DeepPurple: Estimating Sentence Semantic Similarity using N-gram Regression Models and Web Snippets

Nikos Malandrakis, Elias Iosif, Alexandros Potamianos
Department of ECE, Technical University of Crete, 73100 Chania, Greece
[nmalandrakis,iosife,potam]@telecom.tuc.gr

Abstract

We estimate the semantic similarity between two sentences using regression models with features: 1) n-gram hit rates (lexical matches) between sentences, 2) lexical semantic similarity between non-matching words, and 3) sentence length. Lexical semantic similarity is computed via co-occurrence counts on a corpus harvested from the web using a modified mutual information metric. State-of-the-art results are obtained for semantic similarity computation at the word level, however, the fusion of this information at the sentence level provides only moderate improvement on Task 6 of SemEval'12. Despite the simple features used, regression models provide good performance, especially for shorter sentences, reaching correlation of 0.62 on the SemEval test set.

1 Introduction

Recently, there has been significant research activity on the area of semantic similarity estimation motivated both by abundance of relevant web data and linguistic resources for this task. Algorithms for computing semantic textual similarity (STS) are relevant for a variety of applications, including information extraction (Szpektor and Dagan, 2008), question answering (Harabagiu and Hickl, 2006) and machine translation (Mirkin et al., 2009). Word- or term-level STS (a special case of sentence level STS) has also been successfully applied to the problem of grammar induction (Meng and Siu, 2002) and affective text categorization (Malandrakis et al., 2011). In this work, we built on previous research on word-level semantic similarity estimation to design and implement a system for sentence-level STS for Task6 of the SemEval'12 campaign.

Semantic similarity between words can be regarded as the graded semantic equivalence at the lexeme level and is tightly related with the tasks of word sense discovery and disambiguation (Agirre and Edmonds, 2007). Metrics of word semantic similarity can be divided into: (i) knowledge-based metrics (Miller, 1990; Budanitsky and Hirst, 2006) and (ii) corpus-based metrics (Baroni and Lenci, 2010; Iosif and Potamianos, 2010).

When more complex structures, such as phrases and sentences, are considered, it is much harder to estimate semantic equivalence due to the non-compositional nature of sentence-level semantics and the exponential explosion of possible interpretations. STS is closely related to the problems of paraphrasing, which is bidirectional and based on semantic equivalence (Madnani and Dorr, 2010) and textual entailment, which is directional and based on relations between semantics (Dagan et al., 2006). Related methods incorporate measurements of similarity at various levels: lexical (Malakasiotis and Androutsopoulos, 2007), syntactic (Malakasiotis, 2009; Zanzotto et al., 2009), and semantic (Rinaldi et al., 2003; Bos and Markert, 2005). Measures from machine translation evaluation are often used to evaluate lexical level approaches (Finch et al., 2005; Perez and Alfonseca, 2005), including BLEU (Papineni et al., 2002), a metric based on word n-gram hit rates.

Motivated by BLEU, we use n-gram hit rates and word-level semantic similarity scores as features in

*First Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 565–570,
Montréal, Canada, June 7-8, 2012.

a linear regression model to estimate sentence level semantic similarity. We also propose sigmoid scaling of similarity scores and sentence-length dependent modeling. The models are evaluated on the SemEval'12 sentence similarity task.

2 Semantic similarity between words

In this section, two different metrics of word similarity are presented. The first is a language-agnostic, corpus-based metric requiring no knowledge resources, while the second metric relies on WordNet.

Corpus-based metric: Given a corpus, the semantic similarity between two words, w_i and w_j, is estimated as their pointwise mutual information (Church and Hanks, 1990): $I(i,j) = \log \frac{\hat{p}(i,j)}{\hat{p}(i)\hat{p}(j)}$, where $\hat{p}(i)$ and $\hat{p}(j)$ are the occurrence probabilities of w_i and w_j, respectively, while the probability of their co-occurrence is denoted by $\hat{p}(i,j)$. These probabilities are computed according to maximum likelihood estimation. The assumption of this metric is that co-occurrence implies semantic similarity.

During the past decade the web has been used for estimating the required probabilities (Turney, 2001; Bollegala et al., 2007), by querying web search engines and retrieving the number of hits required to estimate the frequency of individual words and their co-occurrence. However, these approaches have failed to obtain state-of-the-art results (Bollegala et al., 2007), unless "expensive" conjunctive AND queries are used for harvesting a corpus and then using this corpus to estimate similarity scores (Iosif and Potamianos, 2010).

