Wordnet-based Evaluation of Large Distributional Models for Polish

Maciej Piasecki, Gabriela Czachor, Arkadiusz Janz, Dominik Kaszewski, Paweł Kędzia
G4.19 Research Group, Computational Intelligence Department
Wrocław University of Science and Technology, Wrocław, Poland
{maciej.piasecki,arkadiusz.janz}@pwr.edu.pl

Abstract

The paper presents construction of large scale test datasets for word embeddings on the basis of a very large wordnet. They were next applied for evaluation of word embedding models and used to assess and compare the usefulness of different word embeddings extracted from a very large corpus of Polish. We analysed also and compared several publicly available models described in literature. In addition, several large word embeddings models built on the basis of a very large Polish corpus are presented.

1 Introduction

Distributional Semantics (DS) is focused on describing semantic associations between words on the basis of their distributional patterns in texts by applying statistical methods. DS methods are used to extract different kinds of the *Measures of Semantic Relatedness* (MSR) from corpora . An MSR can cover the whole range of semantic relations from topic or domain based till lexico-semantic relations. For many applications it is desirable to obtain an MSR which is close to a *Measure of Semantic Similarity* (MSS), i.e. a measure which assigns the highest values to words associated by linguistic lexico-semantic relations. Recently, word embeddings have become one of the best tools of DS. However, word embeddings, e.g. (Mikolov et al., 2013), are based on predicting a word occurrence in a context (mostly a sequence) of other words. This aspect of co-occurrence prediction in a local context can influence an MSR built on the basis of word embeddings. An MSS can be an important source of knowledge supporting wordnet development, e.g. (Piasecki et al., 2009). However, the question is how to evaluate to which extent the given MSR resembles an proper MSS? Experiments with the participation of humans are laborious, costly and the datasets created as a result are of limited size. It is hard to construct an evaluation by application in a way revealing the properties of a potential MSS.

A large wordnet is built on knowledge originating from humans. It includes directly the knowledge about lexico-semantic relations and offers an opportunity to build large scale, realistic tests. Our goal is to construct large scale test datasets for word embeddings on the basis of a large wordnet, apply them for evaluation of word embedding models and next to analyse and compare the usefulness of different word embeddings extracted from a very large corpus of Polish. Finally, we want to publishing word embedding models of known properties built on the basis of a very large corpus of Polish.

2 Related Works

MSR evaluation methods can be roughly divided into intrinsic and extrinsic. The former are based on the direct evaluation of the MSR properties, e.g. by assessment by humans or comparison with a gold standard. The latter is based on applying an MSR as knowledge source in some NLP application.

Typical datasets used in the intrinsic evaluation are small, e.g. (Rubenstein and Goodenough, 1965), WS-353 (Finkelstein et al., 2002) and most of the all 10 data sets discussed in (Faruqui and Dyer, 2014), where only two of them include $\approx$2000 and $\approx$3000 word pairs. They were used in many tests, in fact overused. Small sizes of these datasets make performing proper evaluation more difficult, e.g. because of the lack of the common partitioning into training, tuning and testing parts.

Datasets for MSR evaluation are often collected during experiments based on testing human judgement in reaction to some prompting signal, which is close to reaction to a stimuli, e.g. (Auguste et al., 2017) measured the correlation between the reaction times in the context of priming with ranking based on word embeddings. However, this is slightly different situation than analysis of lexical meanings during language utterance interpretation, especially a textual utterance. MSR is extracted from a text corpus, and it is more natural to evaluate it against language resources. Moreover, (Faruqui et al., 2016) noticed that the distinction between similarity and relatedness is not well defined and consistently expressed in most popular test datasets.

(Schnabel et al., 2015) evaluated systematically different DS models, but finally all tests were based on data collected during crowdsourcing experiments using Amazon Turk. (Jastrzebski et al., 2017) performed “evaluation focused on data efficiency" with respect to 4 categories, namely: “Similarity, Analogy, Sentence and Single word”. In the case of similarity, which is most interesting for us, they used only well known data sets for English. For each type of dataset different combinations of preprocessing and classification algorithms were applied.

It is worth to notice, that the cost of preparing larger datasets for another language than English is quite substantial. This is one of the reasons that it is hard to find such approaches for other languages, with notable exceptions e.g. (Hartmann et al., 2017) for Portuguese. In our case we want to explore the possibility of constructing of large test datasets on the basis of an already existing wordnet. As the primary application we focused on is support for wordnet development, so comparison with data collected in experiments with humans is not necessarily the best solution for us.

3 Wordnet-based Evaluation

In many approaches a wordnet was used to generate a wordnet-based measure of semantic similarity that was next used to assess the correlation between it and an MSR, e.g. (Lin, 1998). It was assumed that similarity rankings generated by the two measures should be similar. However, there are many wordnet-based similarity measures of different properties and some of them depend on additional knowledge like information about the frequency of word senses. Thus, the result of the comparison can be different depending on the wordnet-based similarity measure applied and in all cases is not straightforward in interpretation. We want to follow a different approach and to explore two methods that are free of these problems.

3.1 Synonymy tests

(Freitag et al., 2005) proposed a wordnet-based synonymy test (WBST) in which for a *question word* x an n-tuple is automatically generated:
$\mathbf{D} = \langle d_1, \dots d_n \rangle$, such that one the elements: d_i is the correct *answer*, i.e. it is synonymous with x and belongs to the same synset as x, and all other $d_j \neq d_i$ are *detractors*, i.e. false answers that are not synonymous with x. Elements of $\mathbf{D}$ and the position of the correct answer are randomly selected. MSR is tested by using its values in selecting a possible answer for the problem word x.

In the case of some wordnets, including plWordNet, many synsets are singletons and include only one word. Thus they would be excluded from the test, and this could bias the evaluation result.

To prevent this, in *Hypernymy-expanded WBST* (HWBST) (Piasecki et al., 2009) answers for singleton synsets are selected from their hypernym synset, and in the same time these hypernyms are excluded from possible detractors. For a large wordnet, WBST and HWBST can include many thousands of ⟨question – answer⟩ pairs enabling very intensive testing of an MSR and partitioning the set in many different ways, e.g. test vs train, frequent vs infrequent or according to the domains of words.

