Random Walks for Text Semantic Similarity

Daniel Ramage, Anna N. Rafferty, and Christopher D. Manning
Computer Science Department
Stanford University
Stanford, CA 94305
{dramage,manning}@cs.stanford.edu
rafferty@eecs.berkeley.edu

Abstract

Many tasks in NLP stand to benefit from robust measures of semantic similarity for units above the level of individual words. Rich semantic resources such as WordNet provide local semantic information at the lexical level. However, effectively combining this information to compute scores for phrases or sentences is an open problem. Our algorithm aggregates local relatedness information via a random walk over a graph constructed from an underlying lexical resource. The stationary distribution of the graph walk forms a "semantic signature" that can be compared to another such distribution to get a relatedness score for texts. On a paraphrase recognition task, the algorithm achieves an 18.5% relative reduction in error rate over a vector-space baseline. We also show that the graph walk similarity between texts has complementary value as a feature for recognizing textual entailment, improving on a competitive baseline system.

1 Introduction

Many natural language processing applications must directly or indirectly assess the semantic similarity of text passages. Modern approaches to information retrieval, summarization, and textual entailment, among others, require robust numeric relevance judgments when a pair of texts is provided as input. Although each task demands its own scoring criteria, a simple lexical overlap measure such as cosine similarity of document vectors can often serve as a surprisingly powerful baseline. We argue that there is room to improve these general-purpose similarity measures, particularly for short text passages.

Most approaches fall under one of two categories. One set of approaches attempts to explicitly account for fine-grained structure of the two passages, e.g. by aligning trees or constructing logical forms for theorem proving. While these approaches have the potential for high precision on many examples, errors in alignment judgments or formula construction are often insurmountable. More broadly, it's not always clear that there is a correct alignment or logical form that is most appropriate for a particular sentence pair. The other approach tends to ignore structure, as canonically represented by the vector space model, where any lexical item in common between the two passages contributes to their similarity score. While these approaches often fail to capture distinctions imposed by, e.g. negation, they do correctly capture a broad notion of similarity or aboutness.

This paper presents a novel variant of the vector space model of text similarity based on a random walk algorithm. Instead of comparing two bags-of-words directly, we compare the distribution each text induces when used as the seed of a random walk over a graph derived from WordNet and corpus statistics. The walk posits the existence of a distributional particle that roams the graph, biased toward the neighborhood surrounding an input bag of words. Eventually, the walk reaches a stationary distribution over all nodes in the graph, smoothing the peaked input distribution over a much larger semantic space. Two such stationary distributions can be compared using conventional measures of vector similarity, producing a final relatedness score.

This paper makes the following contributions. We present a novel random graph walk algorithm

Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing, ACL-IJCNLP 2009, pages 23–31,
Suntec, Singapore, 7 August 2009.

Word	Step 1	Step 2	Step 3	Conv.
eat	3	8	9	9
corrode	10	33	53	>100
pasta	–	2	3	5
dish	–	4	5	6
food	–	–	21	12
solid	–	–	–	26

Table 1: Ranks of sample words in the distribution for *I ate a salad and spaghetti* after a given number of steps and at convergence. Words in the vector are ordered by probability at time step t; the word with the highest probability in the vector has rank 1. "–" indicates that node had not yet been reached.

for semantic similarity of texts, demonstrating its efficiency as compared to a much slower but mathematically equivalent model based on summed similarity judgments of individual words. We show that walks effectively aggregate information over multiple types of links and multiple input words on an unsupervised paraphrase recognition task. Furthermore, when used as a feature, the walk's semantic similarity score can improve the performance of an existing, competitive textual entailment system. Finally, we provide empirical results demonstrating that indeed, each step of the random walk contributes to its ability to assess paraphrase judgments.

2 A random walk example

To provide some intuition about the behavior of the random walk on text passages, consider the following example sentence: *I ate a salad and spaghetti.*

No measure based solely on lexical identity would detect overlap between this sentence and another input consisting of only the word *food*. But if each text is provided as input to the random walk, local relatedness links from one word to another allow the distributional particle to explore nearby parts of the semantic space. The number of non-zero elements in both vectors increases, eventually converging to a stationary distribution for which both vectors have many shared non-zero entries.

