations are paradigmatic, but for WSD we also need syntagmatic relations, rarely covered by wordnets, because the plain wordnet structure might be insufficient to successfully disambiguate a text. As the large public sources like Linked Open Data are available, we can try to apply them for the expansion of the lexical knowledge base. They may also contain Named Entities which can be very important for WSD, e.g. helping to identify the narrow semantic context.

More formally, a *lexical knowledge base* is a graph $G(V, E)$ consisting of nodes and edges, where $V = \{v_1, v_2, \dots, v_N\}$ represents a set of concepts (modelling lexical meanings) and $E = \{e_1, e_2, \dots, e_M\}$ a set of edges corresponding to lexico-semantic associations linking these concepts.

3.1 Existing Knowledge Bases

In literature, we can find many attempts to combine Princeton WordNet with resources of many types to obtain a better knowledge base for WSD.

UKB lexical knowledge base, e.g. (Agirre and Soroa, 2009), consists of Princeton WordNet 3.0 or eXtended WordNet (Harabagiu et al., 1999) which was expanded by introducing links extracted from SemCor, manually disambiguated glosses from Princeton WordNet Gloss Corpus, and Wikipedia.

BabelNet (Navigli and Ponzetto, 2012) was initially created by linking the largest multilingual Web encyclopedia, i.e. Wikipedia, with the most popular computational semantic lexicon, i.e.,

WordNet. Later it was expanded with a number of resources:

- OmegaWiki, a large collaborative multilingual dictionary (January 2017 dump);
- Wiktionary, a collaborative project to produce a free-content multilingual dictionary (February 2018 dump);
- Wikidata, a free knowledge base that can be read and edited by humans and machines alike (February 2018 dump);
- Wikiquote, a free online compendium of sourced quotations from notable people and creative works in every language (March 2015 dump);
- VerbNet (Kipperer et al., 2006), a Class-Based Verb Lexicon (version 3.2);
- Microsoft Terminology, a collection of terminologies that can be used to develop localised versions of applications (July 2015 dumps);
- GeoNames, a free geographical database covering all countries and containing over eight million place names (April 2015 dump);
- FrameNet (Ruppenhofer et al., 2016), a lexical database of English that is both human- and machine-readable (version 1.6);
- Open Multilingual WordNet (Bond and Foster, 2013), a collection of wordnets available in different languages.

The mappings provided by BabelNet (especially the links between synsets and Wikipedia pages) were built semi-automatically. We chose to manually map synsets of plWordNet by lexicographers instead of use the semi-automatic mappings provided by BabelNet to have more control over the accuracy of our results.

3.2 Presented Knowledge Base

For the work presented here, two knowledge graphs were built on the basis of the two largest wordnets, namely *plWordNet 3.2* (Maziarz et al., 2016) for Polish, and *Princeton WordNet 3.1* (Fellbaum, 1998) for English. They are mutually mapped on each other as a result of the laborious work of bilingual lexicographers.

3.3 Expansions

The initial performance of the algorithms can be moderate when the knowledge-base is limited only to the plain wordnet structure. We can significantly improve the overall performance by introducing new semantic links to the basis knowledge graph. In this work we expanded the ideas presented in (Agirre and Soroa, 2009), (Agirre et al., 2018) and (Moro et al., 2014). Following the procedure presented in (Agirre et al., 2018) we extended the structure of our knowledge graph by including the links extracted from the the Princeton WordNet Gloss Corpus² including manually disambiguated glosses.

The Suggested Upper Merged Ontology (SUMO) (Pease, 2011) is a formal representation of concepts, organised into hierarchies of classes and subclasses, which is widely used for semantic analysis in NLP. The lexical senses of PWN 3.0 and plWordNet 3.2 have been mapped onto their equivalent concepts of SUMO. The mapping procedure for plWordNet was based on interlingual links existing between plWordNet and PWN (Maziarz et al., 2016) and the initial mapping of PWN senses to SUMO ontology (Kędzia and Piasecki, 2014). We used the structure of SUMO ontology as a more general semantic description for lexical senses existing in wordnet. The semantic structure of our knowledge base was extended with concepts and links existing in the SUMO ontology by attaching the concepts to corresponding synsets and linking them with SUMO relations.

