WikiWalk: Random walks on Wikipedia for Semantic Relatedness

Eric Yeh, Daniel Ramage, Christopher D. Manning
Computer Science Department,
Stanford University
Stanford, CA, USA
{yeh1,dramage,manning}@cs.stanford.edu

Eneko Agirre, Aitor Soroa
Ixa Taldea
University of the Basque Country
Donostia, Basque Country
{e.agirre,a.soroa}@ehu.es

Abstract

Computing semantic relatedness of natural language texts is a key component of tasks such as information retrieval and summarization, and often depends on knowledge of a broad range of real-world concepts and relationships. We address this knowledge integration issue by computing semantic relatedness using personalized PageRank (random walks) on a graph derived from Wikipedia. This paper evaluates methods for building the graph, including link selection strategies, and two methods for representing input texts as distributions over the graph nodes: one based on a dictionary lookup, the other based on Explicit Semantic Analysis. We evaluate our techniques on standard word relatedness and text similarity datasets, finding that they capture similarity information complementary to existing Wikipedia-based relatedness measures, resulting in small improvements on a state-of-the-art measure.

1 Introduction

Many problems in NLP call for numerical measures of semantic relatedness, including document summarization, information retrieval, and textual entailment. Often, measuring the relatedness of words or text passages requires world knowledge about entities and concepts that are beyond the scope of any single word in the document. Consider, for instance, the following pair:

1. *Emancipation Proclamation*
2. *Gettysburg Address*

To correctly assess that these examples are related requires knowledge of the United States Civil War found neither in the examples themselves nor in traditional lexical resources such as WordNet (Fellbaum, 1998). Fortunately, a massive collaboratively constructed knowledge resource is available that has specific articles dedicated to both. Wikipedia is an online encyclopedia containing around one million articles on a wide variety of topics maintained by over one hundred thousand volunteer editors with quality comparable to that of traditional encyclopedias.

Recent work has shown that Wikipedia can be used as the basis of successful measures of semantic relatedness between words or text passages (Strube and Ponzetto, 2006; Gabrilovich and Markovitch, 2007; Milne and Witten, 2008). The most successful measure, Explicit Semantic Analysis (ESA) (Gabrilovich and Markovitch, 2007), treats each article as its own dimension in a vector space. Texts are compared by first projecting them into the space of Wikipedia articles and then comparing the resulting vectors.

In addition to article text, Wikipedia stores a great deal of information about the relationships between the articles in the form of hyperlinks, info boxes, and category pages. Despite a long history of research demonstrating the effectiveness of incorporating link information into relatedness measures based on the WordNet graph (Budanitsky and Hirst, 2006), previous work on Wikipedia has made limited use of this relationship information, using only category links (Bunescu and Pasca, 2006) or just the actual links in a page (Gabrilovich and Markovitch, 2007; Milne and Witten, 2008).

In this work, we combine previous approaches by converting Wikipedia into a graph, mapping input texts into the graph, and performing random walks based on Personalized PageRank (Haveliwala, 2002) to obtain stationary distributions that characterize each text. Semantic relatedness between two texts is computed by comparing their distributions. In contrast to previous work, we explore the use of all these link types when con-

structing the Wikipedia graph, the intuition being these links, or some combination of them, contain additional information that would allow a gain over methods that use only just the article text. We also discuss two methods for performing the initial mapping of input texts to the graph, using techniques from previous studies that utilized WordNet graphs and Wikipedia article text.

We find that performance is signficantly affected by the strategy used to initialize the graph walk, as well as the links selected when constructing the Wikipedia graph. Our best system combines an ESA-initialized vector with random walks, improving on state-of-the-art results over the (Lee et al., 2005) dataset. An analysis of the output demonstrates that, while the gains are small, the random walk adds complementary relatedness information not present in the page text.

2 Preliminaries

A wide range of different methods, from corpus-based distributional similarity methods, such as Latent Semantic Analysis (Landauer et al., 1998), to knowledge-based ones that employ structured sources such as WordNet,[1] have been developed to score semantic relatedness and similarity. We now review two leading techniques which we use as starting points for our approach: those that perform random walks over WordNet's graph structure, and those that utilize Wikipedia as an underlying data source.

