Distributional Semantics Approach to Detecting Synonyms in Croatian Language

Mladen Karan, Jan Šnajder, Bojana Dalbelo Bašić

University of Zagreb
Faculty of Electrical Engineering and Computing
Text Analysis and Knowledge Engineering Lab
Unska 3, 10000 Zagreb, Croatia
{mladen.karan, jan.snajder, bojana.dalbelo}@fer.hr

Abstract

Identifying synonyms is important for many natural language processing and information retrieval applications. In this paper we address the task of automatically identifying synonyms in Croatian language using distributional semantic models (DSM). We build several DSMs using latent semantic analysis (LSA) and random indexing (RI) on the large hrWaC corpus. We evaluate the models on a dictionary-based similarity test – a set of synonymy questions generated automatically from a machine readable dictionary. Results indicate that LSA models outperform RI models on this task, with accuracy of 68.7%, 68.2%, and 61.6% on nouns, adjectives, and verbs, respectively. We analyze how word frequency and polysemy level affect the performance and discuss common causes of synonym misidentification.

Prepoznavanje hrvaških sopomenk s pomočjo distribucijske semantike

Prepoznavanje sopomenk je pomembno za številne aplikacije na področju jezikovnih tehnologij in poizvedovanja po informacijah. V pričujočem prispevku se ukvarjamo z avtomatskim prepoznavanjem sopomenk v hrvaščini, pri čemer uporabljamo modele distribucijske semantike (DSM). S pomočjo latentne semantične analize (LSA) in naključnega indeksiranja (RI) iz korpusa hrWaC zgradimo več različnih modelov. Modele nato ovrednotimo s pomočjo testov sinonimije, ki so avtomatsko izluščeni iz strojno berljivega slovarja. Rezultati kažejo, da so modeli, zgrajeni s pomočjo LSA, za to nalogo uspešnejši, njihova natančnost pa je 68,7% za samostalnike, 68,2% za pridevnike in 61,6% za glagole. V prispevku analiziramo tudi, kako pogostost pojavljanja besed v korpusu in stopnja njihove večpomenskosti vplivajo na rezultate in razpravljamo o najpogostejših razlogih za napake, do katerih pri prepoznavanju prihaja.

1. Introduction

The task of automatically determining the semantic similarity between words (e.g., *car* and *truck*) and semantic relatedness between words (e.g., *car* and *driver*) has received a lot of attention, and many semantic similarity measures (SSMs) have been proposed in the literature. Synonymy is a paradigmatic type of semantic relation between words that are substitutable in any possible context without change in meaning (*absolute synonyms*), substitutable in different contexts depending on their sense (*sense synonyms*), or substitutable in some contexts (*near-synonyms*). Numerous natural language processing and information retrieval applications can benefit from knowledge about synonyms, including word sense disambiguation (Agirre and Soroa, 2009), query expansion (Pantel et al., 2009), text similarity between short texts (Šarić et al., 2012), paraphrasing (Zhao et al., 2009), natural language generation (Inkpen and Hirst, 2004), WordNet acquisition (Broda et al., 2008), and text simplification (Inui et al., 2003).

Methods of semantic modelling can be divided into two broad categories: knowledge-based and distributional methods. The former rely on manually constructed resources, such as WordNet (Miller, 1995) or Wikipedia, to provide information required to measure relatedness. While such approaches work quite well, a resource like WordNet is often of limited coverage and, more importantly, it is not available for every language. The alternative to knowledge-based methods are distributional semantic models (DSMs). DSMs model the meaning of a word by the distribution of its contexts; two words are considered similar if they have similar context distributions. Often used DSMs are latent semantic analysis (LSA) (Deerwester et al., 1990) and random indexing (RI) (Sahlgren, 2005). More recently, approaches have been proposed that combine information from Wikipedia with distributional analysis (Gabrilovich and Markovitch, 2007; Hassan and Mihalcea, 2011). Unlike the knowledge-based approaches, DSMs require no linguistic resources other than a corpus (more sophisticated models, e.g., (Padó and Lapata, 2007), may also require tagging or parsing). This makes DSMs ideal for languages lacking basic lexical resources such as WordNet. On the other hand, the disadvantage of DSMs over their knowledge-based counterparts is that DSMs cannot easily distinguish between the various types of semantic relations (similarity vs. relatedness, paradigmatic vs. syntagmatic relations). Moreover, DSMs are typically token-based, and therefore unable to distinguish between the different senses of polysemous words. These two issues make synonymy detection using DSMs a very challenging task.

