Similarity Measures Based on Latent Dirichlet Allocation

Vasile Rus, Nobal Niraula, and Rajendra Banjade

Department of Computer Science, The University of Memphis
Memphis, TN, 38152, USA
{vrus,nobal}@memphis.edu

Abstract. We present in this paper the results of our investigation on semantic similarity measures at word- and sentence-level based on two fully-automated approaches to deriving meaning from large corpora: Latent Dirichlet Allocation, a probabilistic approach, and Latent Semantic Analysis, an algebraic approach. The focus is on similarity measures based on Latent Dirichlet Allocation, due to its novelty aspects, while the Latent Semantic Analysis measures are used for comparison purposes. We explore two types of measures based on Latent Dirichlet Allocation: measures based on distances between probability distribution that can be applied directly to larger texts such as sentences and a word-to-word similarity measure that is then expanded to work at sentence-level. We present results using paraphrase identification data in the Microsoft Research Paraphrase corpus.

Keywords: semantic similarity, paraphrase identification, unsupervised meaning derivation.

1 Introduction

We address in this paper the problem of semantic similarity between two texts. The task of semantic similarity is about making a judgment with respect to how semantically similar two texts are. The judgment can be quantitative, e.g. a normalized score, or qualitative, e.g. one text is (or is not) a paraphrase of the other.

The problem of semantic similarity is a central topic in Natural Language Processing as it is important to a number of applications such as providing evidence for the correctness of answers in Question Answering [1], increase diversity of generated text in Natural Language Generation [2], assessing the correctness of student responses in Intelligent Tutoring Systems [3, 4], or identifying duplicate bug reports in Software Testing [5].

We focus in this paper on the problem of semantic similarity at word- and sentence- level. At word-level, the task is about judging how similar two words are, e.g. *procedure* and *technique*. As an example of sentence-to-sentence similarity [6, 7], we show below a pair of sentences from the Microsoft Research Paraphrase (MSRP; [8]). The shown example constitutes a positive instance of a paraphrase in MSRP (to be precise, it is instance #51 in MSRP test data).

A. Gelbukh (Ed.): CICLing 2013, Part I, LNCS 7816, pp. 459–470, 2013.

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.*

We propose novel solutions to the task of semantic similarity both at word and sentence level. We rely on probabilistic and algebraic methods that can automatically derive word and sentence meaning representations from large collection of texts in the form of latent topics or concepts. The probabilistic method we use is Latent Dirichlet Allocation (LDA, [9]). LDA models documents as topic distributions and topics as distributions over words in the vocabulary. Each word has a certain contribution to a topic. Based on these distributions and contributions we define both word-to-word semantic similarity measures and text-to-text semantic similarity measures. The LDA-based word-to-word semantic similarity measure is further used in conjunction with a greedy and optimal matching method to measure similarity between larger texts such as sentences. The text-to-text measures are directly used to compute the similarity of texts such as between two sentences, the focus of our work.

The proposed semantic similarity solutions based on LDA are compared with solutions based on Latent Semantic Analysis [10]. Like LDA, LSA is fully automated. LSA starts with a term-document matrix that represents the distribution of words in documents and the distribution of documents over the words. The term vectors (as well as the document vectors) in the original term-document matrix are mapped using the mathematical procedure of Singular Value Decomposition (SVD) into a reduced dimensionality space (300-500 dimensions). Words are represented as vectors in this LSA semantic space whose dimensions form latent semantic concepts. Documents are also represented as vectors in the reduced space. Similarity of individual words and texts are computed based on vector algebra. For instance, the similarity between two words is computed as the cosine (normalized dot-product) between the corresponding word vectors.

Given that both LDA and LSA require the specification of a desired number of latent topics or concepts a priori, an interesting question relates to which of these methods best capture the semantics of words and texts for the same number of topics or concepts. The broader question would be which of these two methods can best capture the meaning of words and texts and in what conditions. This paper is one step in that direction of elucidating the strengths and weaknesses of these methodologies for meaning inference in the context of the paraphrase identification.

