Lemon and Tea Are Not Similar: Measuring Word-to-Word Similarity by Combining Different Methods

Conference Paper · April 2015

DOI: 10.1007/978-3-319-18111-0_25

5 authors, including:

Rajendra Banjade

The University of Memphis

18 PUBLICATIONS **64** CITATIONS

SEE PROFILE

Nabin Maharjan

The University of Memphis

6 PUBLICATIONS **12** CITATIONS

SEE PROFILE

Dipesh Gautam

The University of Memphis

6 PUBLICATIONS **11** CITATIONS

SEE PROFILE

Lemon and Tea Are Not Similar: Measuring Word-to-Word Similarity by Combining Different Methods

Rajendra Banjade, Nabin Maharjan, Nobal B. Niraula, Vasile Rus, Dipesh Gautam

Department of Computer Science, The University of Memphis
Memphis, TN, 38152, USA
{rbanjade, nmharjan, nbnraula, vrus, dgautam}@memphis.edu

Abstract. Substantial amount of work has been done on measuring word-to-word relatedness which is also commonly referred as similarity. Though relatedness and similarity are closely related, they are not the same as illustrated by the words *lemon* and *tea* which *are related but not similar*. The relatedness takes into account a broader range of relations while similarity only considers subsumption relations to assess how two objects are similar. We present in this paper a method for measuring the semantic similarity of words as a combination of various techniques including knowledge-based and corpus-based methods that capture different aspects of similarity. Our corpus based method exploits state-of-the-art word representations. We performed experiments with a recently published significantly large dataset called Simlex-999 and achieved a significantly better correlation ($\rho = 0.642$, $P < 0.001$) with human judgment compared to the individual performance.

Keywords: Similarity, Relatedness, Word-to-Word Similarity

1 Introduction

Understanding the meaning (semantics) of texts is one of the core problems in the field of Natural Language Processing (NLP). Semantic similarity, i.e. quantifying and deciding how similar the meanings of two given texts are, is one approach to the natural language understanding problem. In this paper, we focus on the more specific task of measuring the similarity of words, i.e. quantifying to what extent two words have similar meanings. The dictionary definition of similarity is: *resembling without being identical* (cf. Oxford Dictionary). For example, *intelligent* and *genius* are highly similar. On the other hand, measuring relatedness (also called association) is to find out to what extent the given words are related or associated to each other. The related words are not necessarily similar words. For instance, *lemon* and *tea* are related but they are not similar as they mean very different things. We focus here on assessing how similar two words are.

A considerable amount of effort has been put on calculating the semantic relatedness or association of words which is sometimes referred as similarity. Existing methods, especially those based on co-occurrence of words in a large collection of documents, have achieved significant results on measuring the relatedness of words [9]. However, explicitly quantifying the similarity of words fosters the development of applications that benefit from similarity than those which take into account a broader range of relations. To this end, we present a method that combines several diverse approaches that rely on corpus and knowledge bases. Our hypothesis is that different methods capture different aspects of semantic similarity and their meaningful combination produces better results.

The task of word-to-word similarity has many applications, such as automatic answer grading [7], [18], [24], plagiarism detection [22]. In general, word-to-word similarity can be combined to measure similarity of texts at various levels, thus, being useful in a wide range of applications. For example, word similarity is crucial to accurately measure the correctness of student answers in educational technologies such as intelligent tutoring systems. In such systems, a widely used approach is to assess how semantically similar a target student answer, e.g. to a Physics problem, is to a reference answer, i.e. an answer provided by an expert and which is deemed correct. For instance, if a student answer contains the word *velocity* and the expert answer includes the word *acceleration*, the question is whether we should deem the student response correct. We argue that semantic relatedness measures would lead to an incorrect assessment as the relatedness score will be high. While being related, *acceleration* and *velocity* are two different concepts. Semantic similarity methods would not assess *acceleration* and *velocity* as highly similar.

