Measuring Word Relatedness Using Heterogeneous Vector Space Models

Wen-tau Yih
Microsoft Research
One Microsoft Way
Redmond, WA
scottyih@microsoft.com

Vahed Qazvinian*
Department of EECS
University of Michigan
Ann Arbor, MI
vahed@umich.edu

Abstract

Noticing that different information sources often provide complementary coverage of word sense and meaning, we propose a simple and yet effective strategy for measuring lexical semantics. Our model consists of a committee of vector space models built on a text corpus, Web search results and thesauruses, and measures the semantic word relatedness using the averaged cosine similarity scores. Despite its simplicity, our system correlates with human judgements better or similarly compared to existing methods on several benchmark datasets, including WordSim353.

1 Introduction

Measuring the semantic relatedness of words is a fundamental problem in natural language processing and has many useful applications, including textual entailment, word sense disambiguation, information retrieval and automatic thesaurus discovery. Existing approaches can be roughly categorized into two kinds: knowledge-based and corpus-based, where the former includes graph-based algorithms and similarity measures operating on a lexical database such as WordNet (Budanitsky and Hirst, 2006; Agirre et al., 2009) and the latter consists of various kinds of vector space models (VSMs) constructed with the help of a large collection of text (Reisinger and Mooney, 2010; Radinsky et al., 2011). In this paper, we present a conceptually simple model for solving this problem. Observing that various kinds of information sources, such as general text corpora, Web search results and thesauruses, have different word and sense coverage, we first build individual vector space models from each of them separately. Given two words, each VSM measures the semantic relatedness by the cosine similarity of the corresponding vectors in its space. The final prediction is simply the averaged cosine scores derived from these VSMs. Despite its simplicity, our system surprisingly yields very strong empirical performance. When comparing the predictions with the human annotations on four different datasets, our system achieves higher correlation than existing methods on two datasets and provides very competitive results on the others.

The rest of this paper is organized as follows. Section 2 briefly reviews the related work. Section 3 details how we construct each individual vector space model, followed by the experimental evaluation in Section 4. Finally, Section 5 concludes the paper.

2 Background

Prior work on measuring lexical semantics can be categorized as knowledge-based or corpus-based. Knowledge-based methods leverage word relations encoded in lexical databases such as WordNet and provide graph-based similarity measures. Detailed comparisons of these methods can be found in (Budanitsky and Hirst, 2006). Corpus-based methods assume related words tend to co-occur or to appear in similar context. For example, Gabrilovich and Markovitch (2007) measure word relatedness by whether they tend to occur in the same Wikipedia topic. In contrast, Reisinger and Mooney (2010) use the conventional "context vector" – neighboring

* Work conducted while interning at Microsoft Research.

terms of the occurrences of a target word – as the word representation. In addition, they argue that it is difficult to capture different senses of a word with a single vector, and introduce a multi-prototype representation. More recently, Radinsky et al. (2011) analyze the temporal aspects of words and argue that non-identical terms in two term vectors should also be compared based on their temporal usage when computing the similarity score. They construct the vectors using Wikipedia titles, Flickr image tags, and Del.icio.us bookmarks, and extract the temporal frequency of each concept from 130 years of New York Times archive. Methods that combine models from different sources do exist. For instance, Agirre et al. (2009) derive a WordNet-based measure using PageRank and combined it with several corpus-based vector space models using SVMs.

3 Vector Space Models from Heterogeneous Sources

In this section, we describe how we construct various vector space models (VSMs) to represent words, including *corpus*-based, *Web*-based and *thesaurus*-based methods.

Corpus-based VSMs follow the standard "distributional hypothesis," which states that words appearing in the same *contexts* tend to have similar meaning (Harris, 1954). Each target word is thus represented by a high-dimensional sparse term-vector that consists of words occurring in its context. Given a corpus, we first collect terms within a window of $[-10, +10]$ centered at each occurrence of a target word. This bag-of-words representation is then mapped to the TF-IDF term vector: each term is weighted by $\log(freq) \times \log(N/df)$, where $freq$ is the number of times the term appears in the collection, df the document frequency of the term in the whole corpus and N the number of total documents. We further employed two simple techniques to improve the quality of these term-vectors: *vocabulary* and *term* trimming. Top 1,500 terms with high document frequency values are treated as stopwords and removed from the vocabulary. Moreover, we adopted a document-specific feature selection method (Kolcz and Yih, 2007) designed originally for text classification and retain only the top 200 high-weighted terms for each term-vector[1]. The corpus-based VSMs are created using English Wikipedia (Snapshot of Nov. 2010), consisting of 917M words after preprocessing (markup tags removal and sentence splitting).

Web-based VSMs leverage Web search results to form a vector of each query (Sahami and Heilman, 2006). For each word to compare, we issue it as a query and retrieve the set of relevant snippets (top 30 in our experiments) using a popular commercial search engine, Bing. All these snippets together are viewed as a pseudo-document and mapped to a TF-IDF vector as in the corpus-based method. We do not allow for automatic query expansion in our experiments to ensure that the retrieved snippets are directly relevant to the target word and not expansions based on synonyms, hypernyms or hyponyms. We apply vocabulary trimming (top 1,000 terms with high DF values), but not term-trimming as the vectors have much fewer terms due to the small number of snippets collected.

