Measuring Semantic Similarity in Short Texts through Greedy Pairing and Word Semantics

Mihai Lintean and **Vasile Rus**
Department of Computer Science
The University of Memphis
Memphis, TN 38152, USA
{*mclinten,vrus*}*@memphis.edu*

Abstract

We propose in this paper a greedy method to the problem of measuring semantic similarity between short texts. Our method is based on the principle of compositionality which states that the overall meaning of a sentence can be captured by summing up the meaning of its parts, i.e. the meanings of words in our case. Based on this principle, we extend word-to-word semantic similarity metrics to quantify the semantic similarity at sentence level. We report results using several word-to-word semantic similarity metrics, based on word knowledge or vectorial representations of meaning. Our approach performs better than similar approaches on the tasks of paraphrase identification and recognizing textual entailment, which are two illustrative semantic similarity tasks. We also report the role of word weighting and of function words on the performance of the proposed method.

Introduction

We address in this paper the problem of assessing the semantic similarity between texts. That is, given two texts the challenge is to quantify how similar the texts are. Alternatively, the challenge could be about deciding whether a qualitative semantic relation exists between the two texts. Examples of qualitative semantic relations between two texts are the paraphrase relation, when the two texts have the same meaning, and the entailment relation, when one text logically infers the other.

Our focus here is on assessing, both quantitatively (i.e. computing normalized metric scores) and qualitatively (i.e. detecting relation types), the semantic similarity of short texts. An example of a pair of short texts whose semantic similarity must be computed is given below (taken from the Microsoft Research Paraphrase Corpus (MSRP; (Dolan, Quirk, and Brockett 2004)). The example is simple but interesting enough, as the texts may or may not have the same meaning, depending on the context or the reader's background. We use this example later in the paper.

Text A: *The procedure is generally performed in the second or third trimester.*

Text B: *The technique is used during the second and, occasionally, third trimester of pregnancy.*

The task of semantic similarity of texts is a central problem in Natural Language Processing (NLP) due to its importance to a variety of applications ranging from webpage retrieval (Park, Ra, and Jang 2005), question answering (Ibrahim, Katz, and Lin 2003), text classification (Lodhi et al. 2002) and clustering, to natural language generation (Iordanskaja, Kittredge, and Polguere 1991) and conversational agents in intelligent virtual tutoring (Graesser et al. 2005). Semantic similarity can be measured at different levels, ranging from words and phrases, to paragraphs and documents. In this paper, we focus on quantifying semantic similarity at sentence level, i.e. compute the semantic similarity between two given sentences. The final outcome of the evaluation process is the ability to detect some relations of semantic similarity between the two input texts, i.e. paraphrase or entailment.

To measure the semantic similarity between short texts, we designed an approach based, in part, on the principle of compositionality according to which the meaning of a text can be determined by the meaning of its constituents and the rules used to combine them. The constituents of a text are represented by its lexical tokens (i.e. words, numbers, or punctuation marks), while their interactions are governed by the syntactic structure of the sentence. In the approach presented in this paper, we only consider how the individual word meanings compose the overall meaning of an entire sentence in a simple additive manner thus ignoring the syntactic, semantic, and pragmatics rules that might combine the individual word meanings in more complex ways. This decision had been made for the following reasons. First, an approach focusing on word compositionality is complex enough as it involves making a set of choices, e.g. using or not using weighting when summing up word-level similarities, whose outcome may lead to a more or less competitive solution. A significant amount of work is necessary to better understand the space of methods spanned by those choices and how they impact the overall performance of the basic approach on solving the task at hand, e.g. identifying whether two texts are in a paraphrase relation or not. The work presented here is a step towards a better understanding of the optimal set of choices for the basic additive approach.