Table 1 ranks elements of the sentence vector based on their relative weights. Observe that at the beginning of the walk, *corrode* has a high rank due to its association with the WordNet sense of *eat* corresponding to eating away at something. However, because this concept is not closely linked with other words in the sentence, its relative rank drops as the distribution converges and other word senses more related to *food* are pushed up. The random walk allows the meanings of words to reinforce one another. If the sentence above had ended with *drank wine* rather than *spaghetti*, the final weight on the *food* node would be smaller since fewer input words would be as closely linked to *food*. This matches the intuition that the first sentence has more to do with food than does the second, although both walks should and do give some weight to this node.

3 Related work

Semantic relatedness for individual words has been thoroughly investigated in previous work. Budanitsky and Hirst (2006) provide an overview of many of the knowledge-based measures derived from WordNet, although other data sources have been used as well. Hughes and Ramage (2007) is one such measure based on random graph walks.

Prior work has considered random walks on various text graphs, with applications to query expansion (Collins-Thompson and Callan, 2005), email address resolution (Minkov and Cohen, 2007), and word-sense disambiguation (Agirre and Soroa, 2009), among others.

Measures of similarity have also been proposed for sentence or paragraph length text passages. Mihalcea et al. (2006) present an algorithm for the general problem of deciding the similarity of meaning in two text passages, coining the name "text semantic similarity" for the task. Corley and Mihalcea (2005) apply this algorithm to paraphrase recognition.

Previous work has shown that similarity measures can have some success as a measure of textual entailment. Glickman et al. (2005) showed that many entailment problems can be answered using only a bag-of-words representation and web co-occurrence statistics. Many systems integrate lexical relatedness and overlap measures with deeper semantic and syntactic features to create improved results upon relatedness alone, as in Montejo-Ráez et al. (2007).

4 Random walks on lexical graphs

In this section, we describe the mechanics of computing semantic relatedness for text passages

based on the random graph walk framework. The algorithm underlying these computations is related to topic-sensitive PageRank (Haveliwala, 2002); see Berkhin (2005) for a survey of related algorithms.

To compute semantic relatedness for a pair of passages, we compare the stationary distributions of two Markov chains, each with a state space defined over all lexical items in an underlying corpus or database. Formally, we define the probability of finding the particle at a node n_i at time t as:

$$n_i^{(t)} = \sum_{n_j \in V} n_j^{(t-1)} P(n_i \mid n_j)$$

where $P(n_i \mid n_j)$ is the probability of transitioning from n_j to n_i at any time step. If those transitions bias the particle to the neighborhood around the words in a text, the particle's distribution can be used as a lexical signature.

To compute relatedness for a pair of texts, we first define the graph nodes and transition probabilities for the random walk Markov chain from an underlying lexical resource. Next, we determine an initial distribution over that state space for a particular input passage of text. Then, we simulate a random walk in the state space, biased toward the initial distribution, resulting in a passage-specific distribution over the graph. Finally, we compare the resulting stationary distributions from two such walks using a measure of distributional similarity. The remainder of this section discusses each stage in more detail.

4.1 Graph construction

We construct a graph $G = (V, E)$ with vertices V and edges E extracted from WordNet 3.0. WordNet (Fellbaum, 1998) is an annotated graph of synsets, each representing one concept, that are populated by one or more words. The set of vertices extracted from the graph is all synsets present in WordNet (e.g. *foot#n#1* meaning the part of the human leg below the ankle), all part-of-speech tagged words participating in those synsets (e.g. *foot#n* linking to *foot#n#1* and *foot#n#2* etc.), and all untagged words (e.g. *foot* linking to *foot#n* and *foot#v*). The set of edges connecting synset nodes is all inter-synset edges contained in WordNet, such as hyponymy, synonomy, antonymy, etc., except for regional and usage links. All WordNet relational edges are given uniform weight. Edges also connect each part-of-speech tagged word to all synsets it takes part in, and from each word to all its part-of-speech. These edge weights are derived from corpus counts as in Hughes and Ramage (2007). We also included a low-weight self-loop for each node.

Our graph has 420,253 nodes connected by 1,064,464 edges. Because synset nodes do not link outward to part-of-speech tagged nodes or word nodes in this graph, only the 117,659 synset nodes have non-zero probability in every random walk—i.e. the stationary distribution will always be non-zero for these 117,659 nodes, but will be non-zero for only a subset of the remainder.