Wikipedia has opened many new opportunities for semantic analysis. The structure of Wikipedia has been used as a knowledge-base for the task of named entity disambiguation (NED), but also adapted for WSD. Graph-based approaches for computing semantic relatedness and disambiguation can be improved by using the semantic information contained in Wikipedia and expand the underlying knowledge base e.g. a wordnet. For this work we decided to add the links extracted from Wikipedia by using the mapping of lexical senses to equivalent Wikipedia articles. For every mapped synset we added new semantic links by analysing the content of the page and extracting monosemous words. The lexical senses of monosemous words were linked to mapped synset.

²<http://wordnetcode.princeton.edu/glosstag.shtml>

4 Methods

The main drawback of PageRank-based methods is that to compute the score of a sense given the context they have to compute the features with respect to the global structure of a given knowledge graph. This means the PageRank values have to be computed for every single node in the graph. To avoid this issue we use only the local approximation of PageRank as it was presented in (Avrachenkov et al., 2007).

4.1 PageRank Based Methods

The computational complexity of Personalised PageRank (PPR) is still a limiting factor for fast, real time WSD of large textual data. The naive algorithm for computing PageRank values requires to iterate over the entire graph. The most popular approach to compute PPR scores is the Power Iteration method (Berkhin, 2005), where the score is defined in a recursive way taking into account the global information. Recently the methods of local approximation of PageRank scores have received a lot of attention, especially in the case of dynamic graphs where the structure of the graph changes in time.

Let $G = (V, E)$ be a graph where V is a set of nodes and E represents a set of edges. The overall number of nodes and edges is N and M , respectively. We can define the adjacency matrix $A_{N \times N}$ of the graph G as $A_{N \times N} = [a_{ij}]$, $i \in \{1, 2, \dots, N\}$, $j \in \{1, 2, \dots, N\}$, where the values a_{ij} are representing links existing in the graph, where $a_{ij} = 1$ if there is an edge pointing from node v_i to v_j , otherwise $a_{ij} = 0$. A Markov transition matrix P can be defined as $P_{N \times N} = [p_{ij}]$ with p_{ij} values being normalised by the number of outgoing links (from node v_i) $p_{ij} = \frac{1}{d_i}$ if $a_{ij} = 1$, otherwise $p_{ij} = 0$. The graph might also contain dangling nodes without any outgoing links. To handle these cases p_{ij} values for dangling nodes are usually replaced by a constant $\frac{1}{N}$, which means adding a link to every node in the graph. The static PageRank can be interpreted then as a stationary distribution π of a Markov chain with final transition matrix \tilde{P} (Google's matrix):

$$\tilde{P} = cP + (1 - c)\frac{1}{N}R \quad (2)$$

The matrix R consists of entries being equal to one. The value $c \in (0, 1)$ is the probability that the random walk follows a link according to dis-

tribution P instead of performing a random jump (usually $c = 0.85$). The $\pi = \pi\tilde{P}$ represents a vector of PageRank scores for nodes of a given graph.

We can also show, that the final PageRank distribution π can be then defined as shown in (3), which directly follows from (2).

$$\pi = \frac{1 - c}{n} \sum_{k=0}^{\infty} c^k P^k \quad (3)$$

One of the first attempts to compute a local approximation of PageRank scores, namely (Fogaras et al., 2005), was based on the property presented in (3) which leads us to Monte Carlo methods. The authors in (Avrachenkov et al., 2007) proved that we can easily approximate PageRank values using random walks with restarts. Let the random walk start from a randomly chosen page and terminate with the probability $(1 - c)$. The random walk runs over the graph and makes the transitions according to transition matrix P with probability c . Let π_j be the final PageRank score for node j in the graph G . So, the initial PageRank formula (2) can be replaced with its rough estimate. We can show that using the properties of equation (3) we can also transform the Personalised PageRank formula and compute a rough estimate of its scores (Avrachenkov et al., 2010) with equation (4).

$$\hat{\pi}_j(s) = (1 - c)\frac{1}{m} \sum_{r=1}^m N_j(s, r) \quad (4)$$

A seed of initial nodes s is used to perform random walks over the graph, where c is the probability that the random walks terminates, m represents the overall number of required random walks, $N_j(s, r)$ is the number of random walks ending for node j , starting from seed node s in r -th random walk. The nodes representing a seed are usually randomly sampled from the graph.