Both the corpus-based and Web-based VSMs rely on the distributional hypothesis, which is often criticized for two weaknesses. The first is that word pairs that appear in the same context or co-occur are not necessarily highly semantically related. For example, "bread" and "butter" often have cosine scores higher than synonyms using corpus-based vectors because of the phrase "bread and butter". The second is that general corpora often have skewed coverage of words due to the Zipf's law. Regardless of the size of the corpus, the number of occurrences of a rarely used word is typically very low, which makes the quality of the corresponding vector unreliable. To address these two issues, we include the **thesaurus**-based VSMs in this work as well. For each group of similar words (synset) defined in the thesaurus, we treat it as a "document" and create a document–word matrix, where each word is again weighted using its TF-IDF value. Each column vector in this matrix is thus the thesaurus-based vector of the corresponding word. Notice that given two words and their corresponding vectors, the cosine score is more general than simply checking

[1] In preliminary experiments, we found that active terms with low TF-IDF values tend to be noise. By aggressively removing them, the quality of the term-vectors can be significantly improved.

whether these two words belong to a group of similar words, as it judges how often they overlap in various documents (i.e., sets of similar words). We explored using two different thesauri in our experiments: WordNet and the Encarta thesaurus developed by Bloomsbury Publishing, where the former consists of 227,446 synsets and 190,052 words and the latter contains 46,945 synsets and 50,184 words. Compared to existing *knowledge*-based approaches, our VSM transformation is very simple and straightforward. It is also easy to extend our method to other languages as only a thesaurus is required rather than a complete lexical database such as WordNet.

4 Experimental Evaluation

In this section, we evaluate the quality of the VSMs constructed using methods described in Section 3 on different benchmark datasets, as well as the performance when combining them.

4.1 Benchmark datasets

We follow the standard evaluation method, which directly tests the correlation of the word relatedness measures with human judgements on a set of word pairs, using the Spearman's rank correlation coefficient. Our study was conducted using four different datasets, including WS-353, RG-65, MC-30 and MTurk-287.

The WordSim353 dataset (**WS-353**) is the largest among them and has been used extensively in recent work. Originally collected by Finkelstein et al. (2001), the dataset consists of 353 word pairs. The degree of relatedness of each pair is assessed on a 0-10 scale by 13-16 human judges, where the mean is used as the final score. Examining the relations between the words in each pair, Agirre et al. (2009) further split this dataset into *similar* pairs (**WS-sim**) and *related* pairs (**WS-rel**), where the former contains synonyms, antonyms, identical words and hyponyms/hypernyms and the latter capture other word relations. Collected by Rubenstein and Goodenough (1965), **RG-65** contains 65 pairs of words that are either synonyms or unrelated, assessed on a 0-4 scale by 51 human subjects. Taking 30 pairs from them, Miller and Charles (1991) created the (**MC-30**) dataset by reassessing these word pairs using 38 subjects. These 30 pairs of words are also a subset of WS-353. Although these three datasets contain overlapping word pairs, their scores are different because of the degree of relatedness were given by different human subjects. In addition to these datasets, we also evaluate our VSMs on the **Mturk-287** dataset that consists of 287 word pairs collected by (Radinsky et al., 2011) using Amazon MTurk.

4.2 Results and Analysis

Table 1 summarizes the results of various methods, where the top part lists the performance of state-of-the-art systems and the bottom shows the results of individual vector space models, as well as combining these models using the averaged cosine scores. We make several observations here. First, while none of the four VSMs we tested outperforms the best existing systems on the benchmark datasets, surprisingly, using the averaged cosine scores of these models, the performance is improved substantially. It achieves higher Spearman's rank coefficient on WS-353 and MTurk-287 than any other systems[2] and are close to the state-of-the-art on MC-30 and RG-65. Unlike some approach like (Hughes and Ramage, 2007), which performs well on some datasets but poorly on others, combing the VSMs from heterogeneous sources is more robust. Individually, we notice that Wikipedia context VSM provides consistently strong results, while thesaurus-based models work only reasonable on MC-30 and RG-65, potentially because other datasets contain more out-of-vocabulary words or proper nouns. Due to the inherent ambiguity of the task, there is a high variance among judgements from different annotators. Therefore, it is unrealistic to assume any of the methods can correlate perfectly to the mean human judgement scores. In fact, the inter-agreement study done on the WS-353 dataset indicates that the result of our approach of combining heterogeneous VSMs is close to the averaged human performance.

It is intriguing to see that by using the averaged cosine scores, the performance can be improved over the best individual model (i.e., Wikipedia). Examining the scores of some word pairs carefully sug-

[2]This may not be statistically significant. Without having the exact output of existing systems, it is difficult to conduct a robust statistical significance test given the small sizes of these datasets.