Because detractors in WBST and HWBST are selected completely randomly, the majority of them come from those parts of the wordnet that are very remote in relation to the question word. Thus these types of tests are relatively easy to be solved on the basis of an MSR. In order to make the test harder we need to select detractors in such a way that words from synsets semantically similar to the question words have a higher probability of being selected than words from the synsets of small similarity. This version of the test is called *Extended WBST* (EWBST) (Piasecki et al., 2009). EWBST consists of pairs $\langle x_l, \mathbf{D}_l \rangle$, where x_l is a question word and $\mathbf{D}_l = \langle d_1, \dots d_n \rangle$ is a sequence of possible answers such that d_i is the correct answer, i.e. a synonym or hypernym of x_l, as in HWBST, while the rest of $d_j \in \mathbf{D}_l \wedge d_j \neq d_i$ are selected randomly from the whole wordnet but with the probability correlated to the *wordnet-based similarity measure* (WSM) between d_j and x_l. Any WSM can be used to generate EWBST, but in the experiments presented in this work, we use a simple measure (1) proposed in (Agirre and Edmonds, 2006) based on the normalised length of a shortest path in the wordnet graph. It can be computed without knowing the frequency of senses:

$$WSM(w_1, w_2) = -\log \frac{path(w_1, w_2)}{2D_m} \qquad (1)$$

In (1), w_1 and w_2 are words, $path(w_1, w_2)$ is the shortest path in the extended hypernymy graph between two synsets including, respectively: w_1 and w_2, and D_m is the mean depth of the extended hypernymy graph. While (Agirre and Edmonds, 2006) used normalized path distance, in the recent version of plWordNet many synsets are far away from the root. This effectively flattens the probability distribution to the point where it is no different than uniform random sampling as per HWBST. Using average depth D_a instead reflects better relations contained in plWordNet and promotes synsets closer to the question word. However, this modification results also in negative values of WSM, so they had to be capped off at 0:

$$WSM_a(w_1, w_2) = max\big(-\log \frac{path(w_1, w_2)}{2D_a}, 0 \big) \qquad (2)$$

This reduces probability of choosing a detractor with distance $2D_a$ or greater to 0, so the tests become more difficult due to the elimination of trivial detractors unrelated to the question. The idea of EWBST is to make detractors more similar to the correct answer and more difficult to be properly distinguished from the correct answer on the basis of MSR values.

The graph was built from hypernymy relations and type/instance relations. In addition, as plWordNet hypernymy is not a single-rooted structure, we added to the graph several SUMO concepts (Pease, 2011) as top level nodes on the basis of the mapping of plWordNet hypernymy root synsets onto SUMO concepts.

3.2 Cut-off rendering tests

WBST-family tests illustrate the ability of an MSR to distinguish between words whose senses are located in different parts of the wordnet graph, while EWBST gives also insights into the sensitivity to small local differences. However, WBST-family tests concentrate on synonymy and hypernymy, as these two relations are mostly used in selection of the correct answers and detractors. Nevertheless, from a good MSS we can also expect an ability to express other types of lexico-semantic relations. This can be measured with the help of a simple *Wordnet-based Cut-off Rendering* test (WBCR). In WBCR for each question word x a bag-of-words of words is generated in which they come from:

- the synset S_x of x
- and synsets S_i connected directly and also indirectly to S_x by selected wordnet relations.

S_x and S_i are indirectly connected, if there is a path in the graph of wordnet relations such that it consists of a proper sequence of wordnet relations. Depending on the type of relations allowed for direct and indirect connections, as well as the assumed patterns for the paths and their maximal length, we can define different types of bags-of-words. Next, the evaluated MSR is used to reconstruct the extracted bag-of-words:

1. for the problem word x a ranking list of the words most related to x on the basis of the MSR values is generated; such a list will be called the *k-nearest neighbours* list (henceforth k-NNL) of x.
2. for the assumed k, the top k words from the list are collected as a reconstructed bag-of-words,
3. the reconstructed bag-of-words for x is compared with the wordnet-based bag-of-words, and precision, recall and F-measure are calculated.

This simple test is meaningful only for large, comprehensive wordnets or wordnets describing well some selected domains. However, WBCR has very simple interpretation and can be easily tuned to different subsets or domains of words and senses.

4 Experiments

During experiments, we built several word embeddings models from the largest corpus of Polish available. Next we evaluated them in several tests based on plWordNet 3.1 (i.e. the most contemporary version) and compared with other word embedding models for Polish extracted from smaller corpora and published in the web.

4.1 Corpora and preprocessing

As a basis for the experiments we selected plWordNet 3.1 – a very large wordnet of Polish including ≈190,500 different words, described by ≈282,500 senses, more than 217,000 synsets and more than 750,000 relation links. plWordNet has been built by corpus-based wordnet developed method (Maziarz et al., 2013) and expresses very good coverage of words in large corpora (Maziarz et al., 2016).

We calculated our word embeddings model on the basis of plWordNet Corpus 10.0 (plWNC) of Polish, which includes more than 4 billion words[1]. It is also probably the largest corpus of Polish built in a controlled way and was used during the plWordNet development.

plWNC was used in the experiments in two versions of preprocessing:

plWNC-lem the corpus was first morphosyntactically tagged and lemmatised with the help of WCRFT2 tagger (Radziszewski, 2013; Radziszewski and Warzocha, 2014); strings: "lemma:grammatical class" were in the input to *word2vec* (Mikolov et al., 2013).

plWNC-multi in the morpho-syntactically tagged plWNC Proper Names and multiword expressions described in plWordNet 3.1 were merged to single tokens.

plWNC-multi was prepared with the help of *Liner2* tool (Marcińczuk et al., 2013) for recognition and classification of PNs. plWordNet 3.1 includes almost 60,000 Polish MWEs represented as lexical units and described by lexicalised morpho-syntactic constraints that allow for their efficient and accurate recognition in tagged texts (Kurc et al., 2012). We represent Proper Names (one and multiword, including many common words) and multiword expressions as single tokens in *plWNC-multi* in order to block the interpretation of their components as individual words. Components of PNs and MWEs can have very specific meanings (e.g. in non-compositional MWEs) that can influence the resulting word embeddings.