Based on the types of resources used, the methods that measure semantic relatedness or similarity are broadly of two types: those that rely on knowledge bases, such as WordNet [4], and those that infer word associations from bigger collections of texts based on word co-occurrence, called distributional methods. In the knowledge base category, WordNet based methods for calculating word similarity and relatedness are quite popular [14], [16]. On the other hand, distributional similarity methods include LSA [13], LDA [2], HAL [3], ESA [6], GloVe [21]. Recently, Deep Learning based methods [17] are also in use.

Previous methods, individually or as a combination of different methods, have yielded very good performance when it comes to measuring relatedness [28]. However, as [9] explored, distributional similarity methods are not capturing well the true similarity between words. They also published a dataset containing 999 word pairs (called Simlex-999) with human rated similarity scores. We combined various knowledge based and corpus based methods by applying Linear Regression and Support Vector regression to measure semantic similarity and achieved state-of-the-art results.

The rest of the paper is organized as follows. The next section provides an overview of related work. Then, we describe the approach for combining different methods. The Experiments and Results section describes our experimental setup and the results obtained. We conclude the paper with discussion and conclusions.

2 Related Work

There exist a large number of measures for computing word-to-word relations. As already mentioned, these techniques can be broadly classified into two main categories: knowledge-based, those relying on pre-existing knowledge resources (thesauri, semantic networks, taxonomies or encyclopedia), and corpus-based, those inducing distributional properties of words from corpora.

The knowledge-based techniques use the structure of semantic networks or ontologies (e.g. is-a hierarchy in Princeton WordNet [4]) and work on distance-based measures on the network's paths [14], [15], [30]. These can further be improved by using the Information Content of the lowest common subsumer in the hierarchy and corpus statistics [12], [16], [23]. Moreover, the WordNet gloss overlap measure can be used for inferring similarity [20]. Such methods are implemented in the WordNet::Similarity package [20] and also included the SEMILAR toolkit[1] (a semantic similarity toolkit, hereinafter referred to as SEMILAR) [25].

Another category of word-to-word similarity measures rely on corpus to compute a similarity score. For example, Latent Semantic Analysis (LSA; [13]), Explicit Semantic Analysis (ESA; [6]), Global Vector (GloVE; [21]), or Latent Dirichlet Allocation (LDA; [2]) exploit the distributions of words in large collections of documents. LSA and ESA work by generating semantic models or spaces in which words are represented as vectors, the values of which being, for instance, weighted frequencies of occurrences within given documents. On the other hand, LDA models documents as topic distributions and topics as distributions over words in the vocabulary. In this case, each word can be represented as a vector encoding its contribution to the LDA generated topics. The distributed representations, such as deep learning based models, are another type of methods in this category. In distributed representations, each entity is represented by a pattern of activity distributed over many computing elements, and each computing element is involved in representing many different entities [10]. One of the popular works on distributed representations is by Mikolov et al. [17] where they used probabilistic feed forward neural network language model to estimate word representations in vector space. As such, for all these methods, the similarity between words can be, and usually is, computed in terms of cosine similarity between corresponding vectors.

Datasets to assess the performance of word-to-word similarity and relatedness methods have been developed as well. One of the most widely used, the RG dataset, consists of 65 noun pairs of words collected by Rubenstein and Goodenough [28], who had them judged by 51 human subjects in a scale from 0.0 to 4.0 according to their similarity, but ignoring any other possible semantic relationships that might appear between the terms. However, this dataset contains only nouns and is quite small to build supervised models. Another dataset which has been quite popular is WordSim-353 [5] which contains 353 word pairs, each associated with an average of 13 to 16 human judgments. In WordSim-353, there were no distinctions made be-

[1] http://semanticsimilarity.org

tween similarity and the relatedness during its annotation. Similarly, there are other datasets that do not distinguish similarity and relatedness during their annotation[2].