Method	Spearman's ρ					
	WS-353	**WS-sim**	**WS-rel**	**MC-30**	**RG-65**	**MTurk-287**
(Radinsky et al., 2011)	0.80	-	-	-	-	0.63
(Reisinger and Mooney, 2010)	0.77	-	-	-	-	-
(Agirre et al., 2009)	0.78	0.83	0.72	**0.92**	**0.96**	-
(Gabrilovich and Markovitch, 2007)	0.75	-	-	-	-	0.59
(Hughes and Ramage, 2007)	0.55	-	-	0.90	0.84	-
Web Search	0.56	0.56	0.54	0.48	0.44	0.44
Wikipedia	0.73	0.80	0.73	0.87	0.83	0.62
Bloomsbury	0.45	0.60	0.60	0.71	0.78	0.29
WordNet	0.37	0.49	0.49	0.79	0.78	0.25
Combining VSMs	**0.81**	**0.87**	**0.77**	0.89	0.89	**0.68**

Table 1: The performance of the state-of-the-art methods and different vector space models on measuring semantic word relatedness using the cosine similarity.

gests the broader coverage of different words and senses could be the reason. For example, some of the words in the datasets have multiple senses, such as "jaguar vs. car" and "jaguar vs. cat". Although in previous work, researchers try to capture word senses using different vectors (Reisinger and Mooney, 2010) from the same text corpus, this is in fact difficult in practice. The usage of words in a big text corpus, which contains diversified topics, may still be biased to one word sense. For example, in the Wikipeida term vector that represents "jaguar", we found that most of the terms there are related to "cat". Although some terms are associated with the "car" meaning, the signals are rather weak. Similarly, WordNet does not indicate "jaguar" could be related to "car" at all. In contrast, the "car" sense of "jaguar" dominates the vector created using the search engine. As a result, incorporating models from different sources could be more effective than relying on word sense discovering algorithms operating solely on one corpus. Another similar but different example is the pair of "bread" and "butter", which are treated as synonyms by corpus-based VSMs, but is demoted after adding the thesaurus-based models.

5 Conclusion

In this paper we investigated the usefulness of heterogeneous information sources in improving measures of semantic word relatedness. Particularly, we created vector space models using 4 data sources from 3 categories (corpus-based, Web-based and thesaurus-based) and found that simply averaging the cosine similarity derived from these models yields a very robust measure. Other than directly applying it to measuring semantic relatedness, our approach is complementary to more sophisticated similarity measures such as developing kernel functions for different structured data (Croce et al., 2011), where the similarity between words serves as a basic component.

While this result is interesting and encouraging, it also raises several research questions, such as how to enhance the quality of each vector space model and whether the models can be combined more effectively[3]. We also would like to study whether similar techniques can be useful when comparing longer text segments like phrases or sentences, with potential applications in paraphrase detection and recognizing textual entailment.

Acknowledgments

We thank Joseph Reisinger for providing his prototype vectors for our initial study, Silviu-Petru Cucerzan for helping process the Wikipedia files and Geoffrey Zweig for preparing the Bloomsbury thesaurus data. We are also grateful to Chris Meek for valuable discussions and to anonymous reviewers for their comments.

[3]We conducted some preliminary experiments (not reported here) on tuning the weights of combining different models based on cross-validation, but did not find consistent improvements, perhaps due to the limited size of the data.

References

E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Paşca and A. Soroa. 2009. A study on similarity and relatedness using distributional and wordnet-based approaches. In *NAACL '09*, pages 19–27.

A. Budanitsky and G. Hirst. 2006. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32:13–47, March.

D. Croce, A. Moschitti, and R. Basili. 2011. Structured lexical similarity via convolution kernels on dependency trees. In *Proceedings of EMNLP 2011*, pages 1034–1046, July.

L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin. 2001. Placing search in context: The concept revisited. In *WWW*, pages 406–414. ACM.

E. Gabrilovich and S. Markovitch. 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI '07*, pages 1606–1611.

Z. Harris. 1954. Distributional structure. *Word*, 10(23):146–162.

T. Hughes and D. Ramage. 2007. Lexical semantic relatedness with random graph walks. In *Proceedings of EMNLP-CoNLL-2007*, pages 581–589.

A. Kolcz and W. Yih. 2007. Raising the baseline for high-precision text classifiers. In *KDD '07*, pages 400–409.

G. Miller and W. Charles. 1991. Contextual correlates of semantic similarity. *Language and cognitive processes*, 6(1):1–28.

K. Radinsky, E. Agichtein, E. Gabrilovich, and S. Markovitch. 2011. A word at a time: computing word relatedness using temporal semantic analysis. In *WWW '11*, pages 337–346.

J. Reisinger and R. Mooney. 2010. Multi-prototype vector-space models of word meaning. In *NAACL '10*.

H. Rubenstein and J. Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*, 8:627–633, October.

M. Sahami and T. Heilman. 2006. A web-based kernel function for measuring the similarity of short text snippets. In *Proceedings of the 15th international conference on World Wide Web*, pages 377–386. ACM.