4.2 Initial distribution construction

The next step is to seed the random walk with an initial distribution over lexical nodes specific to the given sentence. To do so, we first tag the input sentence with parts-of-speech and lemmatize each word based on the finite state transducer of Minnen et al. (2001). We search over consecutive words to match multi-word collocation nodes found in the graph. If the word or its lemma is part of a sequence that makes a complete collocation, that collocation is used. If not, the word or its lemma with its part of speech tag is used if it is present as a graph node. Finally, we fall back to the surface word form or underlying lemma form without part-of-speech information if necessary. For example, the input sentence: *The boy went with his dog to the store,* would result in mass being assigned to underlying graph nodes *boy#n, go_with, he, dog#n, store#n*.

Term weights are set with $tf.idf$ and then normalized. Each term's weight is proportional to the number of occurrences in the sentence times the log of the number of documents in some corpus divided by the number of documents containing that term. Our idf counts were derived from the English Gigaword corpus 1994-1999.

4.3 Computing the stationary distribution

We use the power iteration method to compute the stationary distribution for the Markov chain. Let the distribution over the N states at time step t of the random walk be denoted $\vec{v}^{(t)} \in \mathbb{R}^N$, where $\vec{v}^{(0)}$ is the initial distribution as defined above. We denote the column-normalized state-transition matrix as $M \in \mathbb{R}^{N \times N}$. We compute the stationary distribution of the Markov chain with probability β of returning to the initial distribution at each

time step as the limit as $t \to \infty$ of:

$$\vec{v}^{(t)} = \beta\vec{v}^{(0)} + (1-\beta)M\vec{v}^{(t-1)}$$

In practice, we test for convergence by examining if $\sum_{i=1}^{N} \|v_i^{(t)} - v_i^{(t-1)}\| < 10^{-6}$, which in our experiments was usually after about 50 iterations.

Note that the resulting stationary distribution can be factored as the weighted sum of the stationary distributions of each word represented in the initial distribution. Because the initial distribution $\vec{v}^{(0)}$ is a normalized weighted sum, it can be re-written as $\vec{v}^{(0)} = \sum_k \gamma_k \cdot \vec{w}_k^{(0)}$ for $\vec{w}_k$ having a point mass at some underlying node in the graph and with γ_k positive such that $\sum_k \gamma_k = 1$. A simple proof by induction shows that the stationary distribution $\vec{v}^{(\infty)}$ is itself the weighted sum of the stationary distribution of each underlying word, i.e. $\vec{v}^{\infty} = \sum_k \gamma_k \cdot \vec{w}_k^{(\infty)}$.

In practice, the stationary distribution for a passage of text can be computed from a single specially-constructed Markov chain. The process is equivalent to taking the weighted sum of every word type in the passage computed independently. Because the time needed to compute the stationary distribution is dominated by the sparsity pattern of the walk's transition matrix, the computation of the stationary distribution for the passage takes a fraction of the time needed if the stationary distribution for each word were computed independently.

4.4 Comparing stationary distributions

In order to get a final relatedness score for a pair of texts, we must compare the stationary distribution from the first walk with the distribution from the second walk. There exist many measures for computing a final similarity (or divergence) measure from a pair of distributions, including geometric measures, information theoretic measures, and probabilistic measures. See, for instance, the overview of measures provided in Lee (2001).

In system development on training data, we found that most measures were reasonably effective. For the rest of this paper, we report numbers using cosine similarity, a standard measure in information retrieval; Jensen-Shannon divergence, a commonly used symmetric measure based on KL-divergence; and the dice measure extended to weighted features (Curran, 2004). A summary of these measures is shown in Table 2. Justification for the choice of these three measures is discussed in Section 6.

Cosine	$\frac{\vec{x}\cdot\vec{y}}{\|\vec{x}\|_2\|\vec{y}\|_2}$
Jensen-Shannon	$\frac{1}{2}D(x\|\frac{x+y}{2}) + \frac{1}{2}D(y\|\frac{x+y}{2})$
Dice	$\frac{2\sum_i \min(x_i,y_i)}{\sum_i x_i + \sum_i y_i}$

Table 2: Three measures of distributional similarity between vectors $\vec{x}$ and $\vec{y}$ used to compare the stationary distributions from passage-specific random walks. $D(p\|q)$ is KL-divergence, defined as $\sum_i p_i \log \frac{p_i}{q_i}$.