While there is a significant volume of work in this area in term of methods and datasets, there is not much work focusing on measuring similarity of words which is subtly different from measuring relatedness of words. This argument is in fact supported by the recent publication of the Simlex-999 corpus focusing exclusively on similarity. We focus in this paper on measuring similarity by combining knowledge-based and corpus-based measures. A closely related work is by Agirre et al. [1] where they took different approaches for measuring between similarity and relatedness. They proposed a WordNet based method and co-occurrence based methods. They conducted experiments on the RG dataset and annotated word pairs in WordSim-353 as similar or related. However, their dataset does not represent an important class of concept pair (associated but not similar entities) [9]. In our case, we combined various features including the similarity scores calculated with most recently published resources, such as Mikolov's word representations, GloVe vectors. Moreover, we did experiments with the recently published and larger dataset which Simlex-999 which consists of a set of 999 word pairs annotated with human judgments of similarity scores [9]. Each pair in Simlex-999 was explicitly judged for similarity by at least 36 people.

3 Approach for Combining Similarity Methods

Our approach is to combine different methods in a meaningful way. In order to combine methods, the overall performance of individual method should be relatively weak and at the same time capture different aspects of the data. This idea is similar to bagging where a set of weak classifiers can be shown to lead to a significantly stronger classifier by combining their outputs.

Knowledge based approaches are typically based on hand-coded relations among words e.g. synonymy, antonymy etc.; they utilize those relations which are important to define similarity but are hard to extract accurately using fully automated methods. A typical example of a lexical database that encodes explicit lexico-semantic relations among words is WordNet [4]. WordNet based methods quantify the similarity of words based on various relations among words, and heuristics and graph theories. However, their coverage is low. Furthermore, WordNet based methods require the mapping of words to concepts, i.e. a word sense disambiguation steps which could be extremely challenging to learn automatically. On the other hand, distributional representations capture various associations among words based on the principle that words that occur in similar context are related or similar.

There exist different approaches of representing the meaning of words and measuring their relationships e.g. LSA [13], ESA [6], LDA [2], [26], Neural Language Model (NLM) [17]. Even within a category, there exist diverse methods that are based on different premises. For example, Landauer et al. [13] claim that in LSA the meaning

[2] A list of some datasets can be found at http://www.cs.cmu.edu/~mfaruqui/suite.html

of words can be represented based on contextual-usage of the words through statistical computations applied to a large corpus of text. Deep Learning word embeddings are developed based on the idea that Neural Networks mimic the human mind and the connections among nodes are capable of representing complex irregularities [10], and so on.

To assess whether the combination would be helpful, we calculated similarity scores for each pair using different methods (described in Experiments and Results section) and chose the best score (i.e., score most close to the gold score) among them. By using the best scores, we obtained correlation $\rho = 0.959$ which is better than the correlation among any of the individual method's output. It indicates that each of them was performing well as compared to the other methods on particular subsets of instances and that their combination can achieve an impressive correlation with human judgments. This observation forms the basis of our approach. It is important to add that the individual methods we use are built around models developed from fairly large, albeit different, data sets. One can argue that comparing these individual methods is not fair because, as mentioned, they were trained on different data sets. However, it is not in the scope of this paper to compare individual methods but rather exploit the fact that they have different strengths and weaknesses and by combining them we hope to add up the strengths and smooth out the weakness, which is what our results indicate that we achieved.

One obvious way to combine methods would be linear regression. However, linear combination in higher dimension using kernel-based methods such as support vector machines can capture interesting relations between similarity scores obtained using individual methods. Therefore, we also experimented with support vector regression.

4 Experiments and Results

4.1 Data

We use a dataset which was recently released [9]. The dataset consists of 999 word pairs (called Simlex-999) with human generated similarity scores. The word pairs were annotated by human judges with similarity scores using a Likert-scale from 0 (no similarity) to 10 (exactly mean same thing). We were particularly interested in this corpus as the previously available benchmark datasets were not balanced (i.e., contained one category of words), contained a small number of instances, or annotated without making any distinction between relatedness and similarity. For example, RG [28] dataset contains 65 word pairs which is quite small, particularly for similarity model development. The other widely used dataset, WordSim-353 [5], contains 353 word pairs but their annotation does not differentiate similarity and relatedness. Moreover, Hill et al. [9] indicate that Simlex-999 is notably more challenging to model than the alternative datasets.