We have experimented with the above methods using the Microsoft Research Paraphrase corpus [8]. We provide experimental results on this data set using both a greedy method and an optimal matching method based on the job assignment problem, a famous combinatorial optimization problem.

The rest of the paper is organized as in the followings. The next section provides an overview of related work. Then, we describe the semantic similarity measures based on LDA. The Experiments and Results section describes our experimental setup and the results obtained. We conclude the paper with Discussion and Conclusions.

2 Related Work

The task of semantic similarity can be formulated at different levels of granularity ranging from word-to-word similarity to sentence-to-sentence similarity to document-to-document similarity or a combination of these such as word-to-sentence or sentence-to-document similarity. We will first review word-level semantic similarity measures followed by sentence-level measures. It should be noted that some approaches, such as LSA, are directly applicable at both at word- and sentence-level. Standard word-to-word similarity can be expanded to larger texts through some additional mechanism [7, 8]. It is important to add that the review of related work that follows is by no means an exhaustive one.

Research on semantic similarity of texts focused initially on measuring similarity between individual words. A group of word-to-word similarity measures were defined that use lexico-semantic information in WordNet [11]. WordNet is a lexical database of English that groups together words that have the same meaning, i.e. synonyms, into synsets (synonymous sets). Synsets are also referred to as concepts. For instance, the synset of {affectionate, fond, lovesome, tender, warm} corresponds to the concept of (having or displaying warmth or affection), which is the definition of the concept in WordNet.

There are nearly a dozen WordNet-based similarity measures available [12]. These measures are usually divided into two groups: similarity measures and relatedness measures. The similarity measures are limited to within-category concepts and usually they work only for the nouns and verbs categories. The text relatedness measures on the other hand can be used to compute similarity among words belonging to different categories, e.g. between a noun and an adjective.

Examples of word relatedness measures (implemented in the WordNet::Similarity package [12]) are: HSO [13], LESK [14], and VECTOR [15]. Given two WordNet nodes, i.e. concepts, these measures provide a real value indicating how semantically related the two concepts are. The HSO measure is path based, i.e. uses the relations between concepts, and assigns direction to relations in WordNet. For example, is-a relation is upwards, while has-part relation is horizontal. The LESK and VECTOR measures are gloss-based. That is, they use the text of the gloss as the source of meaning for the underlying concept.

One challenge with the above word-to-word relatedness measures is that they cannot be directly applied to compute similarity of larger texts such as sentences. Researchers have proposed methods to extend the word-to-word (W2W) relatedness measures to text-to-text (T2T) relatedness measures [7, 8]. Another challenge with the WordNet similarity measures is the fact that texts express meaning using words and not concepts. To be able to use the WordNet-based word-to-word similarity measures, the words must be mapped to concepts in WordNet, i.e. the word sense disambiguation (WSD) problem must be solved. Other solutions can be adopted such as selecting the most frequent sense for each word or even trying all possible senses [16]. Our proposed word-to-word similarity measure based on Latent Dirichlet Allocation or LSA-based word-level measures do not have this latter challenge.

Given its importance to many semantic tasks such as paraphrase identification [8] or textual entailment [17], the semantic similarity problem at sentence level has been addressed using various solutions that range from simple word overlap to greedy

methods that rely on word-to-word similarity measures [7] to algebraic methods [18] to machine learning based solutions [19].

The most relevant work to ours, [18], investigated the role of Latent Semantic Analysis [10] in solving the paraphrase identification task. LSA is a vectorial representation in which a word is represented as a vector in a reduced dimensionality 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 word vectors. The challenge with such vectorial representations is the derivation of the semantic space, i.e. discovering the latent dimensions or concepts of the LSA space. 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).