Corpora created from the Polish Wikipedia data alone (of ≈ 600M words) were used in two experiments reported in the literature. We evaluated these published word embedding models against our tests, too, see Sec. 5

4.2 Word embedding models tested

For the generation of word2vec models *Gensim* library was used (Řehůřek and Sojka, 2010). On the basis of the set of 6 parameters, we selected during pre-experiments 9 different types of models to be evaluated experimentally, i.e. the following combinations:

1. vector size: 100, 300 and 1000,
2. algorithm type: *Skip-gram*, *CBOW ns* (with negative subsampling) and *CBOW hs* (with hierarchical softmax).

[1]It consists of IPI PAN Corpus (Przepiórkowski, 2004), the first annotated corpus of Polish, National Corpus of Polish (Przepiórkowski et al., 2012), Polish Wikipedia (from 2016), *Rzeczpospolita* Corpus (Weiss, 2008) – corpus of electronic editions of a Polish newspaper from the years 1993-2003, supplemented with text acquired from the Web – only text with small percentage of words unknown to a very comprehensive morphological analyser Morfeusz 2.0 (Woliński, 2014) were included; duplicates were automatically eliminated from the merged corpus.

Vector size	Min freq.	Model	WBST	HWBST	EWBST
1000	1000	w2w-*plWNC-multi*-skipg-ns	**92.43**	89.00	**63.97**
		w2w-*plWNC-multi*-cbow-hs	91.54	**89.34**	63.21
		w2w-*plWNC-multi*-cbow-ns	91.68	89.31	62.99
	200	w2w-*plWNC-multi*-skipg-ns	92.52	89.80	**62.51**
		w2w-*plWNC-multi*-cbow-hs	**92.71**	90.11	60.94
		w2w-*plWNC-multi*-cbow-ns	92.58	**90.11**	60.97
	30	w2w-*plWNC-multi*-skipg-ns	90.43	88.84	**58.92**
		w2w-*plWNC-multi*-cbow-hs	**92.56**	90.05	57.35
		w2w-*plWNC-multi*-cbow-ns	92.51	**90.07**	57.30
300	1000	w2w-*plWNC-multi*-skipg-ns	**90.81**	88.24	**62.50**
		w2w-*plWNC-multi*-cbow-hs	90.32	88.12	61.00
		w2w-*plWNC-multi*-cbow-ns	90.70	**88.49**	62.13
	200	w2w-*plWNC-multi*-skipg-ns	**91.81**	89.36	**61.24**
		w2w-*plWNC-multi*-cbow-hs	91.46	89.29	59.45
		w2w-*plWNC-multi*-cbow-ns	91.11	**89.50**	60.76
	30	w2w-*plWNC-multi*-skipg-ns	90.99	89.43	**58.25**
		w2w-*plWNC-multi*-cbow-hs	**91.36**	89.41	55.97
		w2w-*plWNC-multi*-cbow-ns	91.35	**89.79**	57.50
100	1000	w2w-*plWNC-multi*-skipg-ns	**88.84**	86.01	**59.42**
		w2w-*plWNC-multi*-cbow-hs	87.71	86.14	58.26
		w2w-*plWNC-multi*-cbow-ns	88.14	**86.71**	59.34
	200	w2w-*plWNC-multi*-skipg-ns	**89.78**	87.53	**58.52**
		w2w-*plWNC-multi*-cbow-hs	88.97	87.33	56.75
		w2w-*plWNC-multi*-cbow-ns	89.05	**87.57**	58.12
	30	w2w-*plWNC-multi*-skipg-ns	**89.79**	**88.21**	**55.99**
		w2w-*plWNC-multi*-cbow-hs	89.44	87.62	53.52
		w2w-*plWNC-multi*-cbow-ns	89.63	88.13	55.27
	1000	pl-embeddings-cbow	71.63	69.36	43.71
		pl-embeddings-skip	76.30	74.54	47.16
		fastText.wiki.pl	**80.01**	**78.17**	**52.42**
	200	pl-embeddings-cbow	71.79	69.46	42.31
		pl-embeddings-skip	76.89	74.65	45.53
		fastText.wiki.pl	**80.11**	**79.16**	**51.40**
	30	pl-embeddings-cbow	71.49	70.35	41.85
		pl-embeddings-skip	77.41	75.69	45.28
		fastText.wiki.pl	**81.44**	**80.27**	**51.39**

Table 1: WBST-like tests generated from noun in plWordNet 3.1 and applied to word embedding models extracted from *plWNC-multi*.

Thus, we tested: Skip-gram 100, Skip-gram 300, Skip-gram 1000, CBOW ns 100, CBOW ns 300, CBOW ns 1000, CBOW hs 100, CBOW hs 300 and CBOW hs 1000. In all models the minimal frequency of tokens (i.e. tagged lemmas and/or PN and MWE tokens) was set to ≥ 8 (min_count=8). Pre-trained models are readily available[2]

(Rogalski and Szczepaniak, 2016) first preprocessed a text corpus based on the Polish Wikipedia[3] by changing the text to lower case, numbers were divided into separate digits, and some non-text elements were deleted. Next two word embedding models were constructed: CBOW and Skip-gram models with negative sampling and the vector size: 300. The extracted models are publicly available in the internet[4] and following the original names they will be called in the experiments, respectively: *pl-embeddings-cbow* and *pl-embeddings-skip*.

(Bojanowski et al., 2016) built Skip-gram models[5] using *fastText* technique with the vector size 300 for many languages on the basis of Wikipedia data. For the extraction of the models a novel method in which "each word is represented as a bag of character n-grams", cf (Bojanowski et al., 2016), was applied. It was designed for languages with richer inflection and was meant to better deal with a large number of word forms in such languages. Their model will be simply called *fastText.wiki.pl* in the experiments.

The Polish language has a very rich morphology, which is why we also decided to examine *fastText* models, but plWNC 10.0 corpus was used for training. All of our *fastText* models were trained with the Skip-gram architecture and the vector of size 300. We tested the Skip-gram 300 model with minimal word frequencies of 5, 20 and 50. These models will be named according to given schema *fastText.plWNC* in our experiments.