2.1 Random Graph Walks for Semantic Relatedness

Some of the best performing WordNet-based algorithms for computing semantic relatedness are based on the popular Personalized PageRank algorithm (Hughes and Ramage, 2007; Agirre and Soroa, 2009). These approaches start by taking WordNet as a graph of concepts $G = (V, E)$ with a set of vertices V derived from WordNet synsets and a set of edges E representing relations between synsets. Both algorithms can be viewed as random walk processes that postulate the existence of a particle that randomly traverses the graph, but at any time may jump, or *teleport*, to a new vertex with a given *teleport probability*. In standard PageRank (Brin and Page, 1998), this target is chosen uniformly, whereas for Personalized PageRank it is chosen from a nonuniform distribution of nodes, specified by a *teleport vector*.

The final weight of node i represents the proportion of time the random particle spends visiting it after a sufficiently long time, and corresponds to that node's structural importance in the graph. Because the resulting vector is the stationary distribution of a Markov chain, it is unique for a particular walk formulation. As the teleport vector is nonuniform, the stationary distribution will be biased towards specific parts of the graph. In the case of (Hughes and Ramage, 2007) and (Agirre and Soroa, 2009), the teleport vector is used to reflect the input texts to be compared, by biasing the stationary distribution towards the neighborhood of each word's mapping.

The computation of relatedness for a word pair can be summarized in three steps: First, each input word is mapped with to its respective synsets in the graph, creating its teleport vector. In the case words with multiple synsets (senses), the synsets are weighted uniformly. Personalized PageRank is then executed to compute the stationary distribution for each word, using their respective teleport vectors. Finally, the stationary distributions for each word pair are scored with a measure of vector similarity, such as cosine similarity. The method to compute relatedness for text pairs is analogous, with the only difference being in the first step all words are considered, and thus the stationary distribution is biased towards all synsets of the words in the text.

2.2 Wikipedia as a Semantic Resource

Recent Wikipedia-based lexical semantic relatedness approaches have been found to outperform measures based on the WordNet graph. Two such methods stand out: Wikipedia Link-based Measure (WLM) (Milne and Witten, 2008), and Explicit Semantic Analysis (ESA) (Gabrilovich and Markovitch, 2007).

WLM uses the anchors found in the body of Wikipedia articles, treating them as links to other articles. Each article is represented by a list of its incoming and outgoing links. For word relatedness, the set of articles are first identified by matching the word to the text in the anchors, and the score is derived using several weighting strategies applied to the overlap score of the articles' links. WLM does not make further use of the link graph, nor does it attempt to differentiate the links.

[1] See (Budanitsky and Hirst, 2006) for a survey.

In contrast to WLM, Explicit Semantic Analysis (ESA) is a vector space comparison algorithm that does not use the link structure, relying solely on the Wikipedia article text. Unlike Latent Semantic Analysis (LSA), the underlying concept space is not computationally derived, but is instead based on Wikipedia articles. For a candidate text, each dimension in its ESA vector corresponds to a Wikipedia article, with the score being the similarity of the text with the article text, subject to TF-IDF weighting. The relatedness of two texts is computed as the cosine similarity of their ESA vectors.

Although ESA reports the best results to date on both the WordSim-353 dataset as well as the Lee sentence similarity dataset, it does not utilize the link structure, which motivated a combined approach as follows.

2.3 A Combined Approach

In this work, we base our random walk algorithms after the ones described in (Hughes and Ramage, 2007) and (Agirre et al., 2009), but use Wikipedia-based methods to construct the graph. As in previous studies, we obtain a relatedness score between a pair of texts by performing random walks over a graph to compute a stationary distribution for each text. For our evaluations, the score is simply the cosine similarity between the distributions. In the following sections, we describe how we built graphs from Wikipedia, and how input texts are initially mapped into these structures.

3 Building a Wikipedia Graph

In order to obtain the graph structure of Wikipedia, we simply treat the articles as vertices, and the links between articles as the edges. There are several sources of pre-processed Wikipedia dumps which could be used to extract the articles and links between articles, including DBpedia (Auer et al., 2008), which provides a relational database representation of Wikipedia, and Wikipedia-Miner [2], which produces similar information from Wikipedia dumps directly. In this work we used a combination of Wikipedia-Miner and custom processing scripts. The dump used in this work is from mid 2008.