Second, we wanted to compare several variants of the basic approach with other variants, proposed by others, of the basic additive approach. We are only aware of the variant

presented by Corley and Mihalcea (2005). They used a similar methodology of computing the similarity of short texts based on word-to-word similarity metrics. Our work differs from their work in that we provide a more systematic analysis of additive approaches that rely on word-to-word similarity metrics. Furthermore, due to our systematic exploration of the additive approaches, our most successful variant of the basic approach is significantly better. This best approach differs from Corley and Mihalcea's approach in several aspects. We compute the semantic similarity between two short texts based on the exclusive pairing of words. Furthermore, we rely on different preprocessing and normalization techniques. Similar to Corley and Mihalcea, we evaluate our methods on the same datasets, the Microsoft Research Paraphrase Corpus (MSRP, (Dolan, Quirk, and Brockett 2004)), and the Recognizing Textual Entailment corpus (RTE, (Dagan, Glickman, and Magnini 2005)), which are representative for the two tasks of paraphrase identification and textual entailment recognition, respectively. This allows us for a direct and fair comparison of our work and Corley and Mihalcea's work. To avoid any doubts with respect to the differences between our approach and theirs, e.g. resulting from using different part of speech taggers, we implemented ourselves the Corley and Mihalcea's approach. We are able to provide a very detailed and fair comparison.

We are aware of the limitations of the compositionality approach to capture the meaning of texts. However, given its computational appealing the interesting research questions that we and others, e.g Corley and Mihalcea, have tried to answer is the extent to which it could be used in conjunction with word-to-word similarity measures to solve text-to-text similarity tasks such as paraphrase identification.

We follow this introductory part by describing the word-to-word similarity metrics that we used and then present in detail our methodology and experimental results. We end the paper with conclusions and discussion.

Word-to-Word Similarity

Two major classes of word-based similarity metrics have emerged during the last decade or so. A first class of word-to-word similarity metrics includes knowledge-based metrics. In this class of metrics, the semantic similarity between words is assessed based on dictionaries or thesauri. One very popular dictionary is the WordNet lexical database (Miller 1995). In WordNet, English words are grouped into synonym sets, called synsets, which define one meaning or concept. Synsets are linked to each other via lexico-semantic relations such as hypernymy (equivalent to an IS-A relation in Artificial Intelligence). Several metrics have been proposed to make use of the structure of the WordNet database (Pedersen, Patwardhan, and Michelizzi 2004). A major advantage of using knowledge-based metrics is that they can be very reliable as they are based on expert human judgments. The disadvantage is that they are limited with respect to the class of words which can be compared. Most of these metrics can only be computed between content words (nouns, verbs, adjectives, and adverbs). Furthermore, they can only compute similarity between contents words of the same type, e.g. only between nouns, or certain types, e.g. can only compute similarity between nouns and between verbs but not between adjectives and adverbs. A few metrics are able to compute similarity across content word categories, e.g. between adjectives and adverbs (i.e. HSO or LESK), but because they proved to be very slow in our implementation, we had to ignore them for the time being. Another issue of using WordNet-based metrics is the need to specify the meaning of words. Finding the corect meaning of words in text is still a difficult task in natural language processing, also known as the problem of word sense disambiguation (WSD). In this work we chose to avoid this WSD problem by choosing those meanings that maximize the similarity between two input words, e.g. we picked the first sense in WordNet for a given word. We experimented with several of the WordNet-based metrics which can compute similarity between nouns and verbs only (JCN (Jiang and Conrath 1997), LCH (Leacock and Chodorow 1998), LIN (Lin 1998), RESNIK (Resnik 1995), and WUP (Wu and Palmer 1994)). We made sure all these metrics are properly normalized by their maximum possible values. It should be noted that Corley and Mihalcea also used these metrics.

The second class of word-to-word similarity metrics, which is widely used in current research, includes metrics that rely on vectorial representations of word meanings. In such representations, the word meanings are represented as vectors into a high dimensional space, where each dimension is believed to be representative of an abstract/latent semantic concept. Computing the similarity between two words is equivalent to computing the cosine, i.e. normalized dot product, between the corresponding vectors. The challenge with such vectorial representations is the derivation of the semantic space, i.e. the vector representations. Latent Semantic Analysis (LSA; (Landauer et al. 2007)) is one very popular and mathematically intuitive technique to derive the semantic space based on analyzing word-to-word co-occurrence in large collections of texts. The advantage of the vector-based metrics is that a similarity measure can be computed between virtually any two words that are being found in the analyzed texts. In our work, we experimented with an LSA space computed from the TASA corpus (compiled by Touchstone Applied Science Associates), a balanced collection of representative texts from various genres (science, language arts, health, economics, social studies, business, and others). Using LSA we were able to compute similarity measures for adjectives and adverbs too, not only for nouns and verbs, as in the WordNet-based metrics.