Recently, a scalable approach[1] for harvesting a corpus has been proposed where web snippets are downloaded using individual queries for each word (Iosif and Potamianos, 2012b). Semantic similarity can then be estimated using the $I(i,j)$ metric and *within-snippet word co-occurrence frequencies*. Under the maximum sense similarity assumption (Resnik, 1995), it is relatively easy to show that a (more) lexically-balanced corpus[2] (as the one created above) can significantly reduce the semantic similarity estimation error of the mutual information metric $I(i,j)$. This is also experimentally verified in (Iosif and Potamianos, 2012c).

In addition, one can modify the mutual information metric to further reduce estimation error (for the theoretical foundation behind this see (Iosif and Potamianos, 2012a)). Specifically, one may introduce exponential weights α in order to reduce the contribution of $p(i)$ and $p(j)$ in the similarity metric. The modified metric $I_a(i,j)$, is defined as:

$$I_a(i,j) = \frac{1}{2}\left[\log \frac{\hat{p}(i,j)}{\hat{p}^{\alpha}(i)\hat{p}(j)} + \log \frac{\hat{p}(i,j)}{\hat{p}(i)\hat{p}^{\alpha}(j)}\right]. \quad (1)$$

The weight α was estimated on the corpus of (Iosif and Potamianos, 2012b) in order to maximize word sense coverage in the semantic neighborhood of each word. The $I_a(i,j)$ metric using the estimated value of $\alpha = 0.8$ was shown to significantly outperform $I(i,j)$ and to achieve state-of-the-art results on standard semantic similarity datasets (Rubenstein and Goodenough, 1965; Miller and Charles, 1998; Finkelstein et al., 2002). For more details see (Iosif and Potamianos, 2012a).

WordNet-based metrics: For comparison purposes, we evaluated various similarity metrics on the task of word similarity computation on three standard datasets (same as above). The best results were obtained by the Vector metric (Patwardhan and Pedersen, 2006), which exploits the lexical information that is included in the WordNet glosses. This metric was incorporated to our proposed approach. All metrics were computed using the WordNet::Similarity module (Pedersen, 2005).

3 N-gram Regression Models

Inspired by BLEU (Papineni et al., 2002), we propose a simple regression model that combines evidence from two sources: number of n-gram matches and degree of similarity between non-matching words between two sentences. In order to incorporate a word semantic similarity metric into BLEU, we apply the following two-pass process: first lexical hits are identified and counted, and then the semantic similarity between n-grams not matched dur-

[1] The scalability of this approach has been demonstrated in (Iosif and Potamianos, 2012b) for a 10K vocabulary, here we extend it to the full 60K WordNet vocabulary.

[2] According to this assumption the semantic similarity of two words can be estimated as the minimum pairwise similarity of their senses. The gist of the argument is that although words often co-occur with their closest senses, word occurrences correspond to all senses, i.e., the denominator of $I(i,j)$ is overestimated causing large underestimation error for similarities between polysemous words.

ing the first pass is estimated. All word similarity metrics used are peak-to-peak normalized in the [0,1] range, so they serve as a "degree-of-match". The semantic similarity scores from word pairs are summed together (just like n-gram hits) to obtain a BLEU-like semantic similarity score. The main problem here is one of alignment, since we need to compare each non-matched n-gram from the hypothesis with an n-gram from the reference. We use a simple approach: we iterate on the hypothesis n-grams, left-to-right, and compare each with the *most similar* non-matched n-gram in the reference. This modification to BLEU is only applied to 1-grams, since semantic similarity scores for bigrams (or higher) were not available.

Thus, our list of features are the hit rates obtained by BLEU (for 1-, 2-, 3-, 4-grams) and the total semantic similarity (SS) score for 1-grams[3]. These features are then combined using a multiple linear regression model:

$$\hat{D}_L = a_0 + \sum_{n=1}^{4} a_n \, B_n + a_5 \, M_1, \qquad (2)$$

where $\hat{D}_L$ is the estimated similarity, B_n is the BLEU hit rate for n-grams, M_1 is the total semantic similarity score (SS) for non-matching 1-grams and a_n are the trainable parameters of the model.

