Vector-space models for PPDB paraphrase ranking in context

Marianna Apidianaki
LIMSI, CNRS, Université Paris-Saclay
91403 Orsay, France
marianna.apidianaki@limsi.fr

Abstract

The PPDB is an automatically built database which contains millions of paraphrases in different languages. Paraphrases in this resource are associated with features that serve to their ranking and reflect paraphrase quality. This context-unaware ranking captures the semantic similarity of paraphrases but cannot serve to estimate their adequacy in specific contexts. We propose to use vector-space semantic models for selecting PPDB paraphrases that preserve the meaning of specific text fragments. This is the first work that addresses the substitutability of PPDB paraphrases in context. We show that vector-space models of meaning can be successfully applied to this task and increase the benefit brought by the use of the PPDB resource in applications.

1 Introduction

Paraphrases are alternative ways to convey the same information and can improve natural language processing by making systems more robust to language variability and unseen words. The paraphrase database (PPDB) (Ganitkevitch et al., 2013) contains millions of automatically acquired paraphrases in 21 languages associated with features that serve to their ranking. In PPDB's most recent release (2.0), such features include natural logic entailment relations, distributional and word embedding similarities, formality and complexity scores, and scores assigned by a supervised ranking model (Pavlick et al., 2015b). These features serve to identify good quality paraphrases but do not say much about their substitutability in context.

To judge the adequacy of paraphrases for specific instances of words or phrases, the surrounding context needs to be considered. This can be done using vector-space models of semantics which calculate the meaning of word occurrences in context based on distributional representations (Mitchell and Lapata, 2008; Erk and Padó, 2008; Dinu and Lapata, 2010; Thater et al., 2011). These models capture the influence of the context on the meaning of a target word through vector composition. More precisely, they represent the contextualised meaning of a target word w in context c by a vector obtained by combining the vectors of w and c using some operation such as component-wise multiplication or addition (Thater et al., 2011). We use this kind of representations to rank the PPDB paraphrases in context and retain the ones that preserve the semantics of specific text fragments. We evaluate the vector-based ranking models on data hand-annotated with lexical variants and compare the obtained ranking to confidence estimates available in the PPDB, highlighting the importance of context filtering for paraphrase selection.

2 Context-based paraphrase ranking

2.1 Paraphrase substitutability

The PPDB[1] provides millions of lexical, phrasal and syntactic paraphrases in 21 languages – acquired by applying bi- and multi-lingual pivoting on parallel corpora (Bannard and Callison-Burch, 2005) – and is largely exploited in applications (Denkowski and Lavie, 2010; Sultan et al., 2014; Faruqui et al.,

[1] http://paraphrase.org/#/download

Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 2028–2034,
Austin, Texas, November 1-5, 2016.

2015). PPDB paraphrases come into packages of different sizes (going from S to XXXL): smaller packages contain high-precision paraphrases while larger ones aim for high coverage. Until now, pivot paraphrases have been used as equivalence sets (i.e. all paraphrases available for a word are viewed as semantically equivalent) and their substitutability in context has not yet been addressed.

Substitutability might be restrained by several factors which make choosing the appropriate paraphrase for a word or phrase in different contexts a non-trivial task. In case of polysemous words, paraphrases describe different meanings and can lead to erroneous semantic mappings if substituted in texts (Apidianaki et al., 2014; Cocos and Callison-Burch, 2016). Even when paraphrases capture the same general sense, they are hardly ever equivalent synonyms and generally display subtle differences in meaning, connotation or usage (Edmonds and Hirst, 2002). Stylistic variation might also be present within paraphrase sets and substituting paraphrases that differ in terms of complexity and formality can result in a change in style (Pavlick and Nenkova, 2015). To increase paraphrase applicability in context, Pavlick et al. (2015a) propose to extract domain-specific pivot paraphrases by biasing the parallel training data used by the pivot method towards a specific domain. This customised model greatly improves paraphrase quality for the target domain but does not allow to rank and filter the paraphrases already in the PPDB according to specific contexts. To our knowledge, this is the first work that addresses the question of in-context substitutability of PPDB paraphrases. We show how existing substitutability models can be applied to this task in order to increase the usefulness of this large-scale resource in applications.