Our Approach: Greedy Pairing through Word-to-Word Similarity

As already mentioned, our basic approach to the task of assessing similarity between texts is to extend the word-to-word similarity metrics based on the compositionality principle. To that end, the similarity between two texts is composed in a simple additive manner from the individual similarities between pairs of words. The approach can be summarized in three major steps:

1. First, construct a set (S) of exclusive pairs of similar words between the two input texts. By exclusive we mean that a

particular word can be part of only one pair.

2. Use these pairs of words to compute an overall similarity score (*Sim*) between the texts, through a weighted sum.
3. Normalize the computed sum with a weighted length of the original texts.

It should be noted that the semantic similarity of two texts can be computed in a unidirectional or bidirectional manner, depending on the task at hand. For entailment recognition, we need to compute a unidirectional similarity score from the entailing hypothesis (*H*) to the entailed text (*T*). In case of paraphrase identification, we need to compute a bidirectional similarity score by combining the unidirectional scores from one text (*A*) to the other (*B*) and vice versa. To compute the unidirectional similarity score from a text A to a text B ($Sim(A \rightarrow B)$), we employ a greedy strategy to find the closest matches of words in A, to the words in B. In addition, we exclude those words which have a similarity value lower than a predefined threshold[1].

We exemplify this basic idea on the positive example of paraphrase given at the beginning of this paper. We find that six words in text A have identical correspondents in text B (i.e. *the*, *is*, *the*, *second*, *third*, *trimester*). In addition, two more pairs can be greedily formed, based on the maximum similarity score between their words (i.e. *procedure* with *technique*, and *performed* with *used*), while other words will not be paired, since they miss appropriate corespondents in the other text (i.e. *generally* and *pregnancy*). Notice that the words are not uniquely excluded. For instance, the determiner *the* appears twice in both sentences and therefore will be paired twice. If there is more than one similar match to a word, then only the first, most similar occurence is selected for pairing.

This form of pairing is different form Corley and Mihalcea's work, where words are uniquely selected for pairing. Another difference is that all types of words are included in the matching process in our method. Previous studies only looked at content words and numbers. We explored both these options and found that in the paraphrase case, looking at all words (content plus function words; functions words are words such as prepositions and conjunctions which have more of a grammatical function) will give better results, while for entailment the variant suggested by Corley and Mihalcea, i.e. using only content words, seems to work better.

Another important aspect of our approach concerns the matching of words. Given two words, we have a first choice (Choice A) to compare them lexically by looking at their original, inflected form, or by looking at their base, or lemmatized, form (e.g., the plural word *children* would be mapped to its base form *child*). If these forms are identical then the words are considered a match. In case there is no match, we can compare them through a word-to-word semantic similarity metric. In this latter case, there is a second choice (Choice B) to be made:

- compare only words that have the same part of speech (i.e. match *child* with *boy* since they have the same part-of-speech (NN - common noun) but do not match *children* with *boy* as they have different parts of speech, NNS, for plural noun, and NN, respectively), or
- compare words if they belong to the same broad category, i.e. match all nouns with nouns regardless of their fine differences (e.g., in this case *children* and *boy* would be matched as they are both nouns)

Note that these two choices (A and B) are closely related, since the inflections of words are usually encoded in their part-of-speech. Interestingly, we found that using these options will present different results depending on the dataset that we use. For paraphrasing, it is more effective to compare words based on their base forms, and semantically by their part-of-speech, while for entailment, it is best when comparing them using their original, inflected forms, and semantically, based on their broad category.