In this paper we address the task of identifying synonyms in Croatian language using DSMs. Our primary motivation is the automatic acquisition of WordNet synsets, as proposed by Broda et al. (2008). We perform large-scale experiments with two basic models (LSA and RI) built using the large Croatian Web as Corpus – hrWaC (Ljubešić and Erjavec, 2011). Inspired by the approach proposed by Landauer and Dumais (1997) and refined by Freitag et al. (2005), we evaluate our models on a dictionary-based similarity test (DBST) – a set of synonym questions generated automatically from a machine readable dictionary. To the best of our knowledge, this is the first work that addresses the task of synonymy detection for Croatian language.

The rest of the paper is structured as follows. Section 2 gives a summary of the related work. In Section 3 we describe the construction of DSMs. Evaluation results are presented in Section 4. We conclude in Section 5.

2. Related Work

The first to tackle the synonymy detection task using LSA were Landauer and Dumais (1997). To evaluate their approach, they used a set of synonymy matching questions, as they appear in the *Test of English as a Foreign Language* (TOEFL), a standardized test administered by the Educational Testing Service (ETS). Each synonym question consists of a target word and four answer words, of which one is a synonym of the question word, while the other three are distractors (incorrect answers). The task consists of identifying which of the four words is a synonym of the target word. Landauer and Dumais (1997) reported an accuracy of 64.4% on a set of 80 TOEFL questions. The best reported result on the set of TOEFL questions is 97.5% (Turney et al., 2003), achieved by a combination of methods.

Because work of Turney et al. (2003) essentially solved the synonymy detection task on TOEFL questions, Freitag et al. (2005) proposed a somewhat harder test – the *WordNet-based similarity test* (WBST). The test uses WordNet synsets to generate a large number of questions resembling TOEFL questions. Additional measures are taken to ensure that the distractors are not in a synonymy relation with the answer word. Because WBST has a lot more questions (23,570) than TOEFL has, WBST gives much more reliable performance estimates than TOEFL. The best reported result on WBST is 72.2% (Freitag et al., 2005), achieved using feature-based DSM. Features are unnormalized tokens, sometimes augmented with direction (left vs. right occurrence) and distance information, appearing within a context window of a target term. Freitag et al. (2005) also introduced a vector similarity measure specially tailored to the task of synonymy detection. A similar approach was used for Polish language (Broda et al., 2008; Piasecki et al., 2007). In contrast to Freitag et al. (2005), Broda et al. (2008) use a richer feature set incorporating lexical, morphological, and syntactic information. They also use feature selection methods and an additional feature weighting scheme to accentuate the most informative features of a particular target term. Since WBST may not be stringent enough to demonstrate the advantages of more sophisticated SSMs, Piasecki et al. (2007) proposed the *Extended WordNet based similarity test* (EWBST). This test extends the regular WBST by deliberately using similar and related (but not synonymous) words as distractors. Consequently, EWBST is much harder than WBST, but gives a better estimate of how well a SSM identifies synonyms.

3. Model Construction

3.1. Corpus and preprocessing

To build the DSMs, we use the large Croatian Web as a Corpus (hrWaC) (Ljubešić and Erjavec, 2011). To our knowledge, this is the largest available corpus of Croatian texts. In order to reduce the noise in the corpus, introduced by the use of informal language, we removed from the corpus all documents acquired from discussion forums and blogs. For reasons of computational efficiency, we also filtered out all words with a frequency below 50. This left us with a corpus containing 5,647,652 documents, 1.37 G tokens, 3.89 M word-form types, and 215,499 lemmas. Each document is further split into paragraphs (because the corpus was acquired from the web, the division into paragraphs is not consistent across all documents). To account for the morphological variation, which would disperse distribution vectors over inflectional forms and result in less reliable probability estimates, we employed lemmatization. To this end, we use the semi-automatically acquired morphological lexicon for Croatian language (Šnajder et al., 2008). We did not POS-tag the corpus; in cases of lemma ambiguity, we consider all possible lemmas when building DSMs. Moreover, we did not remove stop-words because all models have weighting schemes that give less emphasis to less discriminant words. Notice that we could have applied more sophisticated preprocessing techniqes, including POS tagging and parsing, but we leave this for future research.