This leads us to the following algorithm of PPR computation:

1. Simulate m runs of the random walks initiated at a node s .
2. Evaluate π_j as a fraction of m random walks which end at node $j \in \{1, 2, \dots, n\}$.

The Monte Carlo approaches are based on random sampling, thus, we may obtain slightly different results for every run, especially when the number of the required iterations is too small to

obtain faster progress of convergence giving a narrow confidence interval for estimated parameter. The unnecessary randomness can be avoided by initiating the random walk from each node in the same way, rather than jumping to random nodes over the graph. The accuracy of our estimate can be also improved by using variance reduction techniques e.g. by using Common Random Numbers approach (Clark, 1990).

4.2 Our Method

The initial idea was to approximate personalised PageRank value using random walk methods over semantic graphs, thus, we adopted the idea presented in (Fogaras et al., 2005) and (Persina et al., 2015). The methods are usually based on random walk algorithm where the main assumption is that the local properties of senses are sufficient to accurately disambiguate the texts. This assumption allows us to reduce processing time of the algorithm when working with large lexical knowledge bases.

The underlying lexical knowledge base is a crucial element of a WSD method and it has a great impact on its performance. The algorithms and their properties presented in Sec. 4.1 were an inspiration for further work on disambiguation algorithms using large semantic networks.

The properties of personalised PageRank algorithm can be a limiting factor for WSD performance. The main issue is that in some specific cases a large node degree does not indicate a high significance, especially when the underlying knowledge base is a heterogenous graph constructed from different semantic resources (ontologies, dictionaries, wordnets). Adding or removing certain links can change the PageRank scores of the target nodes. The main problem is that not all of the links are correct, and to protect PageRank scores against incorrect links we can manipulate the importance of the links by using a heuristic weighting schema. We can also manipulate our seed to make the PPR algorithm more robust to textual noise. Since we are interested only in the importance of our personalisation nodes (representing senses of words for a given context), the initial seed for PPR computation is limited to a set of personalised nodes (usually a set of senses representing the words in a small textual window).

Simulate m random walks of length $L \sim$

$Geom(p)$ starting from each seed node s . The seed consists of the nodes (synsets) representing all of available senses for the words from a given disambiguation context. The importance score γ_j for a given node j is the total number of visits to node j divided by the total number of visited nodes.

$$\hat{\gamma}_j(s) = \frac{1}{m} \sum_{r=1}^m (1-c)[N_j(s,r) + R_j(s,r)] \quad (5)$$

$$\hat{\gamma}_j(s) = \frac{1}{m} \sum_{s \in \mathbf{s}} \sum_{r=1}^m (1-c)[N_j(s,r) + R_j(s,r)] \quad (6)$$

The parameter m denotes the overall number of performed random walks. The expected length of a single random walk is expressed as a geometric distribution $Geom(c)$, usually initialized with the value of parameter c (eq. (4)). To reduce the randomness effects we generate the lengths of walks only once, for all of our seed nodes, which is similar to the variance reduction with Common Random Numbers (Clark, 1990).

$R_j(s,r)$ represents the overall number of random resets, where the decision of a random jump to the starting node s is distributed according to $Bernoulli(p)$. The parameter p is dependent on the smoothed similarity score between usage example and a context.

We can compute a more accurate score estimate by using the recursive property of PageRank formula and averaging the scores of the neighbours. The recomputed final score for a single node j can be defined as:

$$\hat{\gamma}_j(s) = (1-c) \frac{1}{|O(j)|} \sum_{v \in O(j)} \hat{\gamma}_v(s) \quad (7)$$

In (7), the set $O(j)$ represents the neighbourhood of a single personalisation node j and $\hat{\gamma}_v(s)$ is a scoring function computed for neighbour v in $O(j)$.

In (Agirre and Soroa, 2009) and (Agirre et al., 2018) the authors proved that the sense frequency is a strong signal for accurate WSD. To make our methods comparable we decided to use the same source of sense frequency, namely SemCor corpus. Following the idea presented in (Agirre et al., 2018), the final score of a synset is computed as a linear combination of its normalised sense frequency and graph-based scores.

5 Evaluation

The proposed method incorporates several expansions to improve the overall performance of WSD. For the evaluation of WSD in English we utilised available semantic resources: i) manually disambiguated glosses, ii) synsets linked to Wikipedia, iii) the knowledge extracted from SemCor corpus.

