An Open-source Framework for
Multi-level Semantic Similarity Measurement

Mohammad Taher Pilehvar and **Roberto Navigli**
Department of Computer Science
Sapienza University of Rome
{pilehvar,navigli}@di.uniroma1.it

Abstract

We present an open source, freely available Java implementation of *Align, Disambiguate, and Walk* (ADW), a state-of-the-art approach for measuring semantic similarity based on the Personalized PageRank algorithm. A pair of linguistic items, such as phrases or sentences, are first disambiguated using an alignment-based disambiguation technique and then modeled using random walks on the WordNet graph. ADW provides three main advantages: (1) it is applicable to all types of linguistic items, from word senses to texts; (2) it is all-in-one, i.e., it does not need any additional resource, training or tuning; and (3) it has proven to be highly reliable at different lexical levels and multiple evaluation benchmarks. We are releasing the source code at https://github.com/pilehvar/adw/. We also provide at http://lcl.uniroma1.it/adw/ a Web interface and a Java API that can be seamlessly integrated into other NLP systems requiring semantic similarity measurement.

1 Introduction

Semantic similarity quantifies the extent of shared semantics between two linguistics items, e.g., between *deer* and *moose* or *cat* and *a feline mammal*. Lying at the core of many Natural Language Processing systems, semantic similarity measurement plays an important role in their overall performance and effectiveness. Example applications of semantic similarity include Information Retrieval (Hliaoutakis et al., 2006), Word Sense Disambiguation (Patwardhan et al., 2003), paraphrase recognition (Glickman and Dagan, 2003), lexical substitution (McCarthy and Navigli, 2009) or simplification (Biran et al., 2011), machine translation evaluation (Lavie and Denkowski, 2009), tweet search (Sriram et al., 2010), question answering (Mohler et al., 2011), and lexical resource alignment (Pilehvar and Navigli, 2014).

Owing to its crucial importance a large body of research has been dedicated to semantic similarity. This has resulted in a diversity of similarity measures, ranging from corpus-based methods that leverage the statistics obtained from massive corpora, to knowledge-based techniques that exploit the knowledge encoded in various semantic networks. Align, Disambiguate, and Walk (ADW) is a knowledge-based semantic similarity approach which was originally proposed by Pilehvar et al. (2013). The measure is based on the Personalized PageRank (PPR) algorithm (Haveliwala et al., 2002) applied on the WordNet graph (Miller et al., 1990), and can be used to compute the similarity between arbitrary linguistic items, all the way from word senses to texts. Pilehvar et al. (2013) reported state-of-the-art performance on multiple evaluation benchmarks belonging to different lexical levels: senses, words, and sentences.

In this demonstration we present an open-source implementation of our system together with a Java API and a Web interface for online measurement of semantic similarity. We also introduce a method for offline calculation of the PPR stationary distribution for multiple starting nodes. Moreover, we release the compressed semantic signatures for all the 118K synsets and 155K words of WordNet 3.0.

2 Align, Disambiguate, and Walk (ADW)

ADW uses a two-phase procedure to model a given pair of linguistic items:

1. The pair is first disambiguated using an alignment-based disambiguation technique. Let a and b be two linguistic items to be compared, and S_w be the set of senses of a word w in the item a which is to be disambiguated. The alignment-based disambiguation measures the semantic similarity of each sense in S_w to all the senses of all the words in the compared item, i.e., b. The sense of w that produces the maximal similarity is taken as its intended sense. The procedure is repeated for all the other words in a and also in the opposite direction for all the words in b.

2. By using the PPR algorithm on the WordNet network, the two disambiguated items are modeled as high-dimensional vectors, called semantic signatures. To this end, ADW initializes the PPR algorithm from all the nodes in the semantic network that correspond to the disambiguated senses of the linguistic item being modeled. The resulting stationary distribution, which has WordNet synsets as its individual dimensions, is taken as the semantic signature of that item.

Finally, the similarity of the two linguistic items is computed as the similarity of their corresponding semantic signatures. We describe in Section 2.2 the four different signature comparison techniques that are implemented and offered in the package. Note that the two phases of ADW are inter-connected, as the alignment-based disambiguation in the first phase requires the generation of the semantic signatures for individual senses of each word in an item, i.e., the second phase.

2.1 Pre-computed semantic signatures

For each measurement of the semantic similarity between two linguistic items, ADW requires the semantic signatures for the two items to be calculated. Moreover, the alignment-based disambiguation of a pair of textual items requires the computation of all the semantic signatures of all their content words.