The Simlex-99 dataset contains 111 adjective pairs (A), 666 noun pairs (N), and 222 verb pairs (V). Each pair was rated by at least 36 native English speakers and the average score was assigned as final human judgment (i.e., gold score). The inter-rater agreement was calculated as the average of pairwise Spearman ρ correlations between the ratings of all respondents. Overall agreement was $\rho = 0.670$. To make the scores consistent with the system generated scores, we normalized the human-rated scores (by dividing them by 10).

4.2 Features

As mentioned in the previous section, we used similarity scores of various methods as features in regression models. We describe the individual methods below.

WN$\boldsymbol{WN_{Combined}}$: There are various similarity methods based on WordNet. We used *Lesk [20]*, *Jcn* [12], *Lin* [16], *Hso* [11], *Wup* [30], and *Path* [20]. Many of them work only on specific POS categories. For this reason, we combined their outputs and a single set of scores was generated as,

- Average of *Lesk*, *Jcn*, and *Lin* measures for verbs
- Average of *Lesk* and *Hso* measures for adjectives
- Average of *Lesk*, *Wup*, *Res*, *Jcn*, *Lin*, and *Path* measures for noun pairs

A word can have multiple senses. These methods were configured to use the first sense only.

$\mathbf{WN_{Syn}}$, $\mathbf{WN_{Ant}}$: indicates whether there is a synonymy (antonymy for WN_{Ant}) relation in WordNet between the given word pair. We only checked the synsets of given POS category.

$\mathbf{LSA_{Wiki}}$, $\mathbf{LSA_{Tasa}}$: The cosine similarity scores calculated using LSA models developed from the whole Wikipedia articles and TASA (Touchstone Applied Science Associates) corpus as described in Stefanescu et al. [27]. The Wiki LSA models were developed using an early spring 2013 Wikipedia version, containing 4,208,450 articles. TASA comprises 60,527 samples from 6,333 textbooks, works of literature, and popular works of fiction and nonfiction. These LSA models have word representations in 300 latent dimensions. Specifically, we used Wiki_NVAR_f7 and TASA_NVAR models that are available at SEMILAR website. The Wiki_NVAR_f7 model was developed considering only the lemmas of content words occurring at least 7 times. The TASA_NVAR is similar to Wiki_NVAR_f7, but with no frequency threshold.

CRDE: Similarity using word vectors generated by applying Deep Learning technique. We used 200-dimensional word representation model developed by Turian et al. [29][3]. Word embeddings were induced using neural language model. They used RCV1 corpus which has 37 million words in 1.3 million sentences after cleaning.

[3] http://metaoptimize.com/projects/wordreprs/

UMBC: Similarity calculated using UMBC system [8][4] without using POS information. This system calculates similarity using HAL (Hyperspace Analog to Language) [3] model developed using Wikipedia and the similarity score is boosted using WordNet knowledge.

ESA: Score calculated using Explicit Semantic Analysis [6]. We used web service of ESAlib[5] to calculate these scores. However, we got valid numeric scores for 916 word pairs only. Due to this reason, we did not use this feature for the regression but we present the correlation score calculated ignoring the others.

MK-NLM: Neuro probabilistic language model based word representations developed by Mikolov et al. [17] [6]. We used 300-dimensional word vectors developed by training distributed representations of words with the Skip-gram model on part of Google News dataset (about 100 billion words).

GloVe: Score calculated using word representation model proposed by Pennington et al. [21][7] and trained on 42 billion Common Crawl words. We used 300-dimensional word representation model.

LDA*wiki*: Score calculated using a Latent Dirichlet Allocation (LDA) model generated from whole Wikipedia articles (documents with less than 500 words, stopwords and words that occur in less than 500 documents were removed) [31]. Then a 300-topic LDA model was developed (using 270,290 documents and the vocabulary of 59,136 words). The word-topic association vector was used for similarity calculation. The model was developed using JGibbsLDA[8] in high performance computing machines.