5 Evaluation

We evaluate the system on two tasks that might benefit from semantic similarity judgments: paraphrase recognition and recognizing textual entailment. A complete solution to either task will certainly require tools more tuned to linguistic structure; the paraphrase detection evaluation argues that the walk captures a useful notion of semantics at the sentence level. The entailment system evaluation demonstrates that the walk score can improve a larger system that does make use of more fine-grained linguistic knowledge.

5.1 Paraphrase recognition

The Microsoft Research (MSR) paraphrase data set (Dolan et al., 2004) is a collection of 5801 pairs of sentences automatically collected from newswire over 18 months. Each pair was hand-annotated by at least two judges with a binary yes/no judgment as to whether one sentence was a valid paraphrase of the other. Annotators were asked to judge whether the meanings of each sentence pair were reasonably equivalent. Inter-annotator agreement was 83%. However, 67% of the pairs were judged to be paraphrases, so the corpus does not reflect the rarity of paraphrases in the wild. The data set comes pre-split into 4076 training pairs and 1725 test pairs.

Because annotators were asked to judge if the meanings of two sentences were equivalent, the paraphrase corpus is a natural evaluation testbed for measures of semantic similarity. Mihalcea et al. (2006) defines a measure of text semantic similarity and evaluates it in an unsupervised paraphrase detector on this data set. We present their

algorithm here as a strong reference point for semantic similarity between text passages, based on similar underlying lexical resources.

The Mihalcea et al. (2006) algorithm is a wrapper method that works with any underlying measure of lexical similarity. The similarity of a pair of texts T_1 and T_2, denoted as $sim_m(T_1, T_2)$, is computed as:

$$sim_m(T_1, T_2) = \frac{1}{2} f(T_1, T_2) + \frac{1}{2} f(T_2, T_1)$$

$$f(T_a, T_b) = \frac{\sum_{w \in T_a} maxSim(w, T_b) \cdot idf(w)}{\sum_{w \in T_a} idf(w)}$$

where the $maxSim(w, T)$ function is defined as the maximum similarity of the word w within the text T as determined by an underlying measure of lexical semantic relatedness. Here, $idf(w)$ is defined as the number of documents in a background corpus divided by the number of documents containing the term. $maxSim$ compares only within the same WordNet part-of-speech labeling in order to support evaluation with lexical relatedness measures that cannot cross part-of-speech boundaries.

Mihalcea et al. (2006) presents results for several underlying measures of lexical semantic relatedness. These are subdivided into corpus-based measures (using Latent Semantic Analysis (Landauer et al., 1998) and a pointwise-mutual information measure) and knowledge-based resources driven by WordNet. The latter include the methods of Jiang and Conrath (1997), Lesk (1986), Resnik (1999), and others.

In this unsupervised experimental setting, we consider using only a thresholded similarity value from our system and from the Mihalcea algorithm to determine the paraphrase or non-paraphrase judgment. For consistency with previous work, we threshold at 0.5. Note that this threshold could be tuned on the training data in a supervised setting. Informally, we observed that on the training data a threshold of near 0.5 was often a good choice for this task.

Table 3 shows the results of our system and a representative subset of those reported in (Mihalcea et al., 2006). All the reported measures from both systems do a reasonable job of paraphrase detection – the majority of pairs in the corpus are deemed paraphrases when the similarity measure is thresholded at 0.5, and indeed this is reasonable given the way in which the data were collected. The first three rows are the performance of the similarity judgments output by our walk under three different distributional similarity measures (cosine, dice, and Jensen-Shannon), with the walk score using the dice measure outperforming all other systems on both accuracy and macro-averaged F_1. The output of the Mihalcea system using a representative subset of underlying lexical measures is reported in the second and third segments. The fourth segment reports the results of baseline methods—the vector space similarity measure is cosine similarity among vectors using $tf.idf$ weighting, and the random baseline chooses uniformly at random, both as reported in (Mihalcea et al., 2006). We add the additional baseline of always guessing the majority class label because the data set is skewed toward "paraphrase."