Motivated by evidence of cognitive scaling of semantic similarity scores (Iosif and Potamianos, 2010), we propose the use of a sigmoid function to scale D_L sentence similarities. We have also observed in the SemEval data that the way humans rate sentence similarity is very much dependent on sentence length[4]. To capture the effect of length and cognitive scaling we propose next two modifications to the linear regression model. The sigmoid fusion scheme is described by the following equation:

$$\hat{D}_S = a_6 \hat{D}_L + a_7 \hat{D}_L \left[1 + \exp\left(\frac{a_8 - l}{a_9} \right) \right]^{-1}, \qquad (3)$$

where we assume that sentence length l (average length for each sentence pair, in words) acts as a scaling factor for the linearly estimated similarity.

The hierarchical fusion scheme is actually a collection of (overlapping) linear regression models, each matching a range of sentence lengths. For example, the first model D_{L1} is trained with sentences with length up to l_1, i.e., $l \leq l_1$, the second model D_{L2} up to length l_2 etc. During testing, sentences with length $l \in [1, l_1]$ are decoded with D_{L1}, sentences with length $l \in (l_1, l_2]$ with model D_{L2} etc. Each of these partial models is a linear fusion model as shown in (2). In this work, we use four models with $l_1 = 10$, $l_2 = 20$, $l_3 = 30$, $l_4 = \infty$.

4 Experimental Procedure and Results

Initially all sentences are pre-processed by the CoreNLP (Finkel et al., 2005; Toutanova et al., 2003) suite of tools, a process that includes named entity recognition, normalization, part of speech tagging, lemmatization and stemming. The exact type of pre-processing used depends on the metric used. For the plain lexical BLEU, we use lemmatization, stemming (of lemmas) and remove all non-content words, keeping only nouns, adjectives, verbs and adverbs. For computing semantic similarity scores, we don't use stemming and keep only noun words, since we only have similarities between non-noun words. For the computation of semantic similarity we have created a dictionary containing all the single-word nouns included in WordNet (approx. 60K) and then downloaded snippets of the 500 top-ranked documents for each word by formulating single-word queries and submitting them to the Yahoo! search engine.

Next, results are reported in terms of correlation between the automatically computed scores and the ground truth, for each of the corpora in Task 6 of SemEval'12 (paraphrase, video, europarl, WordNet, news). Overall correlation ("Ovrl") computed on the join of the dataset, as well as, average ("Mean") correlation across all task is also reported. Training is performed on a subset of the first three corpora and testing on all five corpora.

Baseline BLEU: The first set of results in Table 1, shows the correlation performance of the plain BLEU hit rates (per training data set and overall/average). The best performing hit rate is the one

[3]Note that the features are computed twice on each sentence in a forward and backward fashion (where the word order is reversed), and then averaged between the two runs.

[4]We speculate that shorter sentences are mostly compared at the lexical level using the short-term memory language buffers, while longer sentences tend to be compared at a higher cognitive level, where the non-compositional nature of sentence semantics dominate.

calculated using unigrams.

Table 1: Correlation performance of BLEU hit rates.

	par	vid	euro	Mean	Ovrl
BLEU 1-grams	**0.62**	**0.67**	**0.49**	**0.59**	**0.57**
BLEU 2-grams	0.40	0.39	0.37	0.39	0.34
BLEU 3-grams	0.32	0.36	0.30	0.33	0.33
BLEU 4-grams	0.26	0.25	0.24	0.25	0.28

Semantic Similarity BLEU (Purple): The performance of the modified version of BLEU that incorporates various word-level similarity metrics is shown in Table 2. Here the BLEU hits (exact matches) are summed together with the normalized similarity scores (approximate matches) to obtain a single B_1+M_1 (Purple) score[5]. As we can see, there are definite benefits to using the modified version, particularly with regards to mean correlation. Overall the best performers, when taking into account both mean and overall correlation, are the WordNet-based and I_a metrics, with the I_a metric winning by a slight margin, earning a place in the final models.