4.3 Experiments

First, we calculated similarity scores using different methods and measured their correlations (r) with the human judgments (see Table 1). For WordNet based methods, similarity scores were calculated for adjectives, nouns, and verbs separately (scores were calculated only if the method supported that POS category). After that, a single set of scores (i.e., $\text{WN}_{\text{Combined}}$) was generated using similarity scores from all WordNet based methods as described before. We also checked whether the words were synonyms or antonyms. We used WordNet 3.0 for all of these operations. For vector based methods, we calculated cosine similarity scores using the word representation vectors. In this case, we did not use POS information of the word pairs as each of the models we used has single representation for each word. For missing words (10 words in the case of the LSA Wiki model, and 6 words in the LSA TASA model), we obtained

[4] http://swoogle.umbc.edu/SimService/api.html

[5] http://ticcky.github.io/esalib/

[6] http://code.google.com/p/word2vec/

[7] http://www-nlp.stanford.edu/projects/glove/

[8] http://jgibblda.sourceforge.net/

synonyms from WordNet and replaced the original word by the vector of one of the synonyms that was found in the models.

Second, we applied Linear Regression (LR) and Support Vector Regression (SVR) to combine the results obtained from different methods - all or subsets of methods (see Table 2 for the results). The Weka tool was used for both linear regression and support vector regression (using LibSVM[9]). For evaluation purpose, we applied 10-fold cross validation method which gives a very good estimate of the performance of the model.

4.4 Results

Table 1 presents correlations (Pearson and Spearman's rank correlation coefficients are separated by /; first one is Pearson correlation) of similarity scores produced using different methods with human judgments. The rows are numbered and the row number is used to refer to the result of that particular method.

Table 1: Correlation (Pearson and Spearman correlation coefficients are separated by /) of similarity scores generated using different methods and human judgment in Simlex-999 dataset.

ID	Method	All	Adjective	Noun	Verb
1	Lesk	0.347/0.404	0.418/0.422	0.373/0.448	0.301/0.315
2	Hso	0.324/0.330	0.264/0.236	0.421/0.460	0.223/0.204
3	Wup	-	-	0.471/0.489	0.246/0.180
4	Res	-	-	0.454/0.443	0.245/0.219
5	Jcn	-	-	0.462/0.451	0.279/0.121
6	Lin	-	-	0.462/0.452	0.289/0.252
7	Path	-	-	0.513/0.507	0.216/0.031
8	Lch	-	-	0.534/0.506	0.109/0.031
9	$\text{WN}_{\text{Combined}}$	0.362/0.322	0.418/0.422	0.535/0.507	0.327/0.285
10	LSA_{Tasa}	0.251/0.271	0.015/0.042	0.332/0.343	0.221/0.214
11	LSA_{Wiki}	0.277/0.273	0.250/0.285	0.325/0.318	0.153/0.154
12	CRDE	0.144/0.157	0.198/0.190	0.136/0.161	0.129/0.119
13	GloVe	0.400/0.373	0.550/0.574	0.433/0.404	0.194/0.177
14	Mk-NLM	0.453/0.442	0.597/0.592	0.459/0.452	0.348/0.321
15	LDA_{wiki}	0.228/0.288	0.321/0.334	0.240/0.325	0.181/0.173
16	UMBC	**0.557/0.558**	**0.624/0.613**	**0.599/0.591**	**0.522/0.490**
17	ESA	0.145/0.271	-	-	-
18	Avg (9-16)	0.488/0.491	0.536/0.556	0.522/0.511	0.427/0.393

[9] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

In addition to the results on overall data, the Table 1 presents the results grouped by adjective, noun, and verb. These results are plotted in a histogram as shown in Figure 2. The UMBC system performed the best in the overall category as well as in the individual category followed by Mk-NLM and GloVe based methods. The noun similarity using WordNet based method is also high. The similarity of adjectives and nouns are better correlated with human judgments than that of verbs. However, the performance of LSA$_{\text{Tasa}}$ on adjectives is very low. It may be due to low presence of adjectives in TASA corpus which contains academic texts.