System	Acc.	F_1: c_1	F_1: c_0	Macro F_1
Random Graph Walk				
Walk (Cosine)	0.687	0.787	0.413	0.617
Walk (Dice)	**0.708**	0.801	**0.453**	**0.645**
Walk (JS)	0.688	0.805	0.225	0.609
Mihalcea et. al., Corpus-based				
PMI-IR	0.699	**0.810**	0.301	0.625
LSA	0.684	0.805	0.170	0.560
Mihalcea et. al., WordNet-based				
J&C	0.693	0.790	0.433	0.629
Lesk	0.693	0.789	0.439	0.629
Resnik	0.690	0.804	0.254	0.618
Baselines				
Vector-based	0.654	0.753	0.420	0.591
Random	0.513	0.578	0.425	0.518
Majority (c_1)	0.665	0.799	—	0.399

Table 3: System performance on 1725 examples of the MSR paraphrase detection test set. Accuracy (micro-averaged F_1), F_1 for c_1 "paraphrase" and c_0 "non-paraphrase" classes, and macro-averaged F_1 are reported.

In an unbalanced data setting, it is important to consider more than just accuracy and F_1 on the majority class. We report accuracy, F_1 for each class label, and the macro-averaged F_1 on all systems. F_1: c_0 and Macro-F_1 are inferred for the system variants reported in (Mihalcea et al., 2006). Micro-averaged F_1 in this context is equivalent to accuracy (Manning et al., 2008).

Mihalcea also reports a combined classifier which thresholds on the simple average of the individual classifiers, resulting in the highest numbers reported in that work, with accuracy of 0.703, "paraphrase" class F_1: $c_1 = 0.813$, and inferred Macro $F_1 = 0.648$. We believe that the scores

Data Set	Cosine	Dice	Jensen-Shannon
RTE2_dev	55.00	51.75	**55.50**
RTE2_test	57.00	54.25	**57.50**
RTE3_dev	**59.00**	57.25	**59.00**
RTE3_test	55.75	55.75	**56.75**

Table 4: Accuracy of entailment detection when thresholding the text similarity score output by the random walk.

Data Set	Baseline	Cosine	Dice	JS
RTE2_dev	66.00	**66.75**	65.75	66.25
RTE2_test	63.62	**64.50**	63.12	63.25
RTE3_dev	70.25	70.50	**70.62**	70.38
RTE3_test	65.44	**65.82**	65.44	65.44

Table 5: Accuracy when the random walk is added as a feature of an existing RTE system (left column) under various distance metrics (right columns).

from the various walk measures might also improve performance when in a combination classifier, but without access to the individual judgments in that system we are unable to evaluate the claim directly. However, we did create an upper bound reference by combining the walk scores with easily computable simple surface statistics. We trained a support vector classifier on the MSR paraphrase training set with a feature space consisting of the walk score under each distributional similarity measure, the length of each text, the difference between those lengths, and the number of unigram, bigram, trigram, and four-gram overlaps between the two texts. The resulting classifier achieved accuracy of 0.719 with $F_1{:}\, c_1 = 0.807$ and $F_1{:}\, c_0 = 0.487$ and Macro $F_1 = 0.661$. This is a substantial improvement, roughly on the same order of magnitude as from switching to the best performing distributional similarity function.

Note that the running time of the Mihalcea et al. algorithm for comparing texts T_1 and T_2 requires $|T_1| \cdot |T_2|$ individual similarity judgments. By contrast, this work allows semantic profiles to be constructed and evaluated for each text in a single pass, independent of the number of terms in the texts.

The performance of this unsupervised application of walks to paraphrase recognition suggests that the framework captures important intuitions about similarity in text passages. In the next section, we examine the performance of the measure embedded in a larger system that seeks to make fine-grained entailment judgments.

5.2 Textual entailment

The Recognizing Textual Entailment Challenge (Dagan et al., 2005) is a task in which systems assess whether a sentence is entailed by a short passage or sentence. Participants have used a variety of strategies beyond lexical relatedness or overlap for the task, but some have also used only relatively simple similarity metrics. Many systems incorporate a number of these strategies, so we experimented with using the random walk to improve an existing RTE system. This addresses the fact that using similarity alone to detect entailment is impoverished: entailment is an asymmetric decision while similarity is necessarily symmetric. However, we also experiment with thresholding random walk scores as a measure of entailment to compare to other systems and provide a baseline for whether the walk could be useful for entailment detection.