Table 2: Correlation performance of 1-gram BLEU scores with semantic similarity metrics (nouns-only).

	par	vid	euro	Mean	Ovrl
BLEU	0.54	0.60	0.39	0.51	0.58
SS-BLEU WordNet	0.56	**0.64**	**0.41**	**0.54**	0.58
SS-BLEU $I(i,j)$	0.56	0.63	0.39	0.53	**0.59**
SS-BLEU $I_a(i,j)$	**0.57**	**0.64**	0.40	**0.54**	0.58

Regression models (DeepPurple): Next, the performance of the various regression models (fusion schemes) is investigated. Each regression model is evaluated by performing 10-fold cross-validation on the SemEval training set. Correlation performance is shown in Table 3 both with and without semantic similarity. The baseline in this case is the Purple metric (corresponding to no fusion). Clearly the use of regression models significantly improves performance compared to the 1-gram BLEU and Purple baselines for almost all datasets, and especially for the combined dataset (overall). Among the fusion schemes, the hierarchical models perform the best. Following fusion, the performance gain from incorporating semantic similarity (SS) is much smaller. Finally, in Table 4, correlation performance of our submissions on the official SemEval test set is shown. The overall correlation performance of the Hierarchical model ranks somewhere in the middle (43rd out of 89 systems), while the mean correlation (weighted by number of samples per set) is notably better: 23rd out of 89. Comparing the individual dataset results, our systems underperform for the two datasets that originate from the machine translation (MT) literature (and contain longer sentences), while we achieve good results for the rest (19th for paraphrase, 37th for video and 29th for WN).

[5]It should be stressed that the plain BLEU unigram scores shown in this table are not comparable to those in Table 1, since here scores are calculated over only the nouns of each sentence.

Table 3: Correlation performance of regression model with (SS) and without semantic similarities on the training set (using 10-fold cross-validation).

	par	vid	euro	Mean	Ovrl
None (SS-BLEU I_a)	0.57	0.64	0.40	0.54	0.58
Linear ($\hat{D}_L, a_5=0$)	0.62	0.72	0.47	0.60	0.66
Sigmoid ($\hat{D}_S, a_5=0$)	0.64	0.73	0.42	0.60	0.73
Hierarchical	0.64	**0.74**	**0.48**	**0.62**	0.73
SS-Linear ($\hat{D}_L$)	0.64	0.73	0.47	0.61	0.66
SS-Sigmoid ($\hat{D}_S$)	**0.65**	**0.74**	0.42	0.60	**0.74**
SS-Hierarchical	**0.65**	**0.74**	**0.48**	**0.62**	0.73

Table 4: Correlation performance on test set.

	par	vid	euro	WN	news	Mean	Ovrl
None	0.50	0.71	**0.44**	0.49	0.24	0.51	0.49
Sigm.	0.60	0.76	0.26	0.60	0.34	0.56	0.55
Hier.	**0.60**	**0.77**	0.43	**0.65**	**0.37**	**0.60**	**0.62**

5 Conclusions

We have shown that: 1) a regression model that combines counts of exact and approximate n-gram matches provides good performance for sentence similarity computation (especially for short and medium length sentences), 2) the non-linear scaling of hit-rates with respect to sentence length improves performance, 3) incorporating word semantic similarity scores (soft-match) into the model can improve performance, and 4) web snippet corpus creation and the modified mutual information metric is a language agnostic approach that can (at least) match semantic similarity performance of the best resource-based metrics for this task. Future work, should involve the extension of this approach to model larger lexical chunks, the incorporation of compositional models of meaning, and in general the phrase-level modeling of semantic similarity, in order to compete with MT-based systems trained on massive external parallel corpora.

References

E. Agirre and P. Edmonds, editors. 2007. *Word Sense Disambiguation: Algorithms and Applications*. Springer.

M. Baroni and A. Lenci. 2010. Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

D. Bollegala, Y. Matsuo, and M. Ishizuka. 2007. Measuring semantic similarity between words using web search engines. In *Proc. of International Conference on World Wide Web*, pages 757–766.

J. Bos and K. Markert. 2005. Recognising textual entailment with logical inference. In *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, page 628635.

A. Budanitsky and G. Hirst. 2006. Evaluating WordNet-based measures of semantic distance. *Computational Linguistics*, 32:13–47.

K. W. Church and P. Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29.

I. Dagan, O. Glickman, and B. Magnini. 2006. The pascal recognising textual entailment challenge. In Joaquin Quionero-Candela, Ido Dagan, Bernardo Magnini, and Florence dAlch Buc, editors, *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, volume 3944 of *Lecture Notes in Computer Science*, pages 177–190. Springer Berlin / Heidelberg.