Therefore, a comparison of two items which contain an average of n words involves around $n \times p$ times the calculation of the PPR, where p is the average polysemy of the n words. This can be time-consuming and computationally expensive, particularly for larger textual items such as paragraphs. In order to speed up ADW we pre-computed the semantic signatures for individual WordNet synsets and words. We also provide a procedure for offline computation of semantic signatures for textual items comprising of multiple words, i.e., corresponding to multiple WordNet synsets, boosting the speed of signature generation for these items.

The WordNet graph is constructed by including all types of WordNet relations, and further enriched by means of relations obtained from Princeton Annotated Gloss Corpus[1]. The graph consists of 117,522 nodes (WordNet synsets) which are connected by means of more than half a million non-directed edges.

Individual synsets. We used the UKB package[2] to generate the semantic signatures for all the 118K synsets in WordNet 3.0. Each signature is truncated to the top 5000 most significant dimensions and compressed for better space utilization.

Words. We also generated semantic signatures for around 155K WordNet 3.0 words. To this end, for each word we initialized the PPR algorithm from all the synsets that contained its different senses. The word signatures can be used for faster computation of similarity, if it is not intended to perform alignment-based disambiguation on the items.

Other textual items. ADW computes the semantic signature of a textual item by initializing the PPR algorithm from all the nodes associated with its disambiguated content words. Given that it is simply unfeasible to pre-compute semantic signatures for all possible linguistic items, we put forward an approach which, given the pre-computed signatures for all WordNet synsets, can generate the semantic signature for an arbitrary linguistic item without the need to resort to the PPR algorithm. Let S be the set of synsets s corresponding to all the disambiguated

---

[1]http://wordnet.princeton.edu/glosstag.shtml
[2]http://ixa2.si.ehu.es/ukb/

```
//the two linguistic items to be compared
String t1 = "fire#v#4";
ItemType t1Type = ItemType.WORD_SENSE;

String t2 = "terminating the employment of a worker";
ItemType t2Type = ItemType.SURFACE;

//method for comparing semantic signatures
SignatureComparison compMethod = new WeightedOverlap();

double similarity = ADW.getInstance().getPairSimilarity(t1, t2,
        DisambiguationMethod.ALIGNMENT_BASED, compMethod, t1Type, t2Type);

System.out.println(similarity);
```

Figure 1: Sample ADW API usage for similarity measurement between a word sense and a phrase.

content words of a given linguistic item T. Considering each normalized semantic signature as a multinomial distribution, the semantic signature of the item T can be alternatively computed as the mean multinomial distribution of the signatures for individual synsets $s \in S$. It can be shown mathematically that the resulting mean distribution is equal to the same stationary distribution obtained by initializing the PPR algorithm from all the nodes corresponding to synsets $s \in S$.

2.2 Signature comparison

Four different methods are included in the package for comparing pairs of semantic signatures: Jensen-Shannon and Kullback-Leibler divergence, cosine, and Weighted Overlap (Pilehvar et al., 2013). Weighted Overlap is a rank similarity measure that computes the similarity of a pair of ranked lists in a harmonic manner, attributing more importance to the top elements than to the bottom ones. Pilehvar et al. (2013) reported improvements over the conventional cosine measure when using Weighted Overlap in multiple tasks and frameworks.

3 Availability

The Java source code can be obtained from ADW's github repository at https://github.com/pilehvar/adw/. We also provide a Java API, an online demo and the set of pre-computed semantic signatures for all the synsets and words in WordNet 3.0 at http://lcl. uniroma1.it/adw/.

4 Using ADW

Figure 1 shows a sample usage of the ADW API. The `getPairSimilarity` method in the `ADW` class receives six parameters: the two linguistic items, the disambiguation method (`ALIGNMENT_BASED` or `NONE`), the signature comparison method, and the types of the two inputs. ADW supports five different types of input:[3]

- **SURFACE**: Raw text (e.g., *A baby plays with a dog*).

- **SURFACE_TAGGED**: Lemmas with part of speech tags (e.g., *baby#n play#v dog#n*). We support only the four open-class parts of speech: nouns (*n*), verbs (*v*), adjectives (*a*), and adverbs (*r*).

- **SENSE_KEYS**: WordNet 3.0 sense keys (e.g., *baby%1:18:00:: play%2:33:00:: dog%1:05:00::*).

- **SENSE_OFFSETS**: WordNet 3.0 synset offsets (e.g., *09827683-n 01072949-v 02084071-n*).