3.2. Latent semantic analysis

Latent semantic analysis (LSA) (Deerwester et al., 1990) is a DSM based on the singular value decomposition (SVD) of a term-context co-occurrence matrix. A context vector for each of n contexts (documents or paragraphs) is extracted from the corpus. Elements of the context vector, corresponding to context c, are occurrence counts in c for each of the m target terms. The context vectors constitute the columns of a term-context co-occurrence matrix A. The row i of matrix A can be interpreted as a distribution of contexts conditioned on word i. Once the matrix A is constructed, it is decomposed by SVD, resulting in three matrices U, D, and V, such that $A = UDV^T$. The rows of U model for each term the distributions of a new set of contexts defined by DV^T. The final step is performing a dimensionality reduction by discarding all but the k largest singular values and the corresponding singular vectors. The semantic relatedness of words can be measured by comparing the corresponding rows of the reduced matrix U.

In our experiments, after constructing the term-context matrix, we apply the classical *tf-idf* weighting scheme. The inverse document frequency is defined as $\mathit{idf}(w) = \log \frac{D}{Q}$, where D is the total of number of documents in our corpus and Q is the number of documents containing word w.

The large size of the hrWaC corpus is reflected in the dimensions of our term-context co-occurrence matrices. The matrix has 215,499 rows (target terms); for document contexts the matrix has 5,647,652 columns and 827.7 M non-zero elements, while for paragraph contexts it had 29,763,686 columns and 1.16 G non-zero elements. For SVD computation we use the freely available ARPACK library.[1] For comparing the vectors, we use the cosine similarity measure.

[1] http://www.caam.rice.edu/software/ARPACK/

3.2.1. Random indexing

Random indexing (RI) (Sahlgren, 2005) is another kind of DSM that is, much like LSA, based on dimensionality reduction. For each context a random *index vector* is generated: a sparse d-dimensional vector containing a small number of randomly generated non-zero values. The so-obtained index vectors are shown to be nearly orthogonal (Sahlgren, 2005). Next, the distributional vectors for each target term are generated. Initially, all distributional vectors are d-dimensional null-vectors. The corpus text is scanned, and each time a term t is associated with context c, the index vector of c is added to the distributional vector of t. Eventually, target terms associated with similar contexts will tend to have similar distributional vectors. This is equivalent to constructing the entire term-context co-occurrence matrix and performing dimensionality reduction using random projection; the rows of the projection matrix are in fact the index vectors. The semantic relatedness of target terms can now be measured by comparing their distributional vectors.

In our models the index vectors are generated with dimension 100 (2 random non-zero elements) and 500 (4 random non-zero elements). The non-zero elements are chosen so that they contain an equal number of $+1$ and -1 values. We apply RI using documents, paragraphs, and neighboring words as contexts. In case of the first two, identically as for LSA, a term is associated with a context (a paragraph or a document) if it appears in the context. In the latter case, a term is associated with the context (the neighboring words) appearing within a ± 5 word window around the term. To take into account that some words are more informative than others, before adding the index vector of context word w, we weigh the whole index vector by the inverse document frequency score of w. Similarly as with LSA, we use the cosine similarity for vector comparison.