Another set of models was introduced in (Mykowiecka et al., 2017). For our experiments

[2] https://clarin-pl.eu/dspace/handle/11321/442

[3] https://pl.wikipedia.org

[4] http://publications.it.p.lodz.pl/2016/word_embeddings/

[5] https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md

Model	VS	Score	Model	VS	Score
w2w-*plWNC-lem*-cbow-hs	100	39.29	w2w-*plWNC-lem*-cbow-ns	300	55.61
w2w-*plWNC-multi*-cbow-hs	100	40.82	w2w-*plWNC-multi*-cbow-ns	300	**57.14**
w2w-*plWNC-lem*-cbow-hs	300	48.47	w2w-*plWNC-lem*-skipg	100	45.92
w2w-*plWNC-multi*-cbow-hs	300	48.98	w2w-*plWNC-multi*-skipg	100	48.98
w2w-*plWNC-lem*-cbow-ns	100	47.96	w2w-*plWNC-lem*-skipg	300	**60.20**
w2w-*plWNC-multi*-cbow-ns	100	47.96	w2w-*plWNC-multi*-skipg	300	59.18
ft-*plWNC-multi*-skipg-mC5	300	50.75	ft-*plWNC-lem*-skipg-mC5	300	50.75
ft-*plWNC-multi*-skipg-mC20	300	**53.30**	ft-*plWNC-lem*-skipg-mC20	300	54.23
ft-*plWNC-multi*-skipg-mC50	300	50.75	ft-*plWNC-lem*-skipg-mC50	300	**59.28**
ncp-lemmas-all-100-cbow-hs	100	48.72	*ncp*-forms-all-100-cbow-hs	100	28.18
ncp-lemmas-all-100-cbow-ns	100	46.67	*ncp*-forms-all-100-cbow-ns	100	35.00
ncp-lemmas-all-100-skipg-hs	100	44.10	*ncp*-forms-all-100-skipg-hs	100	34.55
ncp-lemmas-all-100-skipg-ns	100	44.10	*ncp*-forms-all-100-skipg-ns	100	39.55
ncp-lemmas-all-300-cbow-hs	300	55.38	*ncp*-forms-all-300-cbow-hs	300	35.91
ncp-lemmas-all-300-cbow-ns	300	**57.95**	*ncp*-forms-all-300-cbow-ns	300	43.18
ncp-lemmas-all-300-skipg-hs	300	56.92	*ncp*-forms-all-300-skipg-hs	300	43.64
ncp-lemmas-all-300-skipg-ns	300	54.36	*ncp*-forms-all-300-skipg-ns	300	**46.82**
ncp-lemmas-restricted-100-cbow-hs	100	49.74	*ncp*-forms-restricted-100-cbow-hs	100	32.27
ncp-lemmas-restricted-100-cbow-ns	100	47.69	*ncp*-forms-restricted-100-cbow-ns	100	39.55
ncp-lemmas-restricted-100-skipg-hs	100	43.59	*ncp*-forms-restricted-100-skipg-hs	100	36.82
ncp-lemmas-restricted-100-skipg-ns	100	45.13	*ncp*-forms-restricted-100-skipg-ns	100	40.00
ncp-lemmas-restricted-300-cbow-hs	300	52.82	*ncp*-forms-restricted-300-cbow-hs	300	40.00
ncp-lemmas-restricted-300-cbow-ns	300	**59.49**	*ncp*-forms-restricted-300-cbow-ns	300	43.64
ncp-lemmas-restricted-300-skipg-hs	300	54.87	*ncp*-forms-restricted-300-skipg-hs	300	42.73
ncp-lemmas-restricted-300-skipg-ns	300	54.87	*ncp*-forms-restricted-300-skipg-ns	300	**47.27**

Table 2: Analogy tests from (Mykowiecka et al., 2017) applied to the different word embeddings models, where k is 10, all results in (%).

we selected the models trained with Skip-gram and CBOW architectures and the vector size of 100 and 300. These pre-trained models were generated on National Corpus of Polish. Due to the anticipated problems with the morpho-syntactic tagging, (Mykowiecka et al., 2017) utilised two versions of the corpus: full, further called '*ncp*-lemmas' or '*ncp*-forms' and "restricted data sets [...] which only included tokens classified as nouns, adjectives, adverbs, verb forms, and abbreviations, which constitute 19 parts of speech (POS) out of the 34 foreseen in" NCP.

4.3 Tests

4.3.1 Wordnet-based Synonymy Tests

All three types of tests, namely: WBST, HWBST and EWBST were generated on the basis of the noun part of plWordNet 3.1 in three versions corresponding to the minimal frequency of words in plWNC 10.0: 30, 200 and 1000, i.e. in a given test all question, answer and detractor words had to express the predefined minimal frequency in the corpus. However, still the generated tests are very large e.g. EWBST(min. 1000) includes 19,996 question – answers pairs, HWBST (min. 30) includes 48,263 pairs, WSBT, and WBST(min. 1000) includes 9,100 pairs – the smallest set because singleton synsets are omitted. All tests are open and accessible[6].

4.3.2 Wordnet-based Cut-off Rendering tests

As in the case of the WBST-like tests, the cut-off tests were generated on the basis of nouns in plWordNet 3.1 and in three main versions with respect to the minimal frequency of nouns in plWNC 10.0: 30, 200 and 1000 (the numbers of bag of words are smaller than the number of pairs in WBST-like tests but similarly large).

The wordnet context of a problem word x, which was represented as a bag of words was defined in three different ways:

Cnt – all words linked to x by direct relation links, i.e. from synsets linked directly to the synset of x and also by direct lexical relations to one of the x senses; it also includes synonyms of x.

CntH – **Cnt** expanded with all indirect hyponyms and hypernyms of x up to the hypernymy and hyponymy paths of the maximal length 3.

CntHC – **CntH** expanded with all $k = m + n$ *cousins* of x with $k = 3$, i.e. words from synsets accessible from the synsets of x by hyper/hyponymy paths of up to m hypernymy and n hyponymy links.