The next step in computing the overall similarity score is to combine the word-level semantic similarity of each pair into an overall similarity score at text level. The overall score is then normalized based on the input texts. When computing the overlap score, one might weight the importance of each word. The idea of weighting is to allow for certain words, e.g. very rare words in a short sentence may greatly impact its meaning, to have a larger impact on the overall similarity score. Word specificity has been widely used as a measure of word importance. Usually, the inverse document frequency (idf) of a word, which measures how rare a word is in a large collection of documents, is used as a measure of specificity. The fewer documents the word occurs in the higher the specificity. A related issue is computing the weighting of a pair of words when each word has a different idf. We explored several solutions: considering the idf of the word in the first sentence, computing an average between the two idfs, or taking the maximum between the two. We found that taking the maximum idf works best for both entailment and paraphrase tasks. We therefore employ the following formulas to compute the weighted, Sim_W, or non-weighted, Sim, overlap similarity scores from text A to text B, where w_a is a word in A, w_b is a word in B, $p(w_a, w_b)$ is a pair in the set S of similarly matched words, and $WordSim$ is the computed word-to-word similarity:

$$Sim(A \rightarrow B) = \sum_{p(w_a,w_b) \in S(A \rightarrow B)} WordSim(w_a, w_b) \quad (1)$$

$$Sim_W(A \rightarrow B) = \sum_{p(w_a,w_b) \in S(A \rightarrow B)} WordSim(w_a, w_b) * \\ * Max(idf(w_a), idf(w_b)) \quad (2)$$

Finally, we normalize the overall similarity score on the weighted length of the input texts. This is computed by summing up the idf weights of all the lexical tokens that were considered in the first step of our process (when the set of paired words was constructed). If weighting is not used, the length of the texts is used for normalization,

[1]Corley and Mihalcea also suggested the use of a similarity threshold but they did not apply it in their experiments

where length is computed by counting all tokens previously included in the matching process. Since there are two lengths, one from each of the two input texts (i.e. $norm_A$, $norm_B$), there are several ways to compute the normalization. For the unidirectional case (i.e. entailment, $EntSim$), we normalize by the weighted length of the entailing hypothesis, (i.e. $WeightedNorm_H$). In the bidirectional case (i.e. paraphrasing, $ParaSim$), we normalize by the average of the two texts ($\frac{norm_A+norm_B}{2}$) or the maximum ($Max(norm_A, norm_B)$). In this case, we found that normalizing by the maximum length gives best results.

Our final formulas to compute the semantic similarity on the tasks of paraphrasing and entailment are shown below:

$$ParaSim(A, B) = \frac{Sim(A \rightarrow B) + Sim(B \rightarrow A)}{2 * Max(norm_A, norm_B)} \quad (3)$$

$$EntSim(T, H) = \frac{Sim_W(H \rightarrow T)}{WeightedNorm_H} \quad (4)$$

Experiments and Results

In this section, we present performance results of our approach to computing the semantic similarity between two texts. The results were obtained by applying the approach to two semantic similarity tasks: paraphrase identification and entailment recognition. In each of these tasks, the best variants of the basic similarity approach were found based on exploratory experiments using the evaluation datasets: MSRP for paraphrase identification and RTE for entailment recognition. The training portion of these datasets were used to find and train the best combination of parameters and then performance results were reported on the test portion of the datasets.

For preprocessing, we use the Stanford NLP Core library to tokenize, lemmatize and extract the part-of-speech tags. For idf weighting, we use an idf index which we previously computed from the English Wikipedia.