4. Evaluation

4.1. Dictionary-based similarity test

Because the Croatian WordNet (Raffaelli et al., 2008) is not yet available, we could not directly follow the approach by Freitag et al. (2005) to generate the similarity test. Instead, we relied on a machine readable dictionary derived from the monolingual Croatian dictionary (Anić, 2003). The dictionary lists over 68,500 lexemes divided into almost 100,000 sense entries. For each lexeme, the dictionary provides, inter alia, a basic morphological description and a gloss containing a short description of the word (or a description of every sense of a polysemous word). In many cases the gloss also contains implicit references to synonyms. In most cases these references follow a regular pattern and the referent can be extracted automatically. Using a few heuristic rules, we extracted automatically the synonym references from glosses and established synonymy links between entries. We extracted 43,311 synonym references (an average of 0.44 links per sense). Notice that the synonymy references are often ambiguous as they may refer to a polysemous word. Many ambiguous references could be resolved automatically, but this was not required in our case because we need not distinguish between senses of polysemious words when generating the questions. Interestingly, about 5000 words to which the synonymy references referred to were missing in the dictionary, thus we automatically added these entries to the dictionary.

Table 1: Example questions from the nouns part of DBST (the correct answers are A, D, C, and A)

težak *(farmer)*:		
A.	poljoprivrednik	*(farmer)*
B.	umjetnost	*(art)*
C.	radijacija	*(radiation)*
D.	bod	*(point)*
krov *(roof, home)*:		
A.	zgrada	*(building)*
B.	izvršilac	*(executant)*
C.	sanjkalište	*(sled run)*
D.	dom	*(home)*
karakter *(character)*:		
A.	detalj	*(detail)*
B.	kruška	*(pear)*
C.	lice	*(face, character)*
D.	maharadža	*(maharadja)*
jaran *(friend)*:		
A.	drug	*(friend)*
B.	krivovjerje	*(heresy)*
C.	sulfit	*(sulfite)*
D.	ekscentričnost	*(eccentricity)*

To generate the questions of our dictionary-based similarity test (DBST), we proceeded as follows. Using the synonymy links, we generate from the sense dictionary all pairs of synonymous words (the target word and the correct answer word), such that both words appear in our corpus. To make the test more realistic and more difficult, we use string-distance measures to filter out from this list pairs of words that seem to be morphologically or orthographically related. To generate the distractors for a question, we choose at random three words from the same part-of-speech, subject to the following constraints: (1) a chosen word appears in the corpus, (2) it is not in a (transitive) synonymy relation with any of the other four words used for that question, and (3) it is morphologically and orthographically unrelated to other four words. To check whether two words are in a synonymy relation, for each word we first collect 100 words to which it has transitive synonymy links, by performing a breadth-first search on the symmetric closure of the synonymy graph originating from the corresponding word (thereby disregarding the differences between senses). We then consider two words to be in a (transitive) synonymy relation if one word is contained in the set of 100 synonymy-linked words of the other word. Collecting 100 synonymy-linked words ensures that we have collected all potential synonyms of a given word and that therefore no distractor will be a synonym of any other distractor nor the target word. The requirement that the distractors are not in a synonymy relation makes the test more realistic, as synonymous distractors might be discarded from being the correct answers without consideration. The described procedure yields a set of 11,276 questions, of which 6446 for nouns (N), 2704 for adjectives (A), and 2126 for verbs (V). Example questions are given in Table 1 (for the sake of brevity some senses are omitted). Notice that, by using a dictionary-based evaluation, we subscribe to the definition of synonymy used in compiling the dictionary. The choice of a dictionary (characterized also by its coverage, sense granularity, etc.), together with the strategy used for generating the distractors, determines the appropriateness of a DBST as a means to evaluate DSMs for synonym identification.

DSMs in general tend to perform better for high-frequency words than for low-frequency words, as demonstrated by Piasecki et al. (2007). To test how word frequency affects the model performance, we generated two additional questions sets: one for low/medium-frequency band ($100 \leq f < 1000$) and one for high-frequency band ($f \geq 1000$). Both the target word and the answer words come from the corresponding frequency band, thus the questions contain more or less frequency-balanced words. We did not generate a separate low-frequency set because it would not have a sufficient number of questions.

Apart by frequency, we expect our models to be influenced by the level of polysemy. To test this assumption, we divide the questions based on their *polysemy levels*. Following Freitag et al. (2005), we define the polysemy level of a question to be the sum of the number of senses in the dictionary of its target and answer words.