Two different ways to compute semantic similarity between two texts based on LSA were proposed in [18]. First, they used LSA to compute a word-to-word similarity measure which then they combined with a greedy-matching method at sentence level. For instance, each word in one sentence was greedily paired with a word in the other sentence. An average of these maximum word-to-word similarities was then assigned as the semantic similarity score of the two sentences. Second, LSA was used to directly compute the similarity of two sentences by applying the cosine (normalized dot product) of the LSA vectors of the sentences. The LSA vector of a sentence was computed using vector algebra, i.e. by adding up the vectors of the individual words in a sentence. We present later results with these methods as well as with a method based on optimal matching.

LDA was rarely used for semantic similarity. To the best of our knowledge LDA has not been used so far for addressing the task of paraphrase identification, which we address here. The closest use of LDA for a semantic task was for ranking answers to a question in Question Answering (QA; [20]). Given a question, they ranked candidate answers based on how similar these answers were to the target question. That is, for each question-answer pair they generated an LDA model which they then used to compute a degree of similarity (DES) that consists of the product of two measures: sim1 and sim2. Sim1 captures the word-level similarities of the topics present in an answer and the question. Sim2 measures the similarities between the topic distributions in an answer and the question. The LDA model was generated based solely on each question and candidate answers. As opposed to our task in which we compute the similarity between sentences, the answers to questions in [20] are longer, consisting of more than one sentence. For LDA, this particular difference is important when it comes to semantic similarity as the shorter the texts the sparser the distributions, e.g. the distribution over topics in the text, based on which the similarity is computed.

Another use of LDA for computing similarity between blogs relied on a very simple measure of computing the dot product of topic vectors as opposed to a similarity of distributions [21].

Similar to [20], we define several semantic similarity measures based on various distributions used in the LDA model. We do use Information Radius as [20] and, in addition, propose similarity measures based on Hellinger and Manhattan distances.

Furthermore, we use LDA for measuring word-to-word similarities and use these values in a greedy and optimal matching method at sentence-level. Finally, we compare the results with LSA-based results. The comparison is so designed to be as informative as possible, e.g. the number of topics in LDA matches the number of latent concepts/dimensions in LSA.

It should be noted that LDA has a conceptual advantage over LSA. LDA models explicitly different meaning of words, i.e. it handles polysemy. In LDA, each topic is a set of words that together define a meaning or concept, i.e. a word sense. A word belongs to different topics, which can be regarded as its various senses. On the other hand, LSA has a unique representation for a word. That is, all meanings of a word are represented by the same LSA vector making difficult to infer which meaning is being represented. Some argue that the LSA vector represents that dominant meaning of a word while others believe the LSA vector represents an average meaning of all meanings of the word.

3 Similarity Measures Based on Latent Dirichlet Allocation

As we already mentioned, LDA is a probabilistic generative model in which documents are viewed as distributions over a set of topics and each word in a document is generated based on a distribution over words that is specific to each topic. We denote by θ the distributions over topics and by φ distributions over words. Full details about the LDA model can be found in [9].

A first semantic similarity measure among words would then be defined as a dot-product between the corresponding vectors representing the contributions of each word to a topic ($\varphi_t(w)$ – represents the contribution of word w to topic t). It should be noted that the contributions of each word to the topics does not constitute a distribution, i.e. the sum of contributions does not add up to 1. Assuming the number of topics T, a word-to-word measure is defined by the formula below.

$$LDA - w2w(w,v) = \sum_{t=1}^{T} \varphi_t\,(w)\varphi_t\,(v)$$

More global text-to-text similarity measures could be defined based on the distributions over topics (θ) and distributions over words (φ) defined by LDA. Because a document is a distribution over topics, the similarity of two texts needs to be computed in terms of similarity of distributions. The Kullback-Leibler (KL) divergence defines a distance, or how dissimilar, two distributions p and q are as in the formula below.