We tested performance on the development and test sets for the Second and Third PASCAL RTE Challenges (Bar-Haim et al., 2006; Giampiccolo et al., 2007). Each of these data sets contains 800 pairs of texts for which to determine entailment. In some cases, no words from a passage appear in WordNet, leading to an empty vector. In this case, we use the Levenshtein string similarity measure between the two texts; this fallback is used in fewer than five examples in any of our data sets (Levenshtein, 1966).

Table 4 shows the results of using the similarity measure alone to determine entailment; the system's ability to recognize entailment is above chance on all data sets. Since the RTE data sets are balanced, we used the median of the random walk scores for each data set as the threshold rather than using an absolute threshold. While the measure does not outperform most RTE systems, it does outperform some systems that used only lexical overlap such as the Katrenko system from the second challenge (Bar-Haim et al., 2006). These results show that the measure is somewhat sensitive to the distance metric chosen, and that the best distance metric may vary by application.

To test the random walk's value for improving an existing RTE system, we incorporated the walk as a feature of the Stanford RTE system (Chambers et al., 2007). This system computes

a weighted sum of a variety of features to make an entailment decision. We added the random walk score as one of these features and scaled it to have a magnitude comparable to the other features; other than scaling, there was no system-specific engineering to add this feature.

As shown in Table 5, adding the random walk feature improves the original RTE system. Thus, the random walk score provides meaningful evidence for detecting entailment that is not subsumed by other information, even in a system with several years of feature engineering and competitive performance. In particular, this RTE system contains features representing the alignment score between two passages; this score is composed of a combination of lexical relatedness scores between words in each text. The ability of the random walk to add value to the system even given this score, which contains many common lexical relatedness measures, suggests we are able to extract text similarity information that is distinct from other measures. To put the gain we achieve in perspective, an increase in the Stanford RTE system's score of the same magnitude would have moved the system's two challenge entries from 7th and 25th to 6th and 17th, respectively, in the second RTE Challenge. It is likely the gain from this feature could be increased by closer integration with the system and optimizing the initial distribution creation for this task.

By using the score as a feature, the system is able to take advantage of properties of the score distribution. While Table 4 shows performance when a threshold is picked a priori, experimenting with that threshold increases performance by over two percent. By lowering the threshold (classifying more passages as entailments), we increase recall of entailed pairs without losing as much precision in non-entailed pairs since many have very low scores. As a feature, this aspect of the score distribution can be incorporated by the system, but it cannot be used in a simple thresholding design.