In the case of Polish language the available semantic resources are limited. The glosses and usage examples for Polish senses were not disambiguated, thus, the synsets were linked only with the senses of monosemous words appearing in their glosses or usage examples. In the case of Wikipedia-based expansion we used a different mapping – the Polish synsets were partially linked to Wikipedia in a manual process (around 50,000 of synsets were mapped to Polish Wikipedia). We also translated the links by using interlingual synonymy links between Polish and English (enWordNet) parts of plWordNet.

5.1 Datasets

In the experimental part we evaluate our methods on the English dataset described in (Raganato et al., 2017) and the Polish dataset presented in (Kędzia et al., 2015). The former dataset consists of the five standard English texts prepared for Senseval and SemEval competitions for all-words WSD task. A sense inventory for the gold standard annotations was built on a basis of Princeton WordNet 3.0 which makes it approximately compatible with our knowledge-base, i.e. built on extended Princeton WordNet 3.1 in the case of English and plWordNet 3.2 (mapped to WordNet 3.1 and via it to SUMO) in the case of Polish. As we use WordNet 3.1 some small discrepancies can influence the results of the comparison with the test datasets, but they should not have significant impact on the outcome of the comparison.

Regarding the dataset for Polish only partially sense-annotated corpora exist. *Składnica* is a manually annotated dependency treebank with 13,035 sentences written Polish. The updated version of the semantic annotation in *Składnica* was based on plWordNet 3.2.

The *Polish Corpus of Wrocław University of Technology* (henceforth KPWr) consists of 1,127 documents manually annotated with the plWordNet 2.1 senses. The annotation was limited to a pre-selected set of words representing different cases of homonymy and polysemy. The original

dataset consisted of 74 words in total, including 45 nouns and 29 verbs. The overall number of annotated word occurrences was 5,148 with 3,219 noun occurrences and 1,929 verb ones. Since there is a small inconsistency between senses used in the annotation and produced by WSD methods, i.e. coming from plWordNet 3.2, we decided to exclude from it 473 word occurrences for which the appropriate sense does not exist any more in our WSD model.

5.2 Experimental Setting

To accomplish a satisfactory convergence and obtain a small variance of final accuracy we adapted a following set of parameters for our experimental part: the transition probability $c = 0.3$, the overall number of random walks per node $rw_iter = 1000$, the importance of sense frequencies on the final score $\alpha = 0.5$, and $(1 - \alpha)$ for the importance of random walk-based scores. For a Polish dataset we did not use sense frequencies since there are no sense-tagged corpora available to compute the frequencies. The resultant performance (tables tab. 1 and tab. 2) was computed 10 times and averaged.

6 Results

Table 1. presents the final performance for English dataset. The proposed method and introduced knowledge base expansions mostly outperformed a very strong most frequent sense (MSF) baseline. The method achieved the results being on the comparable level with other weakly supervised baseline methods, namely UKB (Agirre et al., 2018), Babely (Moro et al., 2014), and WoSeDon (Kędzia et al., 2015). Table 2. shows the average disambiguation time per context. As it was expected, the Monte Carlo approach provides better time efficiency (PPR vs PPRMC) and a comparable performance to power iteration method. The best results were achieved by mixing all available sources of semantic knowledge: the links extracted from disambiguated glosses, Wikipedia pages, SemCor texts and the links provided by SUMO ontology. The same tendency was observed for Polish dataset (table 3) – the best performance was noted for a mixed setting. The performance obtained for this dataset was also on a comparable level with the approaches based on power iteration method proposed in (Kędzia et al., 2015). PPRMC-1 uses a knowledge graph of synsets only. PPRMC-2 expands the model of

<i>Method</i>	<i>Sens-2</i>	<i>Sens-3</i>	<i>Sem-07</i>	<i>Sem-13</i>	<i>Sem-15</i>
MFS	66.80	66.20	55.20	63.00	67.80
BabelFy	67.00	63.50	51.60	66.40	70.30
UKB-nf	61.30	54.90	42.20	60.90	62.90
UKB-sf	67.50	66.40	54.10	64.00	67.80
UKB-nf-w2w	64.20	54.80	40.00	64.50	64.50
UKB-sf-w2w	68.80	66.10	53.00	68.80	70.30
PPRMC-1	66.26	64.28	54.06	65.08	67.12
PPRMC-2	66.35	65.13	55.60	65.56	66.63
PPRMC-3	66.47	65.94	56.04	65.26	67.71
PPRMC-4	66.78	66.28	56.48	65.90	68.10

Table 1: Averaged F1-scores of PPRMC for different knowledge base expansions: PPRMC-1: graph of synsets only, PPRMC-2: graph of synsets extended with links extracted from manually disambiguated glosses, PPRMC-3: PPRMC-2 extended with links extracted from SemCor and Wikipedia, PPRMC-4: PPRMC-3 with additional links extracted from SUMO ontology.