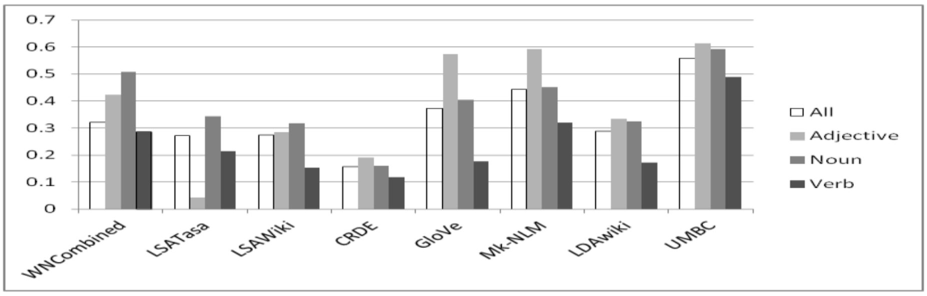

Figure 1: Graph showing the performance (ρ) of different methods grouped by POS category.

As discussed in Section 3, we chose the best score (i.e., score close to the gold score) for each word pair among the WN$_{\text{Combined}}$, WN$_{\text{syn}}$, WN$_{\text{ant}}$, LSA$_{\text{Tasa}}$, LSA$_{\text{Wiki}}$, CRDE, UMBC, GloVe, LDA$_{\text{Wiki}}$ and Mk-NLM scores, and calculated the correlation with human judgments. The correlation (ρ) was 0.959. This indicated the huge potential in improving the similarity calculation by taping the power of individual method.

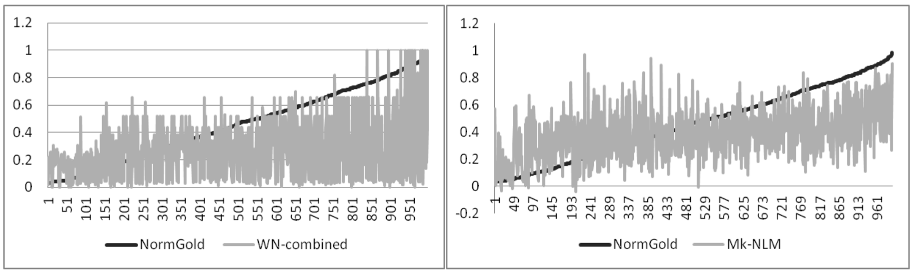

Figure 2: The graphs showing the scores predicted by WN$_{\text{combined}}$ and Mk-NLM and the normalized gold score (instances sorted by the gold score).

Moreover, the graphs in Figure 2 show the output of representative methods WN$_{\text{combined}}$ and Mk-NLM. This also illustrates that combining the methods could improve performance overall, which is what we present next.

The results produced by individual methods were combined by applying linear regression (LR), and support vector regression implementation in LibSVM (SVR). The results are presented in Table 2. The average inter-annotator agreement of SimLex-999 is ρ = 0.670 and the best reported score by Hill et al. [9] is ρ = 0.446 where they used dependency based word embeddings.

Table 2: Correlation (Pearson correlation and Spearman's Rank correlation separated by /) and Root Mean Square Error (RMSE) obtained after combining different methods. The default kernel function used in support vector regression was Radial Basis Function (RBF).