$$KL(p,q) = \sum_{i=1}^{T} p_i \log \frac{p_i}{q_i}$$

If we replace p with θ_d (text/document d's distribution over topics) and q with θ_c (text/document c's distribution over topics) we obtain the KL distance between two documents (documents d and c in our example). Furthermore, KL can be used to compute the distance between two topics using their distributions over words (φ_{t1} and

φ_{t2}). The KL distance has two major problems. In case q_i is zero KL is not defined. Furthermore, KL is not symmetric which does not fit well with semantic similarity measures which in general are symmetric. That is, if text A is a paraphrase of text B that text B is a paraphrase of text A. The Information Radius measure solves these problems by considering the average of p_i and q_i as below.

$$IR(p,q) = \sum_{i=1}^{T} p_i \log \frac{2 \times p_i}{p_i + q_i} + \sum_{i=1}^{T} q_i \log \frac{2 \times q_i}{p_i + q_i}$$

The IR can be transformed into a similarity measure as in the following (Dagan, Lee, & Pereira, 1997):

$$SIM(p,q) = 10^{-\delta IR(c,d)}$$

All our results reported in this paper for LDA similarity measures between two documents c and d are computed by multiplying the similarities between the distribution over topics (θ_d and θ_c) and distribution over words (φ_{t1} and φ_{t2}). That is, two texts would be similar if the texts have similar topic distributions and the topics are similar themselves, i.e. have similar word distributions.

The Hellinger distance between two distributions is another option that allows avoiding the shortcomings of the KL distance.

$$HD(p,q) = \frac{1}{\sqrt{2}} \sqrt{\sum_{1}^{T} (\sqrt{p_i} - \sqrt{q_i})^2}$$

The Hellinger distance varies from 0 to 1 and is defined for all values of p_i and q_i. A value of 1 means the distance is maximum and thus the distributions are very different. A value of 0 means the distributions are very similar. We can transform the Hellinger distance into a similarity measure by subtracting it from 1 such that a zero distance means a large similarity score and vice versa.

Lastly, we used the Manhattan distance between distributions p and q as defined below.

$$MD(p,q) = 2 \times (1 - \sum_{1}^{T} \min(p_i, q_i))$$

MD is symmetric, defined for any values of p and q, and ranges between 0 and 2. We can divide MD by 2 and subtract from 1 to transform it into a similarity measure.

4 Experimental Setup and Results

We present in this section results with the previously described methods on the Microsoft Research Paraphrase corpus (MSRP; [8]). The MSRP corpus is the largest publicly available annotated paraphrase corpus and has been used in most of the

recent studies that addressed the problem of paraphrase identification. The corpus consists of 5,801 sentence pairs collected from newswire articles, 3,900 of which were labeled as paraphrases by human annotators. The whole set is divided into a training subset (4,076 sentences of which 2,753, or 67.5%, are true paraphrases), and a test subset (1,725 pairs of which 1,147, or 66.5%, are true paraphrases). The average number of words per sentence is 17.

A simple baseline for this corpus is the majority baseline, where all instances are labeled with the majority class label in the training corpus, which in MSRP is the positive class label. The baseline gives an accuracy and precision of 66.5% and perfect recall.

For the proposed methods, we first present results obtained using 300 dimensions for the LSA space, a standard value, and a similar number of topics for LDA. This number of dimensions has been empirically established by LSA researchers. Then, we vary the number of topics for the LDA model and observe changes in performance. Fewer topics usually means semantically less coherent topics as many words with different senses will be grouped under the same topic. More topics on the other hand would lead to somehow more coherent topics (this is an open research question, actually) but also sparser topic distributions for short texts as exemplified later.

We follow a training-testing methodology according to which we first train the proposed methods on a set of training data after which we use the learned models on testing data. In our case, we learn a threshold for the text-to-text similarity score above which a pair of sentences is deemed a paraphrase and any score below the threshold means the sentences are not paraphrases. We report performance of the various methods using accuracy (percentage of correct predictions), precision (the percentage of correct predictions out of the positive predictions), recall (percentage of correct predictions out of all true positives), F-measure (harmonic mean of precision and recall), and kappa statistics (a measure of agreement between our method's output and experts' labels while accounting for chance agreement).