4.2. Result analysis

Table 2 shows the accuracy of the models on the generated test sets. We evaluated 10 models: six RI models and four LSA models. The models were built with either 100 or 500 dimensions on contexts consisting of documents (D), paragraphs (P), or words within a window (W), as described in Section 3. Best results are given in bold. Notice that all models outperform the accuracy baseline of 25%.

A general observation is that the LSA models consistently outperform the RI models. The LSA500P model performed best in almost all experiments and outperformed the second-best model (LSA100P) by a significant margin. These results suggest that LSA may be better suited for the task of synonym detection in Croatian language. Results also reveal that a higher-dimensional model almost always significantly outperforms the corresponding lower-dimensional model. This indicates that the number of dimensions plays an important role in our task. The optimal number of dimensions for the task of identifying synonyms may differ when compared to other semantic similarity tasks (e.g., relatedness); to confirm this, additional experiments are required. The results seem to suggest that an additional increase in dimensionality may further improve the performance. We carried out additional experiments and concluded that this is not the case: improvement can only be observed until a plateau is reached at around 200 dimensions.

With respect to the context definition, results suggest that a smaller context – a window and especially a paragraph – gives better performance for LSA, while RI benefits more from a larger context – the entire document. While using a larger context is better for modelling long distance co-occurrences, using a smaller context prevents less relevant words occurring far from the target term from introducing noise into the distributions.

With respect to the word's part-of-speech, we can make two general observations: the performance on nouns and adjectives is comparable (slightly better on adjectives in most cases), while on verbs it is consistently lower. Identical behaviour can be observed for English in the results reported by Freitag et al. (2005).

As regards the frequency bands, results suggest that lower frequency has a detrimental effect on the performance of most models. This is expected, because distributional vectors of high-frequency words are built using more data, allowing for better modelling of word meaning. Notice, however, that high frequency words also tend to be more polysemous, which may again decrease the performance. In our case, however, it seems that higher word frequency still results in better performance. The same was confirmed in (Piasecki et al., 2007; Broda et al., 2008). Notice that in a realistic scenario the target and the answer words will not come from the same frequency band. In this respect, mixed frequencies results give a more realistic performance estimate.

In Table 3 we give the results for the best-performing LSA500P model with respect to polysemy levels of questions. As expected, models perform worse on questions with higher polysemy levels. Distributional representations of each sense of a polysemous word get merged into a single distributional representation – a mixture of distributions. For questions with high polysemy level, the corresponding distributional vectors are blurred and the similarity comparisons between such vectors are less meaningful.

Because the synonym questions contain randomly chosen distractors, the accuracy can vary on different test instances. To measure the variance in accuracy, we generated 30 test instances and calculated the performance of the best-performing model (LSA500P) across all test instances. Results proved to be quite stable: the maximum standard deviation of accuracy was 0.8% (obtained on verbs). This suggests that, owing to the relatively large number of questions, our DBST provides reliable accuracy estimates.

4.3. Error analysis

Most cases of synonym misidentification can be attributed to polysemy or low frequency of the target term in the corpus. We identified four typical causes of errors.

1. Homonyms and homographs – As Croatian is a highly inflected language, it is often the case that two different (often completely unrelated) words share some word-forms. For each ambiguous word-form we considered all its possible lemmas, thereby introducing an interference between the corresponding distributional vectors. The distributional vector of term t will model not only contexts of t, but also a number of additional contexts introduced by words that share word-forms with t. Even if the number of shared word-forms is small, the interference can still be very detrimental if the frequency difference between the words is large (i.e., one very frequent word-form may distort the distributional vector of a less frequent target term). A