2.2 Vector-space models of paraphrase adequacy

Vector-based models of meaning determine a gradual concept of semantic similarity which does not rely on a fixed set of dictionary senses. They are used for word sense discrimination and induction (Schütze, 1998; Turney and Pantel, 2010) and can capture the contextualised meaning of words and phrases (Mitchell and Lapata, 2008; Erk and Padó, 2008; Thater et al., 2011). Vector composition methods build representations that go beyond individual words to obtain word meanings in context. Some models use explicit sense representations while others modify the basic meaning vector of a target word with information from the vectors of the words in its context. In the framework proposed by Dinu and Lapata (2010), for example, word meaning is represented as a probability distribution over a set of latent senses reflecting the out-of-context likelihood of each sense, and the contextualised meaning of a word is modeled as a change in the original sense distribution.[2] Reisinger and Mooney (2010) propose a multi-prototype vector-space model of meaning which produces multiple "sense-specific" vectors for each word, determined by clustering the contexts in which the word appears (Schütze, 1998). The cluster centroids serve as prototype vectors describing a word's senses and the meaning of a specific occurrence is determined by choosing the vector that minimizes the distance to the vector representing the current context. On the contrary, Thater et al. (2011) use no explicit sense representation. Their models allow the computation of vector representations for individual uses of words, characterising the specific meaning of a target word in its sentential context. When used for paraphrase ranking, these models derive a contextualised vector for a target word by reweighting the components of its basic meaning vector on the basis of the context of occurrence.[3] Paraphrase candidates for a target word are then ranked according to the cosine similarity of their basic vector representation to the contextualised vector of the target.[4]

3 Experimental Set-up

Data In our experiments, we use the CoInCo corpus (Kremer et al., 2014), a subset of the "Manually Annotated Sub-Corpus" MASC (Ide et al., 2010) which comprises more than 15K word in-

[2]The latent senses are induced using non-negative matrix factorization (NMF) (Lee and Seung, 2001) and latent Dirichlet allocation (LDA) (Blei et al., 2003).

[3]Depending on the model, the vector combination function might be addition or multiplication of vector elements.

[4]Thater et al.'s (2011) models delivered best results in paraphrase ranking on the CoInCo corpus (Kremer et al., 2014) and the SemEval-2007 Lexical Substitution dataset (McCarthy and Navigli, 2007).

	$\|P\| > 1$			$\|P\| \geq 1$
PPDB	# Instances	# Lemmas	Avg $\|P\|$	# Instances
S	2146	560	2.67	5573
M	3716	855	2.92	7771
L	6228	1394	3.57	10100
XL	13344	2822	10.33	14060
XXL	14507	3308	185.09	14593

Table 1: Number of CoInCo instances and distinct lemmas covered by each PPDB package.

stances manually annotated with single and multi-word substitutes. The manual annotations serve to evaluate the performance of the vector-space models on the task of ranking PPDB paraphrases. For each annotated English target word (noun, verb, adjective or adverb) in CoInCo, we collect the lexical paraphrases ($P = \{p_1, p_2, ..., p_n\}$) available for the word in each PPDB package (from S to XXL).[5] We do not filter by syntactic label as annotations often include substitutes of different grammatical categories. Table 1 shows the number of CoInCo tokens with paraphrases in each PPDB package and the average size of the retained paraphrase sets. The larger the size of the resource, the greater the coverage of target words in CoInCo. The last column of the table gives the total number of instances covered, including the ones with only one paraphrase. In the ranking experiments, we focus on lemmas having more than one paraphrase in the PPDB.[6]

Methodology We follow the methodology proposed in Kremer et al. (2014) to explore the extent to which vector-based models can select appropriate paraphrases for words in context. Given a target word w in a sentential context and a set of paraphrases P extracted for w from a PPDB package, the task is to rank the elements in P according to their adequacy as paraphrases of w in the given context.