We experimented with both word-to-word similarity measures and text-to-text similarity measures. The word-to-word similarity measures were expanded to work at sentence level using the two methods described next: greedy matching and optimal matching. For LDA, we used the word-to-word measure and text-to-text measures described in the previous section. For LSA, we use the cosine between two words' LSA vectors as a measure of word-to-word similarity. For LSA-based text-to-text similarity we first add up the word vectors for all the words in a text thus obtaining two vectors, one for each text, and then compute the cosine between these two text vectors.

Greedy Matching

In the greedy approach words from one sentence (usually the shorter sentence) are greedily matched, one by one, starting from the beginning of the sentence, with the most similar word from the other sentence. In case of duplicates, the order of the words in the two sentences was important such that the first occurrence matches with the first occurrence and so on. To be consistent across all methods presented here and for fairness of comparison across these methods, we require that words must be part of at most one pair.

The greedy method has the advantage of being simple. The obvious drawback of the greedy method is that it does not aim for a global maximum similarity score. The optimal method described next solves this issue.

Optimal Matching

The optimal method aims at finding the best overall word-to-word match, based only on the similarities between words. This is a well-known combinatorial optimization problem. The assignment problem is one of the fundamental combinatorial optimization problems and consists of finding a maximum weight matching in a weighted bipartite graph. Given a complete bipartite graph, $G = (S, T, E)$, with n worker vertices (S), n job vertices (T), and each edge $e_{s \in S, t \in T} \in E$ having a non-negative weight $w(s, t)$ indicating how qualified a worker is for a certain job, the task is to find a matching M from S to T with maximum weight. In case of different numbers of workers or jobs, dummy vertices could be used.

The assignment problem can be formulated as finding a permutation π for which $S_{OPT} = \sum_{i=1}^{n} w(s_i, t_{\pi(i)})$ is maximum. Such an assignment is called optimum assignment. An algorithm, the Kuhn-Munkres method [22, 23], has been proposed that can find a solution to it in polynomial time.

In our case, we model the semantic similarity problem as finding the optimum assignment between words in one text, T_1, and words in another text, T_2, where the fitness between words in the texts can be measured by any word-to-word semantic similarity function. That is, we are after a permutation π for which $S_{OPT} = \sum_{i=1}^{n} word\text{-}sim(v_i, w_{\pi(i)})$ is maximum where *word-sim* can be any word-to-word similarity measure, and v and w are words from the texts T_1 and T_2, respectively. In our case, the word-sim are the word-to-word measures based on LDA and LSA.

Results

A summary of our experiments is shown in Table 1. These are results on MSRP test data obtained using the threshold for similarity that corresponds to the threshold learned from training data. The threshold that led to best accuracy on training data was selected. The threshold varied from method to method. The results obtained using the word-to-word similarity measures are labeled Greedy and Optimal in Table 1. The row labeled LSA shows results obtained when text-level LSA vectors were used, as explained earlier. The *Baseline* method indicates performance when labeling all instances with the dominant label of a true paraphrase. The rest of the rows in the table show results when the text-to-text similarity measures based on various distribution distances were used: IR (Information Radius), Hellinger, and Manhattan.

The LDA-Optimal offers competitive results. It provides best precision and kappa score. As noted from the table, the text-to-text similarity measures based on distribution distances perform close to chance. The problem seems to be rooted in the relative size of texts compared to the number of the topics in the LDA model. In the basic model, we used 300 topics (similar to the 300 dimensions used for LSA) for comparison purposes. The average sentence size in MSRP (after removing stopwords) is 10.3 for training data and 10.4 for testing data. That means that in a typical sentence