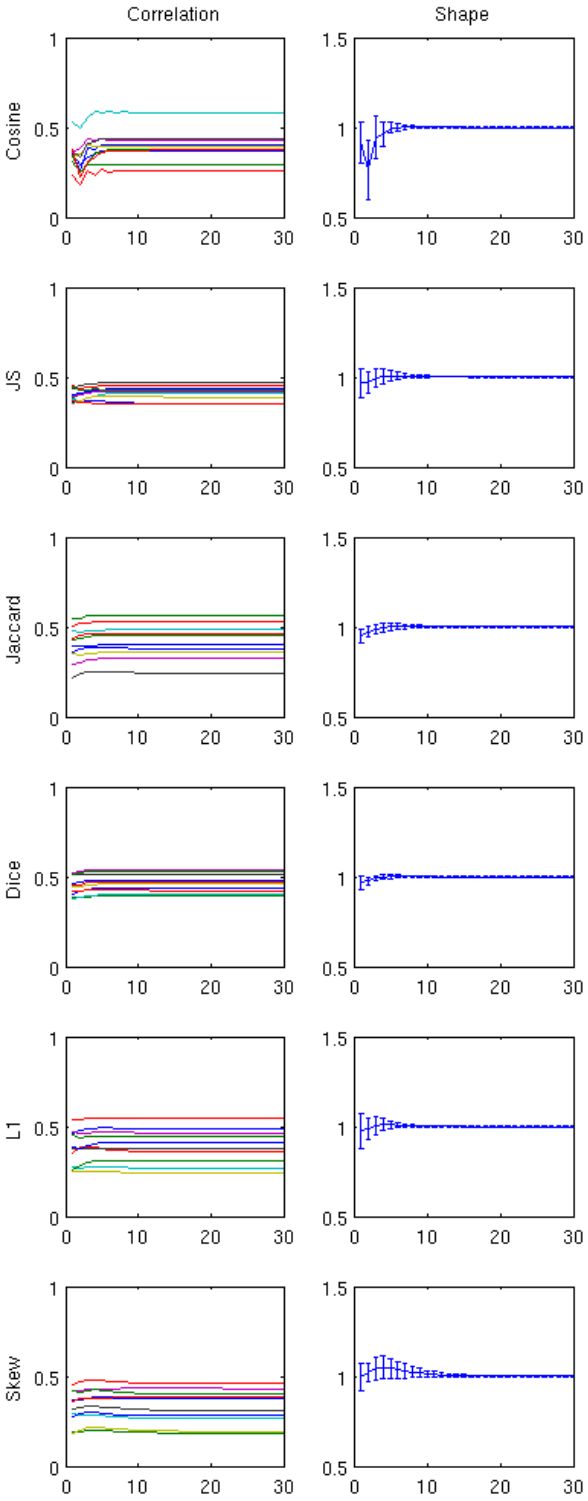

Figure 1: Impact of number of walk steps on correlation with MSR paraphrase judgments. The left column shows absolute correlation across ten resampled runs (y-axis) versus number of steps taken (x-axis). The right column plots the mean ratio of performance at step t (x-axis) versus performance at convergence.

6 Discussion

The random walk framework smoothes an initial distribution of words into a much larger lexical space. In one sense, this is similar to the technique of query expansion used in information retrieval. A traditional query expansion model extends a bag of words (usually a query) with additional related words. In the case of pseudo-relevance feedback,

these words come from the first documents returned by the search engine, but other modes of selecting additional words exist. In the random walk framework, this expansion is analogous to taking only a single step of the random walk. Indeed, in the case of the translation model introduced in (Berger and Lafferty, 1999), they are mathematically equivalent. However, we have argued that the walk is an effective global aggregator of relatedness information. We can formulate the question as an empirical one—does simulating the walk until convergence really improve our representation of the text document?

To answer this question, we extracted a 200 items subset of the MSR training data and truncated the walk at each time step up until our convergence threshold was reached at around 50 iterations. We then evaluated the correlation of the walk score with the correct label from the MSR data for 10 random resamplings of 66 documents each. Figure 1 plots this result for different distributional similarity measures. We observe that as the number of steps increases, performance under most of the distributional similarity measures improves, with the exception of the asymmetric skew-divergence measure introduced in (Lee, 2001).

This plot also gives some insight into the qualitative nature of the stability of the various distributional measures for the paraphrase task. For instance, we observe that the Jensen-Shannon score and dice score tend to be the most consistent between runs, but the dice score has a slightly higher mean. This explains in part why the dice score was the best performing measure for the task. In contrast, cosine similarity was observed to perform poorly here, although it was found to be the best measure when combined with our textual entailment system. We believe this discrepancy is due in part to the feature scaling issues described in section 5.2.

7 Final remarks

Notions of similarity have many levels of granularity, from general metrics for lexical relatedness to application-specific measures between text passages. While lexical relatedness is well studied, it is not directly applicable to text passages without some surrounding environment. Because this work represents words and passages as interchangeable mathematical objects (teleport vectors), our approach holds promise as a general framework for aggregating local relatedness information between words into reliable measures between text passages.

The random walk framework can be used to evaluate changes to lexical resources because it covers the entire scope of a resource: the whole graph is leveraged to construct the final distribution, so changes to any part of the graph are reflected in each walk. This means that the meaningfulness of changes in the graph can be evaluated according to how they affect these text similarity scores; this provides a more semantically relevant evaluation of updates to a resource than, for example, counting how many new words or links between words have been added. As shown in Jarmasz and Szpakowicz (2003), an updated resource may have many more links and concepts but still have similar performance on applications as the original. Evaluations of WordNet extensions, such as those in Navigli and Velardi (2005) and Snow et al. (2006), are easily conducted within the framework of the random walk.

The presented framework for text semantic similarity with random graph walks is more general than the WordNet-based instantiation explored here. Transition matrices from alternative linguistic resources such as corpus co-occurrence statistics or larger knowledge bases such as Wikipedia may very well add value as a lexical resource underlying the walk. One might also consider tailoring the output of the walk with machine learning techniques like those presented in (Minkov and Cohen, 2007).

References

E. Agirre and A. Soroa. 2009. Personalizing pagerank for word sense disambiguation. In *EACL*, Athens, Greece.

R. Bar-Haim, I. Dagan, B. Dolan, L. Ferro, D. Giampiccolo, B. Magnini, and I. Szpektor. 2006. The 2nd PASCAL recognizing textual entailment challenge. In *PASCAL Challenges Workshop on RTE*.