We carry out experiments with three versions of the Thater et al. (2011) ranking model: (a) a *syntactically structured* model (Syn.Vec) that uses vectors recording co-occurrences based on dependency triples, explicitly recording syntactic role information within the vectors; (b) a *syntactically filtered model* (Filter.Vec) using dependency-based co-occurrence information without explicitly representing the syntactic role in the vector representations, as in Padó and Lapata (2007); (c) a bag of words model (Bow.Vec) using a window of $\pm$ 5 words. Co-occurrence counts were extracted from the English Gigaword corpus[7] analysed with Stanford dependencies (de Marneffe et al., 2006). The syntactic model vectors are based on dependency triples that occur at least 5 times in the corpus and have a PMI score of at least 2. The same thresholds apply to the bag of words model where the frequency threshold defines the minimum number of times that two words have been observed in the same context window. The task of the vector-space models for each target word instance is to rank the contents of the corresponding paraphrase set (which contains all the substitution candidates available for the target in the PPDB) so that the actual substitutes are ranked higher than the rest. For example, *newspaper, manuscript* and *document* are good paraphrase candidates for *paper* but we would expect *newspaper* to be ranked higher than the other two in this sentence: "*the paper's local administrator*".

A contextualised vector is derived from the basic meaning vector of a target word w by reinforcing its dimensions that are licensed by the context of the specific instance under consideration. In the Bow.Vec model, the context is made up of 5 words before and after the target while in the syntactic models, it corresponds to the target's direct syntactic dependents. The contextualised vector for w is obtained through vector addition and contains information about the context words. Paraphrase candidates are ranked according to the cosine similarity between the contextualised vector of the target word and the basic meaning vectors of the candidates. Following Kremer et al. (2014), we compare the resulting ranked list to the CoInCo gold standard annotation (the paraphrase set of the target instance) using Generalised Average Precision (GAP) (Kishida, 2005) and annotation frequency as weights. GAP scores range between 0 and 1: a score of 1 indicates a perfect ranking in which all correct substitutes precede all incorrect ones, and

[5] Since the XXL package covers almost all annotated instances in CoInCo (14,507 out of 15,629) and there are 185.09 paraphrases in average for each instance, we exclude the XXXL package from these experiments.

[6] We retain paraphrases of the lemmatised forms of the target words but these unsupervised ranking models can be easily applied to the whole PPDB resource and in different languages.

[7] http://catalog.ldc.upenn.edu/LDC2003T05

	PPDB	Bow.Vec	Syn.Vec	Filter.Vec	Google	AGiga	Ppdb1	Ppdb2	Parprob	Random (5)
$\|P\| > 1$	S	**0.91**	**0.91**	**0.91**	0.78	0.86	0.66	0.83	0.66	0.78
	M	0.91	0.91	**0.92**	0.79	0.87	0.68	0.84	0.68	0.79
	L	0.90	0.90	**0.91**	0.78	0.85	0.66	0.83	0.66	0.77
	XL	0.78	**0.79**	**0.79**	0.58	0.67	0.44	0.66	0.43	0.58
	XXL	0.53	0.56	0.57	0.27	0.36	0.12	**0.58**	0.12	0.27
$\|P\| \geq 1$	S	**0.97**	**0.97**	**0.97**	0.91	0.95	0.87	0.93	0.87	0.91
	M	**0.96**	**0.96**	**0.96**	0.90	0.94	0.85	0.92	0.85	0.90
	L	**0.94**	**0.94**	**0.94**	0.87	0.91	0.79	0.90	0.79	0.86
	XL	0.79	**0.80**	**0.80**	0.60	0.69	0.47	0.68	0.46	0.60
	XXL	0.54	0.56	0.58	0.28	0.37	0.13	**0.59**	0.14	0.28

Table 2: Average GAP scores for the contextual models, five paraphrase adequacy methods and the random ranking baseline against the gold CoInCo annotations. Scores reported for different sizes of the PPDB (from S to XXL).

correct high-weight substitutes precede low-weight ones. For calculating the GAP score, we assign a very low score (0.001) to paraphrases that are not present in CoInCo for a target word (i.e. not proposed by the annotators).