- **WORD_SENSE**: Word senses in the form of lemma[#.]tag[#.]sense_number (e.g., *baby#n#1 play#v#1 dog#n#1* or *baby.n.1 play.v.1 dog.n.1*).

Figure 2 provides a snapshot of ADW's online demo. Two items from two different linguistic levels are being compared: the fourth sense of the verb *fire*[4] and the phrase "terminating the employment of a worker." The user can either choose the input type for each item from the drop-down menu or leave it to be automatically detected by the interface (the "detect automatically" option). The online demo also

---

[3]All word senses, sense keys and offsets are defined according to WordNet 3.0.

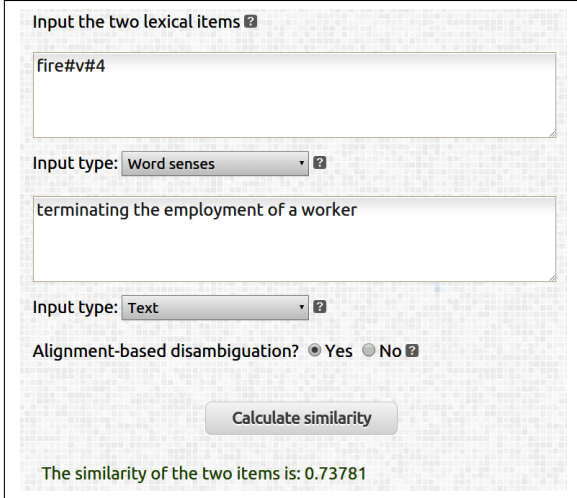[4]Defined as "terminate the employment of; discharge from an office or position."

Figure 2: A screenshot of ADW Web interface.

Dataset		Full vector		Truncated (top 5000)	
		Cosine	WO	Cosine	WO
RG-65	r	0.65	0.81	0.65	0.80
	ρ	0.82	0.86	0.82	0.85
TOEFL	%	96.3	95.0	96.3	95.0

Table 1: Performance of ADW on two different word similarity datasets, i.e., RG-65 (according to Spearman ρ and Pearson r correlations) and TOEFL (accuracy percentage), for two different vector comparison methods, i.e., cosine and Weighted Overlap (WO). We show results for two sets of vectors: full vectors with 118K dimensions and truncated vectors of size 5000 which are provided as a part of the package.

provides users with the possibility to test similarity measurement with no involvement of the disambiguation step.

5   Evaluation

We assessed the implementation of ADW on two evaluation benchmarks: similarity judgement correlation on the RG-65 dataset (Rubenstein and Goodenough, 1965) and synonym recognition on the TOEFL dataset (Landauer and Dumais, 1997). Given a set of word pairs, the task in judgement correlation is to automatically compute the similarity between each pair and judgements are ideally expected to be as close as possible to those assigned by humans. The closeness is usually measured in terms of correlation statistics. In the synonym recognition task, a target word is paired with a set of candidate words from which the most semantically similar word (to the target word) is to be selected.

Table 1 shows the results according to the Spearman ρ and Pearson r correlations on RG-65 and accuracy, i.e., the number of correctly identified synonyms, on TOEFL. We show results for two sets of vectors: full vectors of size 118K and truncated vectors of size 5000 which are provided as a part of the package. As can be seen, despite reducing the space requirement by more than 15 times, our compressed vectors obtain high performance on both the datasets, matching those of the full vectors on the TOEFL dataset and also the cosine measure.

6   Related Work

As the de facto standard lexical database, Word-Net has been used widely in measuring semantic similarity. Budanitsky and Hirst (2006) provide an overview of WordNet-based similarity measures. WordNet::Similarity, a software developed by Pedersen et al. (2004), provides a Perl implementation of a number of these WordNet-based measures. UMLS::Similarity is an adaptation of Word-Net::Similarity to the Unified Medical Language System (UMLS) which can be used for measuring the similarity and relatedness of terms in the biomedical domain (McInnes et al., 2009). Most of these WordNet-based measures suffer from two major drawbacks: (1) they usually exploit only the subsumption relations in WordNet; and (2) they are limited to measuring the semantic similarity of pairs of synsets with the same part of speech. ADW improves both issues by obtaining rich and unified representations for individual synsets, enabling effective comparison of arbitrary word senses or concepts, irrespective of their part of speech.