Thus, **Cnt** measures the ability of an MSR to find words in very close relations (e.g. as a potential tool supporting description of x senses), **CntH** illustrates the use of the MSR as a tool supporting construction of hyper/hyponymy structures, and **CntHC** characterises, e.g., a possibility of using the given MSR for identifying small wordnet subgraphs which lemma senses belong to. All cut-off tests were applied to the *k-best neighbours* lists with $k \in \{10, 100\}$ generated for nouns from plWordNet.

[6] https://clarin-pl.eu/dspace/handle/11321/446

Cut-off Precision

	k NN	10			100		
Model	Min. f.	Cnt	CntH	CntHC	Cnt	CntH	CntHC
w2w-*plWNC-multi*-cbow-hs	1000	13.42	15.12	**35.67**	**3.31**	**4.29**	**17.04**
w2w-*plWNC-multi*-cbow-ns	1000	**13.62**	**15.16**	34.25	3.30	4.22	15.96
w2w-*plWNC-multi*-skipg	1000	12.35	13.47	28.07	2.66	3.18	10.12
ft-*plWNC-multi*-skipg	1000	8.74	9.24	15.72	2.59	3.00	8.14
w2w-*plWNC-lem*-cbow-hs	1000	12.86	14.26	33.38	3.11	3.93	15.75
w2w-*plWNC-lem*-cbow-ns	1000	9.65	10.58	25.40	2.17	2.60	9.71
w2w-*plWNC-lem*-skipg	1000	11.61	12.61	27.15	2.47	2.92	9.82
ft-*plWNC-lem*-skipg	1000	7.39	7.72	13.31	2.25	2.54	7.25
w2w-*plWNC-multi*-cbow-hs	200	**11.54**	**12.94**	**32.91**	**2.70**	**3.47**	**15.48**
w2w-*plWNC-multi*-cbow-ns	200	11.17	12.34	30.92	2.61	3.29	14.41
w2w-*plWNC-multi*-skipg	200	10.37	11.23	25.06	2.15	2.55	9.20
ft-*plWNC-multi*-skipg	200	8.42	8.84	16.09	2.21	2.54	7.95
w2w-*plWNC-lem*-cbow-hs	200	10.50	11.57	30.01	2.46	3.07	14.21
w2w-*plWNC-lem*-cbow-ns	200	8.20	8.94	23.26	1.79	2.12	9.08
w2w-*plWNC-lem*-skipg	200	9.64	10.42	24.03	1.99	2.33	8.84
ft-*plWNC-lem*-skipg	200	7.05	7.32	12.98	1.93	2.16	6.87

Cut-off Recall

	k NN	10			100		
		Cnt	CntH	CntHC	Cnt	CntH	CntHC
w2w-*plWNC-multi*-cbow-hs	1000	**10.33**	**7.10**	**3.42**	**20.83**	**15.69**	**8.61**
w2w-*plWNC-multi*-cbow-ns	1000	10.09	6.84	3.24	20.27	14.84	8.16
w2w-*plWNC-multi*-skipg	1000	9.24	6.26	2.91	17.22	12.20	6.26
ft-*plWNC-multi*-skipg	1000	7.33	4.87	2.18	17.54	12.22	5.80
w2w-*plWNC-lem*-cbow-hs	1000	8.74	6.05	2.85	17.67	13.03	7.03
w2w-*plWNC-lem*-cbow-ns	1000	6.71	4.61	2.18	13.20	9.46	4.99
w2w-*plWNC-lem*-skipg	1000	8.19	5.64	2.60	15.12	10.82	5.41
ft-*plWNC-lem*-skipg	1000	5.92	4.04	1.82	14.88	10.40	4.85
w2w-*plWNC-multi*-cbow-hs	200	**10.76**	**7.40**	**3.89**	**20.90**	**15.75**	**9.42**
w2w-*plWNC-multi*-cbow-ns	200	9.82	6.64	3.54	19.53	14.24	8.76
w2w-*plWNC-multi*-skipg	200	9.18	6.22	3.19	16.65	11.76	6.71
ft-*plWNC-multi*-skipg	200	8.56	5.70	2.84	18.40	12.81	6.92
w2w-*plWNC-lem*-cbow-hs	200	8.45	5.91	3.19	16.89	12.48	7.65
w2w-*plWNC-lem*-cbow-ns	200	6.86	4.78	2.60	13.20	9.52	5.69
w2w-*plWNC-lem*-skipg	200	8.04	5.59	2.91	14.53	10.44	5.93
ft-*plWNC-lem*-skipg	200	6.98	4.83	2.49	15.63	11.04	5.90

F measure

	k NN	10			100		
		Cnt	CntH	CntHC	Cnt	CntH	CntHC
w2w-*plWNC-multi*-cbow-hs	1000	**11.67**	**9.66**	**6.23**	**5.72**	**6.74**	**11.44**
w2w-*plWNC-multi*-cbow-ns	1000	11.59	9.42	5.92	5.68	6.57	10.80
w2w-*plWNC-multi*-skipg	1000	10.57	8.55	5.27	4.61	5.05	7.73
ft-*plWNC-multi*-skipg	1000	7.97	6.38	3.83	4.51	4.82	6.77
w2w-*plWNC-lem*-cbow-hs	1000	10.41	8.49	5.25	5.29	6.04	9.72
w2w-*plWNC-lem*-cbow-ns	1000	7.91	6.42	4.02	3.73	4.08	6.59
w2w-*plWNC-lem*-skipg	1000	9.60	7.80	4.75	4.24	4.60	6.98
ft-*plWNC-lem*-skipg	1000	6.57	5.30	3.20	3.90	4.09	5.81
w2w-*plWNC-multi*-cbow-hs	200	**11.13**	**9.42**	**6.96**	**4.78**	**5.68**	**11.71**
w2w-*plWNC-multi*-cbow-ns	200	10.45	8.63	6.35	4.60	5.35	10.89
w2w-*plWNC-multi*-skipg	200	9.74	8.01	5.66	3.81	4.19	7.76
ft-*plWNC-multi*-skipg	200	8.49	6.93	4.83	3.94	4.23	7.40
w2w-*plWNC-lem*-cbow-hs	200	9.37	7.82	5.77	4.30	4.93	9.95
w2w-*plWNC-lem*-cbow-ns	200	7.47	6.23	4.68	3.15	3.47	7.00
w2w-*plWNC-lem*-skipg	200	8.76	7.28	5.20	3.49	3.81	7.10
ft-*plWNC-lem*-skipg	200	7.02	5.82	4.17	3.43	3.62	6.35

Table 3: Wordnet-based Cut-off Rendering tests for nouns in plWordNet 3.0 applied to word embedding models extracted from *plWNC-multi* (vec. size=300), where kNN is the length of the k-NN lists, all results in (%).