Regression method: features	Correlation	RMSE
Inter-annotator agreement (Hill et. al., 2014)	-/0.670	
Hill et al. (2014)	-/0.446	
LR1: WN_{Ant},$WN_{Syn,}$ 9	0.473/0.429	0.192
LR2: 10-15	0.452/0.436	0.195
LR3: WN_{Ant},$WN_{Syn,}$ 9-15	0.598/0.587	0.175
LR4: WN_{Ant},WN_{Syn}, 9-16	**0.634/0.631**	0.167
SVR1: WN_{Ant},$WN_{Syn,}$ 9	0.495/0.422	0.186
SVR2: 10-15	0.480/0.440	0.189
SVR3: WN_{Ant},$WN_{Syn,}$ 9-15	0.623/0.599	0.167
SVR4: WN_{Ant},WN_{Syn}, 9-16	**0.658/0.642**	0.159
SVR5: (Linear kernel): WN_{Ant},WN_{Syn}, 9-16	0.634/0.626	0.157
SVR6: (Polynomial kernel): WN_{Ant},WN_{Syn}, 9-16	0.536/0.607	0.187
SVR7: (Sigmoid kernel): WN_{Ant},WN_{Syn}, 9-16	0.589/0.604	0.176

We run regressions with all possible combinations of features. However, instead of showing all combinations, we present results generated with four groups of features: WordNet based methods (WN_{Ant}, $WN_{Syn,}$ 9), corpus based methods (10-15), all features except UMBC, and all features. The best result ($\rho = 0.642$) was obtained when the all features were used in support vector regression with Radial Basis Function (RBF) kernel. Moreover, we changed the kernel function in support vector regression and run with the best performing feature set when RBF kernel was used. But it did not improve the result. The results show that the best performance is obtained in both linear and support vector regressions when features from both knowledge-based and corpus-based category were used. The best result obtained using support vector regression (SVR4; $\rho = 0.642$) is significantly better than the individual performance reported in Table 1 where maximum correlation was 0.558 ($P < 0.001$).

5 Conclusion

Assessing the similarity of text is a challenging task. One might argue that similarity between two words in isolation cannot be quantified and should be defined in context. However, when humans need to judge the similarity of two things, they consider various factors and make a holistic judgment which is what the combination of different similarity methods are probably capturing.

To conclude, we presented a way of measuring the similarity of words by combining different methods. Particularly, we applied regressions to combine WordNet and different vector based methods. We found that the results produced by regressions better aligned with the human judgment compared to the individual automated methods. The best result ($\rho = 0.642$) was obtained when features including knowledge-based and corpus-based similarity scores were used in support vector regression with

Radial Basis Function (RBF) kernel. This is significantly ($P < 0.001$) better than the individual performance reported in Table 1 where maximum correlation was 0.557 and the best result obtained using the linear regression. Our similarity model's performance on Simlex-999 has reached close to the average agreement of human annotators. In the future, we would like to work on measuring semantic similarity in context and apply to measure semantic similarity of bigger texts (e.g., sentence level similarity).

Acknowledgements

This research was partially sponsored by The University of Memphis and the Institute for Education Sciences (IES) under award R305A100875 to Dr. Vasile Rus.

References

1. Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., & Soroa, A. (2009, May). A study on similarity and relatedness using distributional and WordNet-based approaches. In *Proceedings of HLT: The 2009 Annual Conference of NAACL* (pp. 19-27). Association for Computational Linguistics.
2. Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *The Journal of machine Learning research*, *3*, 993-1022.
3. Burgess, C., & Lund, K. (1995). Hyperspace analog to language (hal): A general model of semantic representation. In *Proceedings of the annual meeting of the Psychonomic Society* (Vol. 12, pp. 177-210).
4. Fellbaum, C. (1998). *WordNet*. Blackwell Publishing Ltd.
5. Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., & Ruppin, E. (2001, April). Placing search in context: The concept revisited. In*Proceedings of the 10th international conference on World Wide Web* (pp. 406-414). ACM.
6. Gabrilovich, E., & Markovitch, S. (2007, January). Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis. In *IJCAI* (Vol. 7, pp. 1606-1611).
7. Graesser, A. C., Penumatsa, P., Ventura, M., Cai, Z., & Hu, X. (2007). Using LSA in AutoTutor: Learning through mixed initiative dialogue in natural language. *Handbook of latent semantic analysis*, 243-262.
8. Han, L., Kashyap, A., Finin, T., Mayfield, J., & Weese, J. (2013, June). UMBC EBIQUITY-CORE: Semantic textual similarity systems. In *Proceedings of the Second Joint Conference on Lexical and Computational Semantics* (Vol. 1, pp. 44-52).
9. Hill, F., Reichart, R., & Korhonen, A. (2014). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *arXiv preprint arXiv:1408.3456*.
10. Hinton, G. E. (1984). Distributed representations.
11. Hirst, G., & St-Onge, D. (1998). Lexical chains as representations of context for the detection and correction of malapropisms. *WordNet: An electronic lexical database*, *305*, 305-332.
12. Jiang, J. J., & Conrath, D. W. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. *arXiv preprint cmp-lg/9709008*.
13. Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse processes*, *25*(2-3), 259-284.