As in (Milne and Witten, 2008), anchors in Wikipedia articles are used to define links between articles. Because of different distributional properties, we explicitly distinguish three types of links, in order to explore their impact on the graph walk.

Infobox links are anchors found in the *infobox* section of Wikipedia articles. Article infoboxes, when present, often enumerate defining attributes and characteristics for that article's topic.

Categorical links reference articles whose titles belong in the Wiki namespace *"Category,"* as well as those with titles beginning with *"List of."* These pages are often just lists of anchors to other articles, which may be useful for capturing categorical information that roughly contains a mixture of hyponymy and meronymy relations between articles.

Content links are those that are not already classified as *infobox* nor *categorical*, and are intended to represent the set of miscellaneous anchors found solely in the article body. These may include links already found in the categorical and infobox categories.

Links can be further factored out according to *generality*, a concept introduced in (Gabrilovich and Markovitch, 2009). We say that one article is more general than another when the number of inlinks is larger. Although only a rough heuristic, the intuition is that articles on general topics will receive many links, whereas specific articles will receive fewer. We will use $+k$ notation for links which point to more general articles, i.e., where the difference in generality between source s and target t is $\#inlink(t)/\#inlink(s) \geq k$. We will use $-k$ for links to less general articles, i.e., $\#inlink(s)/\#inlink(t) \geq k$. Finally we use $=k$ when the generality is in the same order of magnitude, i.e., when the link is neither $+k$ nor $-k$. The original notion of generality from (Gabrilovich and Markovitch, 2009) restricts consideration to only more general articles by one order of magnitude (+10), without reference to the link types introduced above.

Given the size of the Wikipedia graph, we explored further methods inspired by (Gabrilovich and Markovitch, 2009) to make the graph smaller. We discarded articles with fewer than 2,000 non-stop words and articles with fewer than 5 outgoing and incoming links. We will refer to the complete

[2] http://wikipedia-miner.sourceforge.net

graph as *full* and to this reduced graph as *reduced*.[3]

4 Initializing a Wikipedia Graph Walk

In order to apply Personalized PageRank to a given passage of text or word, we need to construct a custom teleport vector, representing the initial distribution of mass over the article nodes. In this section we introduce two such methods, one based on constructing a direct mapping from individual words to Wikipedia articles (which we call dictionary-based initialization), and the other based directly on the results of ESA. We will see each technique in turn.

4.1 Dictionary based initialization

Given a target word, we would like to define its teleport vector using the set of articles in Wikipedia to which the word refers. This is analogous to a dictionary, where an entry lists the set of meanings pertaining to the entry.

We explored several methods for building such a dictionary. The first method constructed the dictionary using the article title directly, while also including redirection pages and disambiguation pages for additional ways to refer to the article. In addition, we can use the anchor text to refer to articles, and we turned to Wikipedia-Miner to extract this information. Anchors are indeed a rich source of information, as they help to relate similar words to Wikipedia articles. For instance, links to page `Monk` are created by using textual anchors such as *lama*, *brothers*, *monastery*, etc. As a result, the dictionary entries for those words will have a link to the `Monk` page. This information turned out to be very valuable, so all experiments have been carried out using anchors.

An additional difficulty was that any of these methods yielded dictionaries where the entries could refer to tens, even hundreds of articles. In most of the cases we could see that relevant articles were followed by a long tail of loosely related articles. We tried two methods to prune the dictionary. The first, coarse, method was to eliminate all articles whose title contains a space. The motivation was that our lexical semantic relatedness datasets (cf. Section 5) do not contain multiword entries (e.g., *United States*). In the second method, we pruned articles from the dictionary which accounted for less than 1% or 10% of the occurrences of that anchor word, as suggested by (Milne and Witten, 2008).

[3] In order to keep category and infobox links, the 2,000 non-stop word filter was not applied to categories and lists of pages.