Cut-off Precision

	k NN	10			100		
Model	Min. freq.	Cnt	CntH	CntHC	Cnt	CntH	CntHC
pl-embeddings-cbow	1000	**4.97**	**6.10**	**20.98**	1.37	**1.94**	**10.51**
pl-embeddings-skipg	1000	4.19	4.91	15.32	1.27	1.66	7.58
fastText.wiki.pl	1000	4.03	4.24	7.24	**1.52**	1.78	6.04
pl-embeddings-cbow	200	**3.92**	**4.81**	**17.77**	1.08	**1.52**	**8.78**
pl-embeddings-skipg	200	3.42	4.05	13.58	1.02	1.34	6.65
fastText.wiki.pl	200	3.90	4.07	7.33	**1.31**	1.51	5.76
pl-embeddings-cbow	30	3.28	**4.03**	**15.56**	0.90	1.27	**7.67**
pl-embeddings-skipg	30	2.99	3.55	12.56	0.88	1.16	6.14
fastText.wiki.pl	30	**3.72**	3.87	7.41	**1.15**	**1.33**	5.49

Cut-off Recall

	k NN	10			100		
		Cnt	CntH	CntHC	Cnt	CntH	CntHC
pl-embeddings-cbow	1000	3.07	**2.27**	**1.13**	7.44	6.02	**3.35**
pl-embeddings-skip	1000	2.79	1.98	0.96	7.24	5.57	2.91
fastText.wiki.pl	1000	**3.13**	2.12	0.96	**9.52**	**6.62**	3.12
pl-embeddings-cbow	200	2.79	2.08	1.12	6.81	5.55	3.29
pl-embeddings-skipg	200	2.68	1.95	1.02	6.90	5.43	3.04
fastText.wiki.pl	200	**3.74**	**2.55**	**1.26**	**9.86**	**6.89**	**3.62**
pl-embeddings-cbow	30	2.55	1.90	1.05	6.29	5.10	3.13
pl-embeddings-skipg	30	2.64	1.93	1.04	6.75	5.34	3.09
fastText.wiki.pl	30	**4.07**	**2.78**	**1.44**	**9.79**	**6.87**	**3.85**

F measure

	k NN	10			100		
		Cnt	CntH	CntHC	Cnt	CntH	CntHC
pl-embeddings-cbow	1000	**3.79**	**3.31**	**2.15**	2.32	**2.94**	**5.08**
pl-embeddings-skipg	1000	3.35	2.82	1.80	2.15	2.56	4.20
fastText.wiki.pl	1000	3.52	2.83	1.70	**2.63**	2.81	4.12
pl-embeddings-cbow	200	3.26	2.90	2.11	1.86	2.38	**4.79**
pl-embeddings-skipg	200	3.01	2.63	1.89	1.77	2.15	4.17
fastText.wiki.pl	200	**3.82**	**3.14**	**2.16**	**2.31**	**2.48**	4.45
pl-embeddings-cbow	30	2.87	2.58	1.97	1.58	2.03	4.44
pl-embeddings-skipg	30	2.80	2.50	1.93	1.55	1.90	4.11
fastText.wiki.pl	30	**3.89**	**3.24**	**2.41**	**2.06**	**2.22**	**4.53**

Table 4: Wordnet-based Cut-off Rendering tests for nouns in plWordNet 3.0 applied for word embedding models extracted from the Polish Wikipedia, where kNN is the length of the k-NN lists, all results in (%).

4.3.3 Analogy tests

One of the most popular techniques for word embeddings is to test their ability of reflecting word analogies, e.g. applied also in (Mykowiecka et al., 2017) for testing word embeddings for Polish. Analogy consists of 2 pairs of words, the relation between the first pair being similar to second pair. For example, we can say that the relation between *winter* and *snow* is analogous to *autumn* and *rain*, the relation being the typical weather in given season. Another common example is often used to showcase analogy is *man-woman*:*king-queen*, with the relation of male-female counterparts.

For word embeddings, the relation between words is simply the difference between their vectors and therefore the analogy can be written as $\vec{a} - \vec{b} = \vec{c} - \vec{d}$, where $\vec{a}$, $\vec{b}$, $\vec{c}$, $\vec{d}$ are embedding vectors for words.

For the purpose of testing, the above equation is transformed into $(\vec{b} + \vec{c}) - \vec{a} = \vec{d}$. The left hand side is evaluated by the means of vector arithmetic and k vectors most similar to the result are found. If one of the vectors is $\vec{d}$, then the model is said to pass the analogy test. If one of the words in the analogy is not present in the model then the single analogy is omitted as it cannot be evaluated. We used analogy tests from (Mykowiecka et al., 2017) with a kind permission and help of the authors. Only the semantic part of 196 test items was applied.

5 Results

The results of the tests in Tab. 1 illustrate well the differences in the difficulty of the tests: WBST is the simplest one, EWBST the hardest. The difference between EWBST and the two other tests is striking in all experiments. The difficulty of EWBST can be tuned by changing the wordnet-base similarity measure it is based on and the dependency between the similarity measure and the probability distribution of the selection of detractor words.