4 Results

The average GAP scores obtained by the three vector-space models (Bow.Vec, Syn.Vec and Filter.Vec) are shown in Table 2. The upper part of the table reports scores obtained for words with more than one paraphrase in the PPDB ($|P| > 1$) while the lower part gives the scores for all words.

We compare the GAP scores to five different rankings reflecting paraphrase quality in the PPDB (Pavlick et al., 2015b). We retain the following scores: 1. **AGigaSim** captures the distributional similarity of a phrase e_1 and its paraphrase e_2 computed according to contexts observed in the Annotated Gigaword corpus (Napoles et al., 2011); 2. **GoogleNgramSim** reflects the distributional similarity of e_1 and e_2 computed according to contexts observed in the Google Ngram corpus (Brants and Franz, 2006); 3. **ParProb**: the paraphrase probability of e_2 given the original phrase e_1 (Bannard and Callison-Burch, 2005); 4. **Ppdb1**: the heuristic scoring used for ranking in the original release of the PPDB (Ganitkevitch et al., 2013); 5. **Ppdb2**: the improved ranking of English paraphrases available in PPDB 2.0. The results are also compared to the output of a baseline where the paraphrases are randomly ranked. The reported baseline figures are PPDB package-specific since a different paraphrase set is retained from each package, and correspond to averages over 5 runs. The quality of the ranking produced by the baseline clearly decreases as the size of the PPDB resource increases due to the higher number of retained paraphrases which makes ranking harder.

The results in the upper part of the table show that the vector-space models provide a better ranking than the PPDB estimates and largely outperform the random baseline. The three models perform similarly on this ranking task according to average GAP with the syntactically-informed models getting slightly higher scores. Differences between Syn.Vec and Filter.Vec, as well as between Bow.Vec and the syntactic models, are highly significant in the XL and XXL packages (p-value < 0.001) as computed with approximate randomisation (Padó, 2006). In the L package, the difference between Syn.Vec and Filter.Vec is significant ($p < 0.05$) and the one between Bow.Vec and Filter.Vec is highly significant. Finally, in the M package, only the difference between Bow.Vec and Filter.Vec is significant ($p < 0.05$), while Syn.Vec and Filter.Vec seem to deal similarly well with the contents of this package.

Two PPDB ranking methods, AGiga and Ppdb2, obtain good results. AgigaSim reflects the distributional similarity of the paraphrases in the Annotated Gigaword corpus (Napoles et al., 2011). As noted by Kremer et al. (2014), the whole-document annotation in CoInCo faces the natural skewed distribution towards predominant senses which favors non-contextualised baseline models. The good performance of Ppdb2 is due to the use of a supervised scoring model trained on human judgments of paraphrase quality. The human judgments were

used to fit a regression to the features available in PPDB 1.0 plus numerous new features including cosine word embedding similarity, lexical overlap features, WordNet features and distributional similarity features.[8] The small difference observed between the Ppdb2 and the syntactic models score in the XXL package is highly significant. For the moment, Ppdb2 scores are available in the PPDB only for English. Since the vector-space methodology is unsupervised and language independent, it could be easily applied to paraphrase ranking in other languages. The performance of the models remains high with the XL package which contains paraphrase sets of reasonable size (about 10 paraphrases per word) and ensures a high coverage, and lowers in XXL which contains 185 paraphrases in average per word (cf. Table 1). To use this package more efficiently, one could initially reduce the number of erroneous paraphrases on the basis of the Ppdb2 score which provides a good ranking of the XXL package contents before applying the vector-based models.

The increase in GAP score observed when words with one paraphrase are considered shows that these paraphrases are often correct. Here too, the contextual models provide a better ranking than the out-of-context scores and outperform the random baseline. As in the previous case, the Ppdb2 score is slightly higher in the XXL package.