14. Leacock, C., & Chodorow, M. (1998). Combining local context and WordNet similarity for word sense identification. *WordNet: An electronic lexical database*, *49*(2), 265-283.
15. Lee, J. H., Kim, M. H., and Lee, Y. J. (1989). Information retrieval based on conceptual distance in is-a hierarchies. *Journal of Documentation 42*(2):188-207.
16. Lin, D. (1998, July). An information-theoretic definition of similarity. In *ICML* (Vol. 98, pp. 296-304).
17. Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In*Advances in Neural Information Processing Systems* (pp. 3111-3119).
18. Mohler, M., & Mihalcea, R. (2009, March). Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 567-575). Association for Computational Linguistics.
19. Patwardhan, S., Banerjee, S., & Pedersen, T. (2003). Using measures of semantic relatedness for word sense disambiguation. In *Computational linguistics and intelligent text processing* (pp. 241-257). Springer Berlin Heidelberg.
20. Pedersen, T., Patwardhan, S., & Michelizzi, J. (2004, May). WordNet:: Similarity: measuring the relatedness of concepts. In *Demonstration Papers at HLT-NAACL 2004* (pp. 38-41). Association for Computational Linguistics.
21. Pennington J, Socher R., & Manning, C. Glove: Global vectors for word representation.
22. Potthast, M., Gollub, T., Hagen, M., Kiesel, J., Michel, M., Oberländer, A., ... & Stein, B. (2012, September). Overview of the 4th International Competition on Plagiarism Detection. In *CLEF (Online Working Notes/Labs/Workshop)*.
23. Resnik, P. (1995). Using information content to evaluate semantic similarity in taxonomy. *arXiv preprint cmp-lg/9511007*.
24. Rus, V., & Lintean, M. (2012, June). A comparison of greedy and optimal assessment of natural language student input using word-to-word similarity metrics. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP* (pp. 157-162). Association for Computational Linguistics.
25. Rus, V., Lintean, M. C., Banjade, R., Niraula, N. B., & Stefanescu, D. (2013, August). SEMILAR: The Semantic Similarity Toolkit. In *ACL (Conference System Demonstrations)* (pp. 163-168).
26. Rus, V., Niraula, N., & Banjade, R. (2013). Similarity measures based on latent dirichlet allocation. In *Computational Linguistics and Intelligent Text Processing* (pp. 459-470). Springer Berlin Heidelberg.
27. Ştefănescu, D., Banjade, R. Rus, V.: Latent Semantic Analysis Models on Wikipedia and TASA, LREC (2014).
28. Stefanescu, D., Rus, V., Niraula, N. B., & Banjade, R. (2014, March). Combining Knowledge and Corpus-based Measures for Word-to-Word Similarity. In *The Twenty-Seventh International Flairs Conference*.
29. Turian, J., Ratinov, L., & Bengio, Y. (2010, July). Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics* (pp. 384-394). Association for Computational Linguistics.
30. Wu, Z., and Palmer, M. 1994. Verb semantics and lexical selection. In *32nd Annual Meeting of the Association for Computational Linguistics*, 133–138.
31. Niraula, N. B., Gautam, D., Banjade, R., Maharjan, N., & Rus, V. (2015). Combining Word Representations for Measuring Word Relatedness and Similarity. In *The Proceedings of 28th International FLAIRS Conference*.