Table 1. Results on the MSRP test data

Method	*Accuracy*	*Precision*	*Recall*	*F-Measure*	*Kappa*
Baseline	66.55	66.53	1	79.90	0.22
LSA	**73.56**	75.34	**89.53**	**81.83**	34.61
LSA Greedy	72.86	75.50	87.61	81.11	33.89
LSA Optimal	73.04	76.72	85.35	80.80	35.95
LDA-IR	67.47	67.27	99.47	80.26	4.52
LDA-Hellinger	67.36	67.25	99.21	80.16	4.39
LDA-Manhattan	66.78	67.12	98.08	79.70	3.56
LDA-Greedy	73.04	76.07	86.74	81.05	35.01
LDA-Optimal	73.27	**77.05**	85.17	80.91	**36.74**

Table 2. Topic assignment for instance #23 in MSRP test data

Word	*Topic*	*Word*	*Topic*
senator	8	ashamed	1
Clinton	3	playing	7
ashamed	1	politics	8
playing	7	important	10
politics	8	issue	8
important	9	state	8
issue	8	budget	2
homeland	8	division	11
security	2	spokesman	1
funding	8	Andrew	3
		Rush	5

most of the 300 topics will not be assigned to any word leading to very similar topic distributions over the entire set of 300 topics. Even if the probability for topics that are not assigned to a word in a sentence is set to 0, the distance between two values of 0 is 0 which means the distributions are quite similar. To better illustrate this issue, we use the example below, which is instance #23 in MSRP test data.

Text A: *Senator Clinton should be ashamed of herself for playing politics with the important issue of homeland security funding, he said.*

Text B: *She should be ashamed of herself for playing politics with this important issue, said state budget division spokesman Andrew Rush.*

Table 2 shows the topic assignment for each of the non-stop words in the two sentences when using an LDA model of just 12 topics. We used this time 12 topics as it compares to the relative size of sentences in MSRP. We wanted to check whether using fewer topics somehow reduces the topic sparseness problem in short texts. Even in this case, because different words are assigned to the same topic, e.g. *senator*, *politics*, and *homeland* are all assigned to topic 8 which is about politics, there are some topics that do not occur in the sentence.

Figure 1 shows distributions of the 12 topics for the two sentences in the example above (left) and the two sentences in the example given in Introduction (right). From the figure, we notice that for the example on the right, more than half of the topics have identical probabilities in the two sentences (these are topics that are not assigned

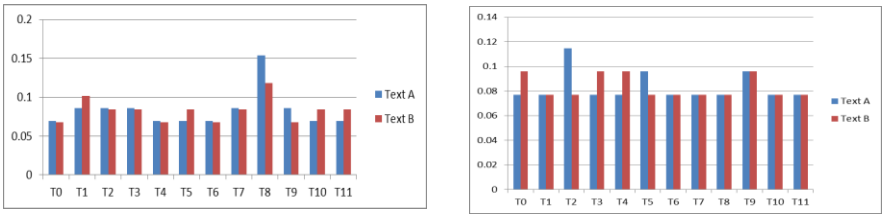

Fig. 1. Examples of topic distributions for the two sentences in instance #23 in MSRP test data (left) and the two sentences in the example given in Introduction (right)

Topic 2:	
number	0.014
money	0.012
system	0.010
business	0.009
information	0.009
special	0.009
set	0.009
job	0.009
amount	0.008
general	0.008

Topic 7:	
day	0.029
good	0.021
thought	0.0194
school	0.017
home	0.017
children	0.015
father	0.014
knew	0.013
told	0.0131
hard	0.011

Topic 8:	
states	0.020
world	0.019
united	0.015
government	0.013
american	0.012
state	0.012
war	0.011
power	0.009
president	0.008
groups	0.007

Fig. 2. Examples of topics and distributions overs words in three topics (top 10 words are shown for each topic)

Table 3. Results on MSRP test data with LDA-based methods for various number of topics