Graphs		
Graph	# Vertices	# Edges
Full	2,483,041	49,602,752
Reduced	1,002,411	30,939,288

Dictionaries		
Dictionary	# Entries	Avg. Articles
all	6,660,315	1.31
1%	6,660,306	1.12
1% noent	1,058,471	1.04

Table 1: Graph and dictionary sizes. Avg. Articles column details the average number of articles per entry.

In short, for this method of initialization, we explored the use of the following variants: *all*, all articles are introduced in the dictionary; *noent*, articles with space characters are omitted; *1%* (*10%*), anchors that account for less than 1% (10%) of the total number of anchors for that entry are omitted. We did not use stemming. If a target word has no matching Wikipedia article in the dictionary, then it is ignored.

Table 1 shows the numbers for some graph and dictionary versions. Although the average number of articles per entry in the dictionary might seem low, it is actually quite high for the words in the datasets: for MC it's 5.92, and for wordsim353 it's 42.14. If we keep the articles accounting for 10% of all occurrences, the numbers drops drastically to 1.85 and 1.64 respectively.

As we will see in the results section, smaller graphs and dictionaries are able to attain higher results, but at the cost of losing information for some words. That is, we observed that some factored, smaller graphs contained less noise, but that meant that some articles and words are isolated in the graph, and therefore we are not able to compute relatedness for them. As a solution, we devised an alternative way to initialize the random walk. Instead of initializing it according to the articles in the dictionary, we initialized it with the vector weights returned by ESA, as explained in the next section.

4.2 Initialization with ESA

In addition to the dictionary based approach, we also explored the use of ESA to construct the teleport vector. In contrast to dictionary initialization, ESA uses the text of the article body instead of anchor text or the article titles. Because ESA maps query text to a weighted vector of Wikipedia articles, it can be naturally adapted as a teleport vector for a random walk with a simple L_1 normalization. We used Apache Lucene[4] to implement both ESA's repository of Wikipedia articles, and to return vectors for queries. Each article is indexed as its own document, with page text preprocessed to strip out Wiki markup.

Although we followed the steps outlined in (Gabrilovich and Markovitch, 2007), we had to add an extension to the algorithm: for a return vector from ESA, we order the articles by score, and retain only the scores for the top-n articles, setting the scores of the remaining articles to 0. Without this modification, our performance results were will below the reported numbers, but with a cutoff at 625 (determined by a basic grid search), we obtained a correlation of 0.76 on the Lee sentence similarity dataset, over the previously published score of 0.72.

4.3 Teleport Probability

For this work, we used a value of 0.15 as the probability of returning to the teleport distribution at any given step. The walk terminates when the vector converges with an L_1 error of 0.0001 (circa 30 iterations). Some preliminary experiments on a related Word Sense Disambiguation task indicated that in this context, our algorithm is quite robust to these values, and we did not optimize them. However, we will discuss using different return parameters in Section 6.1.

5 Experiments

In this section, we compare the two methods of initialization as well as several types of edges. For a set of pairs, system performance is evaluated by how well the generated scores correlate with the gold scores. Gold scores for each pair are the average of human judgments for that pair. In order to compare against previous results obtained on the datasets, we use the Spearman correlation coefficient on the Miller Charles (MC) and WordSim-353 word-pair datasets, and the Pearson correlation coefficient on the (Lee et al., 2005) document-pair dataset.

[4] `http://lucene.apache.org`

Dictionary	Graph	MC
all	full	0.369
1%	full	0.610
1%, noent	full	0.565 (0.824)
1%	reduced	0.563
1%	reduced +2	0.530
1%	reduced +4	0.601
1%	reduced +8	0.512
1%	reduced +10	0.491 (0.522)
10%	full	0.604 (0.750)
10%	reduced	0.605 (0.751)
10%	reduced +2	0.491 (0.540)
10%	reduced +4	0.476 (0.519)
10%	reduced +8	0.474 (0.506)
10%	reduced +10	0.430 (0.484)
WordNet		0.90 / 0.89
WLM		0.70
ESA		0.72

Table 2: Spearman correlation on the MC dataset with dictionary-based initialization. Refer to Section 3 for explanation of dictionary and graph building methods. Between parenthesis, results excluding pairs which had a word with an empty dictionary entry.