A. Berger and J. Lafferty. 1999. Information retrieval as statistical translation. *SIGIR 1999*, pages 222–229.

P. Berkhin. 2005. A survey on pagerank computing. *Internet Mathematics*, 2(1):73–120.

A. Budanitsky and G. Hirst. 2006. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47.

N. Chambers, D. Cer, T. Grenager, D. Hall, C. Kiddon, B. MacCartney, M. de Marneffe, D. Ramage, E. Yeh, and C. D. Manning. 2007. Learning alignments and leveraging natural logic. In *ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*.

K. Collins-Thompson and J. Callan. 2005. Query expansion using random walk models. In *CIKM '05*, pages 704–711, New York, NY, USA. ACM Press.

C. Corley and R. Mihalcea. 2005. Measuring the semantic similarity of texts. In *ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*, pages 13–18, Ann Arbor, Michigan, June. ACL.

J. R. Curran. 2004. *From Distributional to Semantic Similarity*. Ph.D. thesis, University of Edinburgh.

I. Dagan, O. Glickman, and B. Magnini. 2005. The PASCAL recognizing textual entailment challenge. In Quinonero-Candela et al., editor, *MLCW 2005, LNAI Volume 3944*, pages 177–190. Springer-Verlag.

B. Dolan, C. Quirk, and C. Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Coling 2004*, pages 350–356, Geneva, Switzerland, Aug 23–Aug 27. COLING.

C. Fellbaum. 1998. *WordNet: An electronic lexical database.* MIT Press.

D. Giampiccolo, B. Magnini, I. Dagan, and B. Dolan. 2007. The 3rd PASCAL Recognizing Textual Entailment Challenge. In *ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 1–9, Prague, June.

O. Glickman, I. Dagan, and M. Koppel. 2005. Web based probabilistic textual entailment. In *PASCAL Challenges Workshop on RTE*.

T. H. Haveliwala. 2002. Topic-sensitive pagerank. In *WWW '02*, pages 517–526, New York, NY, USA. ACM.

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

M. Jarmasz and S. Szpakowicz. 2003. Roget's thesaurus and semantic similarity. In *Proceedings of RANLP-03*, pages 212–219.

J. J. Jiang and D. W. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *ROCLING X*, pages 19–33.

T.K. Landauer, P.W. Foltz, and D. Laham. 1998. An introduction to latent semantic analysis. *Discourse Processes*, 25(2-3):259–284.

L. Lee. 2001. On the effectiveness of the skew divergence for statistical language analysis. In *Artificial Intelligence and Statistics 2001*, pages 65–72.

M. Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. *ACM SIGDOC: Proceedings of the 5th Annual International Conference on Systems Documentation*, 1986:24–26.

V. I. Levenshtein. 1966. *Binary Codes Capable of Correcting Deletions, Insertions, and Reversals*. Ph.D. thesis, Soviet Physics Doklady.

C. Manning, P. Raghavan, and H. Schutze, 2008. *Introduction to information retrieval*, pages 258–263. Cambridge University Press.

R. Mihalcea, C. Corley, and C. Strapparava. 2006. Corpus-based and knowledge-based measures of text semantic similarity. *AAAI 2006*, 6.

E. Minkov and W. W. Cohen. 2007. Learning to rank typed graph walks: Local and global approaches. In *WebKDD and SNA-KDD joint workshop 2007*.

G. Minnen, J. Carroll, and D. Pearce. 2001. Applied morphological processing of English. *Natural Language Engineering*, 7(03):207–223.

A. Montejo-Ráez, J.M. Perea, F. Martínez-Santiago, M. A. García-Cumbreras, M. M. Valdivia, and A. Ureña López. 2007. Combining lexical-syntactic information with machine learning for recognizing textual entailment. In *ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 78–82, Prague, June. ACL.

R. Navigli and P. Velardi. 2005. Structural semantic interconnections: A knowledge-based approach to word sense disambiguation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(7):1075–1086.

P. Resnik. 1999. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *JAIR*, (11):95–130.

R. Snow, D. Jurafsky, and A. Y. Ng. 2006. Semantic taxonomy induction from heterogenous evidence. In *ACL*, pages 801–808.