We report our results in comparison with the previous work of Corley and Mihalcea (2005). We report their results in two ways: the original results reported by Corley and Mihalcea (2005) and also the results we obtained using our own implementation of their approach. In our experiments, we focus only on a single word-to-word metric at a time. Corley and Mihalcea report their best results when all measures are combined into an averaged value. We did not find an improvement of the combined approach when using our own implementation. In addition, our focus was on individual word metrics in order to determine which one is better for which similarity task. Similarly to Corley an Mihalcea we compare the performance of each word-to-word metric in terms of accuracy (percentage of correct predictions), precision (percentages of correct predictions out of all positive predictions by the system), recall (percentage of positive instance correctly predicted by the system) and F-measure (the harmonic mean of precision and recall). In a first step of our experiments, we compare 5 word-to-word metrics based on WordNet knowledge (JCN, LCH, WUP, RESNIK, LIN), one LSA-based metric, and one baseline method where we do not use any semantic similarity metric (i.e. the system will pair only those words that have identical lexical forms). Then we take the best metric found and compare it with other variants of our method and check for significant improvement in performance (i.e. using idf weighting vs. not, using word-to-word semantic similarity versus no word-to-word semantic similarity).

For the word-to-word similarity metrics, which are normalized, we found that using a minimum word-to-word threshold of 0.5 (words need to be at least that similar to accept them as a pair) usually gives best results. Therefore for the simplicity of our experiments we decided to keep this threshold fixed.

After computing the similarity score, we use a simple version of the perceptron algorithm to find the optimum similarity threshold, which gives maximum classification accuracy on the training data, in order to separate positive instances (i.e. instances in the dataset with true paraphrase or entailment relations) from the negative instances. Once this threshold is found, the system will classify a new instance as negative, if its score is below the threshold, or positive, if it is equal or above the threshold.

Paraphrase Identification

For the task of paraphrase identification we used the MSRP Corpus (Dolan, Quirk, and Brockett 2004) which has a total of 4076 training instances and 1725 testing instances. A simple baseline for this corpus is the majority baseline, where all instances are classified as positive. The baseline gives an accuracy and precision of 66.5% and perfect recall. In Table 1, we show results for the six word-to-word similarity metrics, when using our implementation of Corley and Mihalcea's algorithm versus their own reported results versus our proposed method. We also compare these results with one other approach when using a baseline lexical word-to-word measure (for the baseline, the similarity between two input words is 1 if their corresponding lemmas are lexically identical and 0, otherwise). Because in some cases it is not clear whether we have a significant improvement in performance, we ran paired t-tests between our method and our implementation of Corley and Mihalcea's method and also between methods using word-to-word semantic similarity metrics and the baseline method.

In addition, we investigated the role of idf weighting. For that, we selected the top three best performing WordNet measures (JCN, LCH and LIN) and compare results of the approach with and without idf (Table 3). As before, we checked for significance in these results using paired t-tests.

We next summarize our findings based on Tables 1 and 3:

- All our methods except LSA have a significant improvement in performance compared to the baseline, i.e. when not using any word-to-word similarity metric.
- All Corley and Mihalcea's metrics except LCH, LIN and WUP have a significant improvement in performance compared to the baseline, i.e. when not using any word-to-word similarity metric.
- All our methods present a significant improvement in performance when compared to Corley and Mihalcea.

Table 1: Results on the Paraphrase Identification Task (MSRP).

W2W Metric	Corley&Mihalcea - reported				Corley&Mihalcea - our version				Our method			
	Acc	Prec	Rec	F-Measure	Acc	Prec	Rec	F-Measure	Acc	Prec	Rec	F-Measure
JCN	.699	.707	.935	0.805	.714	.723	.924	.811	.747	.755	.918	.828
LCH	.699	.708	.931	0.804	.711	.732	.890	.804	**.757**	.783	.879	.828
LIN	.702	.706	.947	0.809	.714	.719	.934	.813	.746	.787	.847	.816
RES	.692	.705	.921	0.799	.712	.737	.881	.803	.742	.783	.847	.814
WUP	.699	.705	.941	0.806	.703	.706	.950	.810	.746	.772	.877	.821
LSA	NA				.706	.717	.921	.806	.730	.773	.840	.805
Lexical	NA				.696	.738	.843	.787	.731	.776	.838	.806

Table 2: Results on the Entailment Recognition Task (RTE).