5 Conclusion

We have shown that vector-based models of semantics can be successfully applied to in-context ranking of PPDB paraphrases. Allowing for better context-informed substitutions, they can be used to filter PPDB paraphrases on the fly and select variants preserving the correct semantics of words and phrases in texts. This processing would be beneficial to numerous applications that need paraphrase support (e.g. summarisation, query reformulation and language learning), providing a practical means for exploiting the extensive multilingual knowledge available in the PPDB resource.

This study opens up many avenues for future work. Although tested on English, the proposed methodology can be applied to all languages in the PPDB even to the ones that do not dispose of a dependency parser (as shown by the high performance of the Bow.Vec models).

An ideal testbed for evaluation in a real application and on multiple languages is offered by MT evaluation. The METEOR-NEXT metric (Denkowski and Lavie, 2010) provides a straightforward framework for testing as it already exploits PPDB paraphrases for capturing sense correspondences between text fragments. In its current version, the metric views paraphrases as equivalent classes which can lead to erroneous sense mappings due to semantic distinctions present in the paraphrase sets. We have recently showed that the context-based filtering of semantic variants improves METEOR's correlation with human judgments of translation quality (Marie and Apidianaki, 2015). We believe that a context-based paraphrase ranking mechanism will enhance correct substitutions and further improve the metric. Last but not least, the paraphrase vectors can be used for mapping the contents of the PPDB resource to other multilingual resources for which vector representations are available (Camacho-Collados et al., 2015a; Camacho-Collados et al., 2015b). The interest of mapping paraphrases in the vector space to concepts found in existing semantic resources is twofold: it would permit to analyse the semantics of the paraphrases by putting them into correspondence with explicit concept representations and would serve to enrich other semantic resources (e.g. BabelNet synsets) with semantically similar paraphrases.

Handling phrasal paraphrases is another natural extension of this work. We consider using a vector space model of semantic composition to calculate the meaning of longer candidate paraphrases (Dinu et al., 2013; Paperno et al., 2014) and select appropriate substitutes for phrases in context.

Acknowledgments

We would like to thank Stefan Thater for sharing the vector-space models, Benjamin Marie for his support with the paraphrase ranking models and the anonymous reviewers for their valuable comments and suggestions.

[8] The features used for computing the paraphrase ranking in PPDB 2.0 are described in detail in Pavlick et al. (2015b).

References

Marianna Apidianaki, Emilia Verzeni, and Diana McCarthy. 2014. Semantic Clustering of Pivot Paraphrases. In *Proceedings of LREC*, Reykjavik, Iceland.

Colin Bannard and Chris Callison-Burch. 2005. Paraphrasing with Bilingual Parallel Corpora. In *Proceedings of ACL*, pages 597–604, Ann Arbor, Michigan, USA.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Thorsten Brants and Alex Franz. 2006. *The Google Web 1T 5-gram Corpus Version 1.1*. LDC2006T13, Philadelphia.

José Camacho-Collados, Mohammad Taher Pilehvar, and Roberto Navigli. 2015a. Nasari: a novel approach to a semantically-aware representation of items. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 567–577, Denver, Colorado, May–June.

José Camacho-Collados, Mohammad Taher Pilehvar, and Roberto Navigli. 2015b. A unified multilingual semantic representation of concepts. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 741–751, Beijing, China, July.

Anne Cocos and Chris Callison-Burch. 2016. Clustering paraphrases by word sense. In *The 2016 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2016)*, San Diego, California, USA.

Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *To appear at LREC-06*.

Michael Denkowski and Alon Lavie. 2010. METEOR-NEXT and the METEOR Paraphrase Tables: Improved Evaluation Support for Five Target Languages. In *Proceedings of WMT/MetricsMATR*, pages 339–342, Uppsala, Sweden.

Georgiana Dinu and Mirella Lapata. 2010. Measuring distributional similarity in context. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1162–1172, Cambridge, MA, October.

Georgiana Dinu, Nghia The Pham, and Marco Baroni. 2013. Dissect - distributional semantics composition toolkit. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 31–36, Sofia, Bulgaria, August.