Table 2: Accuracy for all considered models

	Mixed freq.			Low/medium freq.			High freq.		
Model	A	N	V	A	N	V	A	N	V
RI100D	42.0	41.8	36.2	39.7	29.5	29.3	43.9	46.5	37.7
RI500D	58.1	53.2	47.2	22.4	41.8	25.2	57.4	57.0	48.7
RI100P	40.7	39.5	36.3	24.1	31.9	29.3	43.6	42.5	37.8
RI500P	54.9	51.7	43.0	44.8	40.9	29.3	56.4	56.0	45.8
RI100W	56.8	49.8	43.5	44.8	42.4	35.0	52.5	48.8	40.3
RI500W	54.4	48.9	43.2	36.2	43.0	43.1	51.5	48.1	39.2
LSA100D	54.1	55.2	43.6	50.0	51.3	43.1	57.7	58.9	44.7
LSA500D	61.2	59.1	50.4	51.7	53.0	45.5	63.5	64.2	51.3
LSA100P	63.2	66.0	55.6	60.3	**67.7**	60.2	64.8	67.0	57.7
LSA500P	**68.2**	**68.7**	**61.6**	**67.2**	67.4	**65.0**	**69.3**	**70.1**	**60.3**

Table 3: Accuracy for model LSA500P with respect to different polysemy levels

	Mixed freq.			Low/medium freq.			High freq.		
Polysemy level	A	N	V	A	N	V	A	N	V
5–7	72.6	74.3	67.7	66.7	69.7	76.5	75.9	81.2	66.0
8–10	66.8	70.9	62.9	62.5	66.9	62.1	69.2	74.3	68.8
11–13	60.2	64.2	64.6	100.0	54.7	55.2	60.9	67.8	57.3
14–16	63.6	61.1	56.4	–	33.3	100.0	57.8	63.9	61.8
17–	47.6	60.0	58.4	–	–	–	46.5	64.0	52.9

case in point is the first question from Table 1: the word *težak* is a homonym, which in fact is more often used as an adjective (*hard, heavy*) than a noun. Obviously, this problem could for the most part be solved by a morphological disambiguation of the corpus.

2. Semantically related distractors – In some questions one of the distractors is a word that, albeit not synonymous, is semantically related to the target word. A case in point is the second question from Table 1: while the correct answer *dom (home)* does receive the second-best similarity score, the highest score goes to *zgrada (building)*. This is because the word *krov* in its dominant sense of *roof* happens to be more related to *zgrada* than to *dom*. While the polysemy of *krov* certainly contributes to misidentification, the dominant cause of misidentification is the fact that the semantically related word *zgrada* was among the distractors. Notice that, depending on the application, false synonyms that are paradigmatically related to the target word may still be usable in practice. However, manual inspection revealed that most false synonyms are syntagmatically related to the target word. To avoid this kind of error, we would need a method to distinguish between synonymy and general semantic relatedness.

3. Rare senses – In some cases the target word and the correct answer are sense synonyms via a very specific and seldom used sense. Contexts of such senses make a very small fraction of the total contexts on which the distributional vectors are built. Consequently, such senses are poorly modelled and the performance for them is worse. A case in point is the third question from Table 1: the correct answer *lice (face, character)* is almost never used in the sense of *character*, except in a few phrases.

4. Rare variants – In some cases the target word and the correct answer are not only sense synonyms, but also *variants*, i.e., they differ in register, dialect, or affect. This is often accompanied by a big difference in frequency; e.g., a dialectal form occurs very rarely in the corpus. Because the meanings of rare words are poorly modelled, this may lead to misidentification. An example is the fourth question from Table 1: the word *jaran (friend)* is an informal dialectal word used much less frequently than its sense synonym *drug (friend)*.

5. Conclusion and Future Work

In this paper we have addressed the task of automatically identifying synonyms in Croatian language using distributional semantic models (DSMs). We build several DSMs using latent semantic analysis (LSA) and random indexing (RI) on the large hrWaC corpus and evaluated the models on a dictionary-based similarity test. Results indicate that LSA models outperform RI models on this task. The best accuracy was obtained using LSA (500 dimensions, paragraph context): 68.7%, 68.2%, and 61.6% on nouns, adjectives, and verbs, respectively. Our results are along the lines of those obtained for English by Freitag et al. (2005). Compared to the result for Polish (Piasecki et al., 2007; Broda et al., 2008), our result is slightly worse, however, we make at present no use of rich morphological and syntactic features.