W2W Metric	Corley&Mihalcea - reported				Corley&Mihalcea - our version				Our method			
	Acc	Prec	Rec	F-Measure	Acc	Prec	Rec	F-Measure	Acc	Prec	Rec	F-Measure
JCN	.575	.566	.643	.602	.567	.554	.692	.616	.589	.554	.910	.689
LCH	.583	.573	.650	.609	.559	.545	.715	.618	**.598**	.565	.848	.678
LIN	.574	.568	.620	.593	.575	.561	.685	.617	.582	.562	.752	.643
RES	.579	.572	.628	.598	.569	.554	.705	.620	.577	.552	.825	.661
WUP	.580	.570	.648	.607	.560	.540	.815	.649	.591	.562	.830	.670
LSA	NA				**.584**	.559	.793	.656	.579	.556	.780	.649
Lexical	NA				.553	.542	.683	.604	.554	.542	.695	.609

Table 3: The importance of idf to Paraphrase Identification.

W2W Metric	With idf			No idf		
	Acc	Prec	Rec	Acc	Prec	Rec
Corley and Mihalcea - our version						
JCN	.714	.723	.924	.725	.767	.841
LCH	.711	.732	.890	.703	.765	.797
LIN	.714	.719	.934	.723	.752	.873
Our Method						
JCN	.701	.728	.880	.747	.755	.918
LCH	.710	.722	.915	.757	.783	.879
LIN	.704	.724	.897	.746	.787	.847

Table 4: The importance of idf to Entailment Recognition.

W2W Metric	With idf			No idf		
	Acc	Prec	Rec	Acc	Prec	Rec
Corley and Mihalcea - our version						
JCN	.567	.554	.692	.561	.544	.752
LCH	.559	.545	.715	.553	.537	.755
LIN	.575	.561	.685	.558	.544	.707
Our Method						
JCN	.589	.554	.910	.561	.540	.817
LCH	.598	.565	.848	.571	.551	.775
LIN	.582	.562	.752	.569	.542	.880

Entailment Recognition

For the task of entailment recognition we use the Recognizing Textual Entailment Corpus (RTE, (Dagan, Glickman, and Magnini 2005)) proposed by the PASCAL European research group, which consists of 567 training pairs of text-hypothesis pairs, and 800 testing pairs. Both the training and the testing are equally distributed, so the simple random-guessing baseline provides a 50% chance of success on detecting whether a given instance is positive (meaning there is an entailment relation between the text and hypothesis) or negative. Similarly to the paraphrase task, we show in Table 2 results on the six word-to-word metrics and a baseline lexical metric. Then, just as before, in Table 4 we show results obtained with the top performing three WordNet metrics and compare the role of idf weighting. Again, we used paired t-tests to check for significance of the results. We found that:

- All our methods except RES have a significant improvement in performance than when using just the lexical baseline metric.
- Only Corley and Mihalcea's method with LSA has a significant improvement in performance than when not using any word-to-word similarity metric.
- Although all our methods, except LSA, show improvement in performance over Corley and Mihalcea's method, only the ones using LCH and WUP show a significant improvement. Regarding the LSA, Corley and Mihalcea is better than our method, but not significantly better.

Discussions and Conclusions

Based on the results presented in Tables 1 and 2 we conclude that using different preprocessing and idf weighting schemes results in different performance. We see that for paraphrasing, our implementation of Corley's and Mihalcea shows an improvement in accuracy and precision, while for entailment it shows a relative decrease in accuracy and precision. When compared with our methods, we managed to get a significant improvement in performance on the paraphrasing task (highest accuracy of .757 when using the LCH measure), and a modest improvement in accuracy on the entailment (maximum accuracy of .598 when also using LCH). Since most of the methods using word-to-word semantic metrics show a significant improvement on both tasks over a baseline lexical metric, we conclude that these metrics are a good choice for measuring the semantic similarity between short texts. LCH seems to be the best metric as we obtained best accuracy and significant improvement over the Corley and Mihalcea's method using it. Interestingly, for the LSA metric, the performance was rather weak on the paraphrasing task, but it was slightly better, in terms of accuracy (.584), with the Corley and Mihalcea's entailment method.