Skip-gram model is better than CBOW according to WBST and EWBST in most of the cases while in the other cases the differences are small. Only in HWBST CBOW-ns achieved higher result that can be attributed to a kind of generalisation introduced by the inclusion of hypernyms into correct answers. Also among the models from the literature, models based on Skip-gram

Cut-off Precision

	k NN	10			100		
Model	Min. freq.	Cnt	CntH	CntHC	Cnt	CntH	CntHC
ncp-lemmas-all-cbow-hs	1000	11.71	13.27	**32.67**	2.88	3.73	**15.51**
ncp-lemmas-all-cbow-ns	1000	**12.15**	**13.51**	31.57	**2.95**	**3.74**	14.61
ncp-lemmas-all-skipg-hs	1000	10.40	11.52	27.19	2.51	3.11	12.00
ncp-lemmas-all-skipg-ns	1000	10.00	10.92	23.15	2.17	2.57	8.42
ncp-lemmas-all-cbow-hs	200	9.56	**10.79**	**28.55**	2.28	**2.93**	**13.42**
ncp-lemmas-all-cbow-ns	200	**9.70**	10.72	27.38	**2.30**	2.88	12.72
ncp-lemmas-all-skipg-hs	200	8.67	9.60	24.67	2.03	2.51	10.85
ncp-lemmas-all-skipg-ns	200	7.98	8.69	19.66	1.70	1.99	7.25

Cut-off Recall

	k NN	10			100		
		Cnt	CntH	CntHC	Cnt	CntH	CntHC
ncp-lemmas-all-cbow-hs	1000	**8.16**	**5.81**	2.75	**16.73**	**12.88**	**6.81**
ncp-lemmas-all-cbow-ns	1000	8.10	5.59	**2.60**	16.66	12.36	6.54
ncp-lemmas-all-skipg-hs	1000	7.39	5.11	2.37	15.16	11.23	5.82
ncp-lemmas-all-skipg-ns	1000	7.26	5.02	2.30	13.79	9.88	4.90
ncp-lemmas-all-cbow-hs	200	**8.01**	**5.77**	**3.07**	**16.16**	**12.52**	**7.28**
ncp-lemmas-all-cbow-ns	200	7.64	5.34	2.83	15.62	11.62	6.94
ncp-lemmas-all-skipg-hs	200	7.45	5.25	2.76	15.04	11.30	6.50
ncp-lemmas-all-skipg-ns	200	6.73	4.71	2.41	12.71	9.17	5.11

F measure

	k NN	10			100		
		Cnt	CntH	CntHC	Cnt	CntH	CntHC
ncp-lemmas-all-cbow-hs	1000	9.62	**8.08**	**5.07**	**4.91**	**5.78**	**9.46**
ncp-lemmas-all-cbow-ns	1000	**9.72**	7.91	4.80	5.01	5.74	9.04
ncp-lemmas-all-skipg-hs	1000	8.64	7.08	4.36	4.30	4.88	7.84
ncp-lemmas-all-skipg-ns	1000	8.42	6.88	4.18	3.75	4.07	6.20
ncp-lemmas-all-cbow-hs	200	**8.71**	**7.52**	**5.54**	4.00	**4.75**	**9.44**
ncp-lemmas-all-cbow-ns	200	8.55	7.13	5.12	**4.01**	4.61	8.98
ncp-lemmas-all-skipg-hs	200	8.01	6.79	4.96	3.58	4.11	8.13
ncp-lemmas-all-skipg-ns	200	7.30	6.11	4.29	3.00	3.28	6.00

Table 5: Wordnet-based Cut-off Rendering tests for nouns in plWordNet 3.0 and applied for word embedding models from (Mykowiecka et al., 2017), where kNN is the length of the k-NN lists, all results in (%).

scheme, including *fastText.wiki.pl* (which is a Skip-gram model too) express higher results. This is especially visible in the case of the more difficult EWBST test. The *wiki.pl* was superior among the models built only on the data from Wikipedia, i.e. several times smaller than plWNC 10.0. However, all models built on much smaller corpus produced much worse results. We tested also models based on *plWNC-lem* version of the large corpus and all models were slightly but significantly worse in the WBST-family of tests.

Contrary to the synonymy tests, in the case of WBCR evaluations of the models generated from *plWNC-multi* presented in Tab. 3, we can notice that CBOW models are superior in all cases in comparison to Skip-gram models. It means that Skip-gram models are better in describing differences between word meanings, while CBOW enable broader exploration of potential lexico-semantic relations. However, relatively good precision signals that instances of lexico-semantic relations receive higher values. Definitely the results of the test are negatively biased by lacking relation instances in plWordNet. This kind of tests and evaluations can be used also as a diagnostic tool to spot these subdomains in a wordnet that are potentially not well enough described by relation links. We can observe also that the application of hierarchical softmax consistently produces better results in all frequency ranges. However, hierarchical softmax should result in better estimation of the representation.

We evaluated also word embedding models extracted from *plWNC-lem*, i.e. a version without folding PNs and MWEs into single tokens. WBCR tests showed lower performance due to the lack of MWE description. We also plan to apply tests not including MWEs in the future in order to investigate the effect of folding more precisely. Thus, models based on *plWNC-multi* offer a unique opportunity of obtaining good distributional description of PNs and MWEs.

Quite surprisingly, we can observe in Tab. 4 that models built on a smaller corpus of Wikipedia behave in a slightly different way in WBCR tests for less frequent words than those constructed on a very large corpus. In Tab. 4 Skip-gram models express higher recall, fastText Skip-gram with sub-word representation have much higher recall for words with lower frequency. However, this can be an effect of different preprocessing and filtering of the data. Nevertheless, all results obtained on the Polish Wikipedia are worse than those in Tab. 3 generated from a very large corpus (including also the Wikipedia data). It means that for WBCR tests, that cover a wider spectrum of relations, larger data used result in the improvement of the model.

Finally, in analogy tests, see Table 2, Skip-gram model built on the very large corpus *plWNC* is still the best one, like in EWBST, but the difference to models constructed on much smaller NCP is minimal. However, the analogy tests of (Mykowiecka et al., 2017) include mostly general and frequent words. Moreover, the differences are small only for models based on the restricted version of NCP, i.e. focused on the words included in the tests. Potential influence of the corpus preprocessing and filtering on distinguishing relations and lexical meanings is worth further investigation.

6 Conclusions

We showed that a large comprehensive wordnet can be successfully used as a basis for two different types of MSR evaluation methods, namely the family of Wordnet-based Synonymy Tests and Wordnet-based Cut-off Rendering tests. In both types of tests very large datasets can be generated allowing for very intensive testing and high statistical significance of the test results. The datasets are enough large to conveniently partitioned according to the frequency criteria of semantic criteria. In fact the datasets and tests are based on human decisions expressed in the wordnet structure. Both tests describe the ability of an MSR to be used as a basis for developing a lexico-semantic language resource.