Philip Edmonds and Graeme Hirst. 2002. Near-Synonymy and Lexical Choice. *Computational Linguistics*, 28(2):105–144.

Katrin Erk and Sebastian Padó. 2008. A Structured Vector Space Model for Word Meaning in Context. In *Proceedings of EMNLP*, pages 897–906, Honolulu, Hawaii.

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A. Smith. 2015. Retrofitting word vectors to semantic lexicons. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1606–1615, Denver, Colorado.

Juri Ganitkevitch, Benjamin VanDurme, and Chris Callison-Burch. 2013. PPDB: The Paraphrase Database. In *Proceedings of NAACL*, Atlanta, Georgia, USA.

Nancy Ide, Collin Baker, Christiane Fellbaum, and Rebecca Passonneau. 2010. The manually annotated sub-corpus: A community resource for and by the people. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 68–73, Uppsala, Sweden.

Kazuaki Kishida. 2005. Property of average precision and its generalization: An examination of evaluation indicator for information retrieval experiments. Technical report, Technical Report NII-2005-014E.

Gerhard Kremer, Katrin Erk, Sebastian Padó, and Stefan Thater. 2014. What Substitutes Tell Us - Analysis of an "All-Words" Lexical Substitution Corpus. In *Proceedings of EACL*, pages 540–549, Gothenburg, Sweden.

Daniel D. Lee and H. Sebastian Seung. 2001. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems 13 (NIPS 2000)*, pages 556–562. MIT Press.

Benjamin Marie and Marianna Apidianaki. 2015. Alignment-based sense selection in METEOR and the RATATOUILLE recipe. In *Proceedings of WMT*, pages 385–391, Lisbon, Portugal.

Diana McCarthy and Roberto Navigli. 2007. Semeval-2007 task 10: English lexical substitution task. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 48–53, Prague, Czech Republic.

Jeff Mitchell and Mirella Lapata. 2008. Vector-based Models of Semantic Composition. In *Proceedings of ACL/HLT*, pages 236–244, Columbus, Ohio, USA.

Courtney Napoles, Chris Callison-Burch, Juri Ganitkevitch, and Benjamin Van Durme. 2011. Paraphrastic sentence compression with a character-based metric: Tightening without deletion. In *Proceedings of the Workshop on Monolingual Text-To-Text Generation*, pages 84–90, Portland, Oregon.

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

Sebastian Padó, 2006. *User's guide to `sigf`: Significance testing by approximate randomisation*.

Denis Paperno, Nghia The Pham, and Marco Baroni. 2014. A practical and linguistically-motivated approach to compositional distributional semantics. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 90–99, Baltimore, Maryland, June.

Ellie Pavlick and Ani Nenkova. 2015. Inducing lexical style properties for paraphrase and genre differentiation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 218–224, Denver, Colorado.

Ellie Pavlick, Juri Ganitkevitch, Tsz Ping Chan, Xuchen Yao, Benjamin Van Durme, and Chris Callison-Burch. 2015a. Domain-Specific Paraphrase Extraction. In *Proceedings of ACL/IJCNLP*, pages 57–62, Beijing, China.

Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2015b. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of ACL/IJCNLP*, pages 425–430, Beijing, China.

Joseph Reisinger and Raymond J. Mooney. 2010. Multi-prototype vector-space models of word meaning. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 109–117, Los Angeles, California.

Hinrich Schütze. 1998. Automatic Word Sense Discrimination. *Computational Linguistics*, 24:97–123.

Md Arafat Sultan, Steven Bethard, and Tamara Sumner. 2014. Dls@cu: Sentence similarity from word alignment. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 241–246, Dublin, Ireland, August.

Stefan Thater, Hagen Fürstenau, and Manfred Pinkal. 2011. Word Meaning in Context: A Simple and Effective Vector Model. In *Proceedings of IJCNLP*, pages 1134–1143, Chiang Mai, Thailand.

Peter D. Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37(1):141–188.