<i>KB</i>	<i>PPR</i>	<i>PPRMC</i>	<i>#nodes</i>	<i>#edges</i>
base	0.46	0.09	125,303	304,296
+gloss	0.59	0.12	125,303	659,860
+gloss_semcor_wiki	0.69	0.14	125,303	2,041,953
+gloss_semcor_wiki_sumo	0.73	0.16	152,966	2,158,986

Table 2: Average disambiguation time [s] per context for SemEval’15 dataset with respect to the size of underlying knowledge base. PPR settings: *damping_factor*=0.85, *max_iterations*=25, PPRMC settings: *c*=0.3, *rw_count*=1000. The disambiguation context was limited to a small window of three sentences.

<i>Method</i>	<i>Sklad.-N</i>	<i>Sklad.-V</i>	<i>KPWr-N</i>	<i>KPWr-V</i>
PPRMC-1	63.19	44.75	52.92	33.42
PPRMC-2	64.27	46.01	53.24	33.73
PPRMC-3	64.88	46.22	53.31	33.66
PPRMC-4	65.28	46.51	53.66	33.09
WoSeDon	63.92	46.43	53.61	33.71
WoSeDon	64.85	47.29	53.80	34.08
WoSeDon	65.27	47.55	54.02	34.00
WoSeDon	66.18	48.74	54.90	33.89

Table 3: Averaged precision of PPRMC for Polish datasets computed for nouns (N) and verbs (V) separately. A comparison with knowledge-based solution presented in (Kędzia et al., 2015) adapted to the structure of plWordNet 3.2.

PPRMC-1 with links to monosemous words extracted from glosses – in case of Polish glosses, or to disambiguated senses – in case of English glosses. PPRMC-3 adds the links to monosemous words extracted from Wikipedia, and PPRMC-4 introduces additional semantic links from SUMO ontology.

7 Conclusions

Weakly supervised Word Sense Disambiguation (WSD) methods express very good coverage that is only limited by the coverage of the underlying

knowledge base used to define word senses and provide a basis for their disambiguation. However, very good results are obtained only when a rich and large knowledge base is utilised. In such a case, PPR-based WSD methods become slow that limits their applicability. We have shown that an estimation of the PPR algorithm on the basis of the Monte Carlo scheme can preserve most of the quality of the method while gaining a lot in terms of the efficiency of computation. We proposed a WSD tool³ that achieves the performance comparable to the state-of-the-art among the weakly supervised methods, but it is 4-5 times faster. In addition, the efficiency of the proposed tool does not deteriorate so quickly with the increasing complexity of the knowledge base. The proposed method also offers an opportunity to balance between accuracy and processing speed by selecting the number of random walks.