Method	***Accuracy/Kappa (T=300)***	***Accuracy/Kappa (T=40)***	***Accuracy/Kappa (T=12)***	***Accuracy/Kappa (T=6)***
LDA-IR	67.47/4.52	67.94/5.88	66.66/3.11	67.13/9.87
LDA-Hellinger	67.36/4.39	67.82/5.22	67.13/10.62	66.95/13.35
LDA-Manhattan	66.78/3.56	**68.23**/8.12	66.78/8.44	67.01/9.93
LDA-Greedy	73.04/35.01	72.63/34.72	71.94/26.20	71.71/32.36
LDA-Optimal	**73.27/36.74**	**73.50/36.77**	72.11/31.19	71.53/31.63

to any words). This topic sparseness issue leads to short distances between the topic distributions which in turn leads to very high similarity scores (low distance means the texts are close to each other semantically, i.e. high similarity). Figure 2 shows tops ten words in topics assigned to words in Table 2.

To further investigate the impact of number of topics on the behavior of the LDA-based measures, we experimented with LDA models that use 300, 40, 12, and 6 topics. The performance of these models on the MSRP test data is provided in Table 2. While the results slightly improve for the text-to-text similarity measures based on distribution distances the performance is still modest. The best results are obtained for T=40 topics.

5 Discussion, Conclusions, and Future Work

We presented in this paper our work with LDA-based semantic similarity measures. The conclusion of our investigation is that the word-to-word LDA-based measure (defined as the dot-product between the corresponding topic vectors) leads to competitive results. The proposed measure outperforms the algebraic method of LSA, in particular when combined with the Optimal matching method. As noted, LDA has the conceptual advantage of handling polysemy.

The distance-based similarity measures suffer from a data sparseness problem which could be alleviated if the number of topics used in the underlying Latent Dirichlet Allocation model is reduced to a number that is comparable or smaller than the average length of the texts for which a semantic similarity judgment is sought. We plan to further investigate and address the topic sparseness issue for the similarity measures based on distance among distributions. For instance, we would like to explore methods that try to optimize topic coherence, e.g. based on pointwise mutual information (PMI; [24]).

Another future line of work we would like to pursue is about combining LDA's topic distribution in a document with LSA-based word-to-word similarity. For instance, we would apply and use the LSA word-to-word similarity between two words only when an LDA model assigns the same topic to the two words, i.e. we would rely on the LSA score only when LDA indicates the two words have the same meaning as represented by an LDA topic.

Acknowledgments. This research was supported in part by Institute for Education Sciences under awards R305A100875. Any opinions, findings, and conclusions or recommendations expressed in this material are solely the authors' and do not necessarily reflect the views of the sponsoring agencies.

References

1. Ibrahim, A., Katz, B., Lin, J.: Extracting structural paraphrases from aligned monolingual corpora. In: Proceedings of the Second International Workshop on Paraphrasing (ACL 2003)
2. Iordanskaja, L., Kittredge, R., Polgere, A.: Lexical selection and paraphrase in a meaning-text generation model. In: Natural Language Generation in Artificial Intelligence and Computational Linguistics. Kluwer Academic (1991)
3. Graesser, A.C., Olney, A., Haynes, B.C., Chipman, P.: Autotutor: A cognitive system that simulates a tutor that facilitates learning through mixed-initiative dialogue. In: Cognitive Systems: Human Cognitive Models in Systems Design. Erlbaum, Mahwah (2005)
4. Rus, V., Graesser, A.C.: Deeper natural language processing for evaluating student answers in intelligent tutoring systems. In: Paper Presented at the Annual Meeting of the American Association of Artificial Intelligence (AAAI 2006), Boston, MA, July 16-20 (2006)
5. Rus, V., Nan, X., Shiva, S., Chen, Y.: Clustering of Defect Reports Using Graph Partitioning Algorithms. In: Proceedings of the 20th International Conference on Software and Knowledge Engineering, Boston, MA, July 2-4 (2009)
6. Fernando, S., Stevenson, M.: A semantic similarity approach to paraphrase detection. In: Proceedings of the Computational Linguistics UK, CLUK 2008 (2008)