Distributional semantic similarity measures have also attracted a considerable amount of research attention. The S-Space Package (Jurgens and Stevens, 2010) is an evaluation benchmark and a development framework for word space algorithms, such as Latent Semantic Analysis (Landauer and Dumais, 1997). The package is integrated in DKProSimilarity (Bär et al., 2013), a more recently developed package geared towards semantic similarity of

textual items. DKProSimilarity provides an open-source implementation of several semantic similarity techniques, from simple string-based measures such as character *n*-gram overlap, to more sophisticated vector-based measures such as Explicit Semantic Analysis (Gabrilovich and Markovitch, 2007). ADW was shown to improve the performance of DKProSimilarity (Pilehvar et al., 2013) on the task of semantic textual similarity (Agirre et al., 2012).

Acknowledgments

References

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. SemEval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of SemEval-2012*, pages 385–393, Montreal, Canada.

Daniel Bär, Torsten Zesch, and Iryna Gurevych. 2013. DKPro Similarity: An open source framework for text similarity. In *Proceedings of ACL: System Demonstrations*, pages 121–126, Sofia, Bulgaria.

Or Biran, Samuel Brody, and Noémie Elhadad. 2011. Putting it simply: a context-aware approach to lexical simplification. In *Proceedings of ACL*, pages 496–501, Portland, Oregon.

Alexander Budanitsky and Graeme Hirst. 2006. Evaluating WordNet-based measures of Lexical Semantic Relatedness. *Computational Linguistics*, 32(1):13–47.

Evgeniy Gabrilovich and Shaul Markovitch. 2007. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *Proceedings of IJCAI*, pages 1606–1611, Hyderabad, India.

Oren Glickman and Ido Dagan. 2003. Acquiring lexical paraphrases from a single corpus. In *Proceedings of RANLP*, pages 81–90, Borovets, Bulgaria.

Taher Haveliwala, A. Gionis Dan Klein, and P. Indyk. 2002. Evaluating strategies for similarity search on the web. In *Proceedings of WWW*, pages 432–442, Honolulu, Hawaii.

Angelos Hliaoutakis, Giannis Varelas, Epimenidis Voutsakis, Euripides GM Petrakis, and Evangelos Milios. 2006. Information retrieval by semantic similarity. *International Journal on Semantic Web and Information Systems*, 2(3):55–73.

David Jurgens and Keith Stevens. 2010. The S-Space package: An open source package for word space models. In *Proceedings of the ACL: System Demonstrations*, pages 30–35, Uppsala, Sweden.

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

Alon Lavie and Michael J. Denkowski. 2009. The Meteor metric for automatic evaluation of Machine Translation. *Machine Translation*, 23(2-3):105–115.

Diana McCarthy and Roberto Navigli. 2009. The English lexical substitution task. *Language Resources and Evaluation*, 43(2):139–159.

Bridget T. McInnes, Pedersen Ted, and Serguei V.S. Pakhomov. 2009. UMLS-interface and UMLS-similarity: open source software for measuring paths and semantic similarity. In *Proceedings of AMIA*, pages 431–435, San Fransico, CA.

George A. Miller, R.T. Beckwith, Christiane D. Fellbaum, D. Gross, and K. Miller. 1990. WordNet: an online lexical database. *International Journal of Lexicography*, 3(4):235–244.

Michael Mohler, Razvan Bunescu, and Rada Mihalcea. 2011. Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In *Proceedings of ACL*, pages 752–762, Portland, Oregon.

Siddharth Patwardhan, Satanjeev Banerjee, and Ted Pedersen. 2003. Using measures of semantic relatedness for Word Sense Disambiguation. In *Proceedings of CICLing*, pages 241–257.

Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi. 2004. WordNet::Similarity: Measuring the relatedness of concepts. In *Proceedings of HLT-NAACL 2004: Demonstration Papers*, pages 38–41, Boston, Massachusetts.

Mohammad Taher Pilehvar and Roberto Navigli. 2014. A robust approach to aligning heterogeneous lexical resources. In *Proceedings of ACL*, pages 468–478, Baltimore, USA.

Mohammad Taher Pilehvar, David Jurgens, and Roberto Navigli. 2013. Align, Disambiguate and Walk: a Unified Approach for Measuring Semantic Similarity. In *Proceedings of ACL*, pages 1341–1351, Sofia, Bulgaria.

Herbert Rubenstein and John B. Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

Bharath Sriram, Dave Fuhry, Engin Demir, Hakan Ferhatosmanoglu, and Murat Demirbas. 2010. Short text classification in Twitter to improve information filtering. In *Proceedings of ACM SIGIR*, pages 841–842, Geneva, Switzerland.