A. Finch, S. Y. Hwang, and E. Sumita. 2005. Using machine translation evaluation techniques to determine sentence-level semantic equivalence. In *Proceedings of the 3rd International Workshop on Paraphrasing*, page 1724.

J. R. Finkel, T. Grenager, and C. D. Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370.

L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin. 2002. Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20(1):116–131.

S. Harabagiu and A. Hickl. 2006. Methods for Using Textual Entailment in Open-Domain Question Answering. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 905–912.

E. Iosif and A. Potamianos. 2010. Unsupervised semantic similarity computation between terms using web documents. *IEEE Transactions on Knowledge and Data Engineering*, 22(11):1637–1647.

E. Iosif and A. Potamianos. 2012a. Minimum error semantic similarity using text corpora constructed from web queries. *IEEE Transactions on Knowledge and Data Engineering* (submitted to).

E. Iosif and A. Potamianos. 2012b. Semsim: Resources for normalized semantic similarity computation using lexical networks. *Proc. of Eighth International Conference on Language Resources and Evaluation* (to appear).

E. Iosif and A. Potamianos. 2012c. Similarity computation using semantic networks created from web-harvested data. *Natural Language Engineering* (submitted to).

N. Madnani and B. J. Dorr. 2010. Generating phrasal and sentential paraphrases: A survey of data-driven methods. *Computational Linguistics*, 36(3):341387.

P. Malakasiotis and I. Androutsopoulos. 2007. Learning textual entailment using svms and string similarity measures. In *Proceedings of of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 42–47.

P. Malakasiotis. 2009. Paraphrase recognition using machine learning to combine similarity measures. In *Proceedings of the 47th Annual Meeting of ACL and the 4th Int. Joint Conference on Natural Language Processing of AFNLP*, pages 42–47.

N. Malandrakis, A. Potamianos, E. Iosif, and S. Narayanan. 2011. Kernel models for affective lexicon creation. In *Proc. Interspeech*, pages 2977–2980.

H. Meng and K.-C. Siu. 2002. Semi-automatic acquisition of semantic structures for understanding domain-specific natural language queries. *IEEE Transactions on Knowledge and Data Engineering*, 14(1):172–181.

G. Miller and W. Charles. 1998. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28.

G. Miller. 1990. Wordnet: An on-line lexical database. *International Journal of Lexicography*, 3(4):235–312.

S. Mirkin, L. Specia, N. Cancedda, I. Dagan, M. Dymetman, and S. Idan. 2009. Source-language entailment modeling for translating unknown terms. In *Proceedings of the 47th Annual Meeting of ACL and the 4th Int. Joint Conference on Natural Language Processing of AFNLP*, pages 791–799.

K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318.

S. Patwardhan and T. Pedersen. 2006. Using WordNet-based context vectors to estimate the semantic relatedness of concepts. In *Proc. of the EACL Workshop on Making Sense of Sense: Bringing Computational Linguistics and Psycholinguistics Together*, pages 1–8.

T. Pedersen. 2005. WordNet::Similarity. `http://search.cpan.org/dist/WordNet-Similarity/`.

D. Perez and E. Alfonseca. 2005. Application of the bleu algorithm for recognizing textual entailments. In *Proceedings of the PASCAL Challenges Worshop on Recognising Textual Entailment*.

P. Resnik. 1995. Using information content to evaluate semantic similarity in a taxanomy. In *Proc. of International Joint Conference for Artificial Intelligence*, pages 448–453.

F. Rinaldi, J. Dowdall, K. Kaljurand, M. Hess, and D. Molla. 2003. Exploiting paraphrases in a question answering system. In *Proceedings of the 2nd International Workshop on Paraphrasing*, pages 25–32.

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

I. Szpektor and I. Dagan. 2008. Learning entailment rules for unary templates. In *Proceedings of the 22nd International Conference on Computational Linguistics*, pages 849–856.

K. Toutanova, D. Klein, C. D. Manning, and Y. Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 173–180.

P. D. Turney. 2001. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In *Proc. of the European Conference on Machine Learning*, pages 491–502.

F. Zanzotto, M. Pennacchiotti, and A. Moschitti. 2009. A machine-learning approach to textual entailment recognition. *Natural Language Engineering*, 15(4):551582.