WBST-family tests focus on the ability of an MSR to distinguish between different lexical meanings, while WBCR is sensitive more to representation of different types of wordnet relations by an MSR. As a result both types of tests are quite complementary. Moreover, by changing the similarity and context definitions in EWBST we can obtain tests of different difficulty.

In the further work, we develop a wordnet-based test that has properties of contextual tests, e.g. which is similar to Stanford contextual word similarity dataset (SCWS) (Huang et al., 2012).

We will also expand the presented evaluation to the dataset covering all four PoS, namely nouns, adjectives, verbs and adverbs.

The constructed word embedding models and evaluation datasets have been published on open licences under the link: `https://clarin-pl.eu/dspace/handle/11321/446`.

Acknowledgments

Co-financed by the Polish Ministry of Education and Science, CLARIN-PL Project.

References

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

Jeremy Auguste, Arnaud Rey, and Benoit Favre. 2017. Evaluation of word embeddings against cognitive processes: primed reaction times in lexical decision and naming tasks. In *Proceedings of the 2nd Workshop on Evaluating Vector-Space Representations for NLP*, pages 21–26, Copenhagen, Denmark, September. ACL.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Manaal Faruqui and Chris Dyer. 2014. Community evaluation and exchange of word vectors at `wordvectors.org`. In *Proc. of ACL: System Demo*.

Manaal Faruqui, Yulia Tsvetkov, Pushpendre Rastogi, and Chris Dyer. 2016. Problems with evaluation of word embeddings using word similarity tasks. In *Proceedings of the 1st Workshop on Evaluating Vector Space Representations for NLP*, pages 30–35, Berlin, Germany, August. ACL.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2002. Placing search in context: The concept revisited. *ACM Transactions on Informa- tion Systems*, 20(1).

Dayne Freitag, Matthias Blume, John Byrnes, Edmond Chow, Sadik Kapadia, Richard Rohwer, and Zhiqiang Wang. 2005. New experiments in distributional representations of synonymy. In *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005)*, pages 25–32, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Nathan Hartmann, Erick R. Fonseca, Christopher Shulby, Marcos Vinícius Treviso, Jessica Rodrigues, and Sandra M. Aluísio. 2017. Portuguese word embeddings: Evaluating on word analogies and natural language tasks. *CoRR*, abs/1708.06025.

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 ACL*.

Stanislaw Jastrzebski, Damian Lesniak, and Wojciech Marian Czarnecki. 2017. How to evaluate word embeddings? on importance of data efficiency and simple supervised tasks. *CoRR*, abs/1702.02170.

Roman Kurc, Maciej Piasecki, and Bartosz Broda. 2012. Constraint based description of polish multiword expressions. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 2408–2413, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *International Conference On Computational Linguistics (COLING'98). Proceedings of the 17th International Conference on Computational Linguistics*, volume 2, pages 768–774. ACL.

Michał Marcińczuk, Jan Kocoń, and Maciej Janicki. 2013. Liner2 – a customizable framework for proper names recognition for Polish. In Robert Bembenik, Lukasz Skonieczny, Henryk Rybinski, Marzena Kryszkiewicz, and Marek Niezgodka, editors, *Intelligent Tools for Building a Scientific Information Platform*, pages 231–253.

Marek Maziarz, Maciej Piasecki, Ewa Rudnicka, and Stan Szpakowicz. 2013. Beyond the transfer-and-merge wordnet construction: plWordNet and a comparison with WordNet. In Ruslan Mitkov, Galia Angelova, and Kalina Boncheva, editors, *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 443–452, Hissar, Bulgaria, September. INCOMA Ltd. Shoumen, BULGARIA.

Marek Maziarz, Maciej Piasecki, Ewa Rudnicka, Stan Szpakowicz, and Paweł Kędzia. 2016. plwordnet 3.0 – a comprehensive lexical-semantic resource. In N. Calzolari, Y. Matsumoto, and R. Prasad, editors, *Proc. of COLING 2016, 26th Inter. Conf. on Computational Linguistics*, pages 2259–2268. ACL.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

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

Adam Pease. 2011. *Ontology - A Practical Guide*. Articulate Software Press.

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

Adam Przepiórkowski. 2004. *The IPI PAN Corpus: Preliminary version*. Institute of Computer Science, Polish Academy of Sciences.

Adam Przepiórkowski, Mirosław Bańko, Rafał L. Górski, and Barbara Lewandowska-Tomaszczyk, editors. 2012. *Narodowy Korpus Języka Polskiego [Eng.: National Corpus of Polish]*. Wydawnictwo Naukowe PWN.

Adam Radziszewski and Radosław Warzocha. 2014. WCRFT2. CLARIN-PL digital repository, `http://hdl.handle.net/11321/36`.

Adam Radziszewski. 2013. A tiered CRF tagger for Polish. In R. Bembenik, Ł. Skonieczny, H. Rybiński, M. Kryszkiewicz, and M. Niezgódka, editors, *Intelligent Tools for Building a Scientific Information Platform: Advanced Architectures and Solutions*. Springer Verlag.

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`.

Marek Rogalski and Piotr S. Szczepaniak. 2016. Word embeddings for the polish language. In L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L.A. Zadeh, and J.M. Zurada, editors, *15th International Conference, ICAISC 2016, Zakopane, Poland, June 12-16, 2016, Proceedings*, volume 9692 of *LNAI, Artificial Intelligence and Soft Computing*, pages 126–135. Springer.

Herbert Rubenstein and John B. Goodenough. 1965. Contextual correlates of synonymy. *Communications of ACM*, 8(10).

Tobias Schnabel, Igor Labutov, David Mimno, and Thorsten Joachims. 2015. Evaluation methods for unsupervised word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 298–307, Lisbon, Portugal, September. ACL.

Dawid Weiss. 2008. Korpus Rzeczpospolitej [Corpus of text from the online edition of "Rzeczpospolita"]. `http://www.cs.put.poznan.pl/dweiss/rzeczpospolita`.

Marcin Woliński. 2014. Morfeusz reloaded. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014*, pages 1106–1111, Reykjavík, Iceland. ELRA.