For future work we intend to address the most common causes of synonym misidentification discussed above. Following the work of Piasecki et al. (2007), we plan to develop a more stringent version of the similarity test. Another possibility for future research is to experiment with ways to mitigate the negative effect of polysemy by employing WSD techniques prior to building the distributional vectors, as done by Fišer et al. (2012). Finally, it would be interesting to experiment with many other types of distributional semantic models, such as the grammatical feature models (Freitag et al., 2005; Piasecki et al., 2007), the syntax-based model (Padó and Lapata, 2007), or Wikipedia-based models (Hassan and Mihalcea, 2011; Gabrilovich and Markovitch, 2007).

6. Acknowledgments

We thank the anonymous reviewers for their insightful comments and suggestions. This work has been supported by the Ministry of Science, Education and Sports, Republic of Croatia under Grant 036-1300646-1986.

7. References

E. Agirre and A. Soroa. 2009. Personalizing pagerank for word sense disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 33–41.

V. Anić. 2003. *Veliki rječnik hrvatskoga jezika*. Novi Liber.

B. Broda, M. Derwojedowa, M. Piasecki, and S. Szpakowicz. 2008. Corpus-based semantic relatedness for the construction of polish WordNet. In *Proceedings of the 6th Language Resources and Evaluation Conference (LREC'08)*.

S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6).

D. Fišer, N. Ljubešić, and O. Kubelka. 2012. Addressing polysemy in bilingual lexicon extraction from comparable corpora.

D. Freitag, M. Blume, J. Byrnes, E. Chow, S. Kapadia, R. Rohwer, and Z. Wang. 2005. New experiments in distributional representations of synonymy. In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, pages 25–32.

E. Gabrilovich and S. Markovitch. 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the 20th international joint conference on artificial intelligence*, volume 6, page 12.

S. Hassan and R. Mihalcea. 2011. Semantic relatedness using salient semantic analysis. In *Proceedings of AAAI Conference on Artificial Intelligence*.

D.Z. Inkpen and G. Hirst. 2004. Near-synonym choice in natural language generation. In *Recent Advances in Natural Language Processing*, volume 3, pages 141–152.

K. Inui, A. Fujita, T. Takahashi, R. Iida, and T. Iwakura. 2003. Text simplification for reading assistance: a project note. In *Proceedings of the second international workshop on Paraphrasing-Volume 16*, pages 9–16.

T.K. Landauer and S.T. Dumais. 1997. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review; Psychological Review*, 104(2):211.

N. Ljubešić and T. Erjavec. 2011. hrWaC and slWaC: compiling web corpora for Croatian and Slovene. In *Text, Speech and Dialogue*, pages 395–402. Springer.

G.A. Miller. 1995. WordNet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

J. Šnajder, B. Dalbelo Bašić, and M. Tadić. 2008. Automatic acquisition of inflectional lexica for morphological normalisation. *Information Processing & Management*, 44(5).

S. Padó and M. Lapata. 2007. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199.

P. Pantel, E. Crestan, A. Borkovsky, A.M. Popescu, and V. Vyas. 2009. Web-scale distributional similarity and entity set expansion. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pages 938–947.

M. Piasecki, S. Szpakowicz, and B. Broda. 2007. Extended similarity test for the evaluation of semantic similarity functions. *Vetulani (Vetulani, 2007)*, pages 104–108.

I. Raffaelli, M. Tadić, B. Bekavac, and Ž. Agić. 2008. Building croatian wordnet. In *Proceedings of the 4th Global WordNet Conference, Szeged, Global WordNet Association*, pages 349–359.

M. Sahlgren. 2005. An introduction to random indexing. In *Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering, TKE*, volume 5.

F. Šarić, G. Glavaš, M. Karan, J. Šnajder, and B. Dalbelo Bašić. 2012. Takelab: Systems for measuring semantic text similarity.

P. Turney, M.L. Littman, J. Bigham, V. Shnayder, et al. 2003. Combining independent modules to solve multiple-choice synonym and analogy problems.

S. Zhao, X. Lan, T. Liu, and S. Li. 2009. Application-driven statistical paraphrase generation. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 834–842.