7. Lintean, M., Rus, V.: Measuring Semantic Similarity in Short Texts through Greedy Pairing and Word Semantics. In: Proceedings of the 25th International Florida Artificial Intelligence Research Society Conference, Marco Island, FL (2012)
8. Dolan, B., Quirk, C., Brockett, C.: Unsupervised Construction of Large Paraphrase Corpora: Exploiting Massively Parallel News Sources. In: COLING 2004 (2004)
9. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. The Journal of Machine Learning Research 3, 993–1022 (2003)
10. Landauer, T., McNamara, D.S., Dennis, S., Kintsch, W.: Handbook of Latent Semantic Analysis. Erlbaum, Mahwah (2007)
11. Miller, G.: Wordnet: a lexical database for English. Communications of the ACM 38(11), 39–41 (1995)
12. Pedersen, T., Patwardhan, S., Michelizzi, J.: WordNet: Similarity-Measuring the Relatedness of Concepts. In: The Proceedings of the Nineteenth National Conference on Artificial Intelligence (AAAI 2004), San Jose, CA (Intelligent Systems Demonstration), July 25-29, pp. 1024–1025 (2004)
13. Hirst, G., Stonge, D.: Lexical chains as representations of context for the detection and correction of malapropisms. In: Fellbaum, C. (ed.) WordNet: An Electronic Lexical Database. MIT Press (1998)
14. Banerjee, S., Pedersen, T.: Extended gloss overlaps as a measure of semantic relatedness. In: Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, pp. 805–810 (2003)
15. Patwardhan, S.: Incorporating dictionary and corpus information into a context vector measure of semantic relatedness. Master's thesis, Univ. of Minnesota, Duluth (2003)
16. Rus, V., Lintean, M., Graesser, A., McNamara, D.: Assessing Student Paraphrases Using Lexical Semantics and Word Weighting. In: Proceedings of the 14th International Conference on Artificial Intelligence in Education, Brighton, UK (2009)
17. Dagan, I., Glickman, O., Magnini, B.: The PASCAL Recognizing Textual Entailment Challenge. In: Proceedings of the Recognizing Textual Entailment Challenge Workshop (2005)
18. Lintean, M., Moldovan, C., Rus, V., McNamara, D.: The Role of Local and Global Weighting in Assessing the Semantic Similarity of Texts Using Latent Semantic Analysis. In: Proceedings of the 23rd International Florida Artificial Intelligence Research Society Conference, Daytona Beach, FL (2010)
19. Kozareva, Z., Montoyo, A.: Paraphrase Identification on the Basis of Supervised Machine Learning Techniques. In: Salakoski, T., Ginter, F., Pyysalo, S., Pahikkala, T. (eds.) FinTAL 2006. LNCS (LNAI), vol. 4139, pp. 524–533. Springer, Heidelberg (2006)
20. Celikyilmaz, A., Hakkani-Tür, D., Tur, G.: LDA Based Similarity Modeling for Question Answering. In: NAACL-HLT, Workshop on Semantic Search, Los Angeles, CA (June 2010)
21. Chen, X., Li, L., Xiao, H., Xu, G., Yang, Z., Kitsuregawa, M.: Recommending Related Microblogs: A Comparison between Topic and WordNet based Approaches. In: Proceedings of the 26th International Conference on Artificial Intelligence (2012)
22. Kuhn, H.W.: The Hungarian Method for the assignment problem. Naval Research Logistics Quarterly 2, 83–97 (1955)
23. Munkres, J.: Algorithms for the assignment and transportation problems. Journal of the Society for Industrial and Applied Mathematics 5(1), 32–38 (1957)
24. Newman, D., Lau, J.H., Grieser, K., Baldwin, T.: Automatic evaluation of topic coherence. In: Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics, Los Angeles, United States, pp. 100–108 (2010)