When looking at the importance of using idf weighting for our two tasks we see that, for paraphrasing, using idf is significantly detrimental for our method, but has no significant effect on Corley and Mihalcea's method. This may be due to the fact that our method is also making use of function words to compute the similarity. These function words are very common words, and therefore, by having very low idf values, their importance in the final score is decreased significantly. A broader conclusion is that, while using function words is important for the task of paraphrase assessment, weighting them is not. In the entailment case, we found that including function words in our approach is not helpful and thus excluded them. Using idf only on content words is beneficial to the entailment task but presents only a modest improvement on our and Corley and Mihalcea's methods. Overall, we conclude that using idf weighting, although a very useful technique when analyzing larger documents in Information Retrieval tasks, might not be such a good idea when comparing texts of rather short length. Also, we found that the way the computed overlap score is normalized has a significant effect on the performance scores.

For future work, it would be interesting to know how are these methods performing on texts of larger length, either the size of a paragraph or larger documents.

As a final remark, when we compare our results to any other previous work not only word-to-word similarity-based, we claim that for the paraphrasing task we achieved state of the art results (see (Androutsopoulos and Malakasiotis 2010) for a comparative review of these methods).

References

Androutsopoulos, I., and Malakasiotis, P. 2010. A survey of paraphrasing and textual entailment methods. *Journal of Artificial Intelligence Research* 38:135–187.

Corley, C., and Mihalcea, R. 2005. Measuring the semantic similarity of texts. In *Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*. Ann Arbor, MI.

Dagan, I.; Glickman, O.; and Magnini, B. 2005. The pascal recognising textual entailment challenge. In *Proceedings of the PASCAL Challenge Workshop on Recognizing Textual Entailment*.

Dolan, B.; Quirk, C.; and Brockett, C. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of 20th International Conference on Computational Linguistics (COLING)*.

Graesser, A. C.; Olney, A.; Haynes, B. C.; and Chipman, P. 2005. Autotutor: A cognitive system that simulates a tutor that facilitates learning through mixed-initiative dialogue. In *Cognitive Systems: Human Cognitive Models in Systems Design.* Mahwah: Erlbaum.

Ibrahim, A.; Katz, B.; and Lin, J. 2003. Extracting structural paraphrases from aligned monolingual corpora. In *Proceedings of the ACL Workshop on Paraphrasing*, 57–64.

Iordanskaja, L.; Kittredge, R.; and Polguere, A. 1991. *Natural Language Generation in Artificial Intelligence and Computational Linguistics*. Norwell, MA, USA: Kluwer Academic Publishers. chapter Lexical selection and paraphrase in a meaning-text generation model, 293–312.

Jiang, J. J., and Conrath, D. W. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proc. of the Int'l. Conf. on Research in Computational Linguistics*.

Landauer, T. K.; McNamara, D. S.; Dennis, S.; and Kintsch, W. 2007. *Handbook of Latent Semantic Analysis*. Mahwah, NJ: Erlbaum.

Leacock, C., and Chodorow, M. 1998. *WordNet, An Electronic Lexical Database*. The MIT Press. chapter Combining local context and WordNet sense similarity for word sense identification.

Lin, D. 1998. An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning*.

Lodhi, H.; Saunders, C.; Shawe-Taylor, J.; Cristianini, N.; and Watkins, C. 2002. Text classification using string kernels. *Journal of Machine Learning Research* 2:419–444.

Miller, G. A. 1995. Wordnet: A lexical database for english. *Communications of the ACM* 38(11):39–41.

Park, E.-K.; Ra, D.-Y.; and Jang, M.-G. 2005. Techniques for improving web retrieval effectiveness. *Information Processing and Management* 41(5):1207–1223.

Pedersen, T.; Patwardhan, S.; and Michelizzi, J. 2004. Wordnet::similarity - measuring the relatedness of concepts. In *Proceedings of Fifth Annual Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-2004)*, 38–41.

Resnik, P. 1995. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, 448–453.

Wu, Z., and Palmer, M. S. 1994. Verb semantics and lexical selection. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 133–138.