³gitlab.clarin-pl.eu/ajanz/wsd-mc

References

- Eneko Agirre, Oier Lopez de Lacalle, and Aitor Soroa. 2014. Random walks for Knowledge-Based Word Sense Disambiguation. *Computational Linguistics* 40(1):57–84.
- Eneko Agirre, Oier López de Lacalle, and Aitor Soroa. 2018. The risk of sub-optimal use of open source nlp software: Ukb is inadvertently state-of-the-art in knowledge-based wsd. *arXiv preprint arXiv:1805.04277*.
- 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*. Association for Computational Linguistics, pages 33–41.
- Konstantin Avrachenkov, Nelly Litvak, Danil Nemirovsky, and Natalia Osipova. 2007. Monte carlo methods in PageRank computation: When one iteration is sufficient. *SIAM Journal on Numerical Analysis* 45(2):890–904.
- Konstantin Avrachenkov, Nelly Litvak, Danil A Nemirovsky, Elena Smirnova, and Marina Sokol. 2010. Monte carlo methods for top-k personalized pagerank lists and name disambiguation. *arXiv preprint arXiv:1008.3775*.
- Dominik Baś, Bartosz Broda, and Maciej Piasecki. 2008. **Towards word sense disambiguation of Polish**. In *Proceedings of the International Multiconference on Computer Science and Information Technology — 3rd International Symposium Advances in Artificial Intelligence and Applications (AAIA'08)*. pages 65–71. <http://www.proceedings2008.imcsit.org/pliks/162.pdf>.
- Pavel Berkhin. 2005. A survey on pagerank computing. *Internet Mathematics* 2(1):73–120.
- Francis Bond and Ryan Foster. 2013. Linking and extending an Open Multilingual Wordnet. In *51st Annual Meeting of the Association for Computational Linguistics*. ACL, pages 1352–1362.
- Bartosz Broda and Maciej Piasecki. 2011. Evaluating LexCSD in a large scale experiment. *Control and Cybernetics* 40(2):419–436.
- Gordon M Clark. 1990. Use of common random numbers in comparing alternatives. In *Proceedings of the 22nd conference on Winter simulation*. IEEE Press, pages 367–371.
- Christiane Fellbaum, editor. 1998. *WordNet – An Electronic Lexical Database*. The MIT Press.
- Dániel Fogaras, Balázs Rácz, Károly Csalogány, and Tamás Sarlós. 2005. Towards scaling fully personalized pagerank: Algorithms, lower bounds, and experiments. *Internet Mathematics* 2(3):333–358.
- Sanda M. Harabagiu, George A. Miller, and Dan I. Moldovan. 1999. Wordnet 2 - a morphologically and semantically enhanced resource. In *Proceedings of SIGLEX 1999*.
- Paweł Kędzia and Maciej Piasecki. 2014. Ruled-based, interlingual motivated mapping of plwordnet onto sumo ontology. In Nicoletta Calzolari et al., editor, *Proc. Ninth International Conference on Language Resources and Evaluation (LREC'14)*. European Language Resources Association.
- Paweł Kędzia, Maciej Piasecki, Jan Kocoń, and Agnieszka Indyka-Piasecka. 2014. **Distributionally extended network-based Word Sense Disambiguation in semantic clustering of Polish texts**. *IERI Procedia* 10(Complete):38–44. <https://doi.org/10.1016/j.ieri.2014.09.073>.
- Paweł Kędzia, Maciej Piasecki, and Marlena Orlińska. 2015. Word sense disambiguation based on large scale polish clarin heterogeneous lexical resources. *Cognitive Studies* (15).
- Karin Kipperer, Anna Korhonen, Neville Ryant, and Martha Palmer. 2006. Extensive classifications of english verbs. In *Proceedings of the 12th EURALEX International Congress*.
- Marek Maziarz, Maciej Piasecki, Ewa Rudnicka, Stan Szpakowicz, and Paweł Kędzia. 2016. **plwordnet 3.0 – a comprehensive lexical-semantic resource**. In Nicoletta Calzolari, Yuji Matsumoto, and Rashmi Prasad, editors, *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*. ACL, ACL, pages 2259–2268. **ACL Anthology**. <http://www.aclweb.org/anthology/C16-1213>.
- 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*. Association for Computational Linguistics, Stroudsburg, PA, USA, COLING '04. <https://doi.org/10.3115/1220355.1220517>.
- Andrea Moro, 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.
- Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence* 193:217–250.
- Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1998. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford Digital Libraries Working Paper.

- Adam Pease. 2011. *Ontology - A Practical Guide*. Articulate Software Press.
- Maria Pershina, Yifan He, and Ralph Grishman. 2015. Personalized page rank for named entity disambiguation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. pages 238–243.
- Alessandro Raganato, Jose Camacho-Collados, and Roberto Navigli. 2017. Word sense disambiguation: A unified evaluation framework and empirical comparison. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. pages 99–110.
- Josef Ruppenhofer, Michael Ellsworth, Miriam R. L. Petruck, Christopher R. Johnson, Collin F. Baker, and Jan Scheffczyk. 2016. *FrameNet II: Extended Theory and Practice*.
- Mark Stevenson, Eneko Agirre, and Aitor Soroa. 2012. Exploiting domain information for Word Sense Disambiguation of medical documents. *JAMIA* 19(2):235–240.