5.1 Dictionary-based Initialization

Given the smaller size of the MC dataset, we explored the effect of the different variants to build the graph and dictionary on this dataset. Some selected results are shown in Table 2, alongside those of related work, where we used WordNet for (Hughes and Ramage, 2007) and (Agirre et al., 2009) (separated by "/" in the results), WLM for (Milne and Witten, 2008) and ESA for (Gabrilovich and Markovitch, 2007).

We can observe that using the full graph and dictionaries yields very low results. Reducing the dictionary (removing articles with less than 1% or 10% of the total occurrences) produces higher results, but reducing the graph does not provide any improvement. On a closer look, we realized that pruning the dictionary to 10% or removing multiwords (*noent*) caused some words to not get any link to articles (e.g., *magician*). If we evaluate only over pairs where both words get a Personalized PageRank vector, the results raise up to 0.751 and 0.824, respectively, placing our method close

Dictionary	Graph	WordSim-353
1%	full	0.449
1%, noent	full	0.440 (0.634)
1%	reduced	0.485
WordNet		0.55 / 0.66
WLM		0.69
ESA		0.75
WikiRelate		0.50

Table 3: Spearman correlation on the WordSim-353 dataset with dictionary-based initialization. Refer to Section 3 for explanation of dictionary and graph building methods. Between parenthesis, results excluding pairs which had a word with an empty dictionary entry.

Dictionary	Graph	(Lee et al., 2005)
1%, noent	Full	0.308
1%	Reduced +4	0.269
ESA		0.72

Table 4: Pearson correlation on (Lee et al., 2005) with dictionary-based initialization. Refer to Section 3 for explanation of dictionary and graph building methods.

to the best results on the MC dataset. This came at the cost of not being able to judge the relatedness of 3 and 5 pairs, respectively. We think that removing multiwords (noent) is probably too drastic, but the positive effect is congruent with (Milne and Witten, 2008), who suggested that the coverage of certain words in Wikipedia is not adequate.

The results in Table 3 show the Spearman correlation for some selected runs over the WordSim-353 dataset. Again we see that a restrictive dictionary allows for better results on the pairs which do get a dictionary entry, up to 0.63. WikiRelate refers to the results in (Strube and Ponzetto, 2006).

We only tested a few combinations over (Lee et al., 2005), with results given in Table 4. These are well below state-of-the-art, and show that initializing the random walk with all words in the document does not characterize the documents well, resulting in low correlation.

5.2 ESA-based initialization

While the results using a dictionary based approach were encouraging, they did not come close to the state-of-the-art results achieved by ESA. Here, we explore combining ESA and random walks, by using ESA to initialize the teleport vector. Following section 4.2, we used a top-n cutoff of 625.

Method	Text Sim
ESA@625	0.766
ESA@625+Walk All	0.556
ESA@625+Walk Categories	0.410
ESA@625+Walk Content	0.536
ESA@625+Walk Infobox	0.710

Table 5: Pearson correlation on the (Lee et al., 2005) dataset when walking on various types of links. Note that walking tends to hurt performance overall, with Infobox links by far the least harmful.

Table 5 displays the results of our ESA implementation followed by a walk from that ESA distribution. Walking on any link type actually depresses performance below the baseline ESA value, although the Infobox links seem the least harmful.

However, as mentioned in Section 3, links between articles represent many different types of relationships beyond the few well-defined links present in lexical resources like WordNet. This also extends to where the link is found, and the article it is pointing to. As such, not all links are created equal, and we expect that some types of links at different levels of generality will perform better or worse than others. Table 6 presents a sample grid search across the category links choosing more general, less general, or similar generality at several factors of k, showing that there is a consistent pattern across multiple link types. Note that the best value indeed improves upon the score of the ESA distribution, albeit modestly.

We performed a similar analysis across all link types and found that the best link types were Category links at +6 and Infobox links at =2. Intuitively, these link types make sense: for semantic relatedness, it seem reasonable to expect more general pages within the same category to help. And for Infobox links, much rarer and much more common pages can both introduce their own kind of noise. While the improvement from each type of edge walk is small, they are additive—the best results on the sentence similarity dataset was from walking across both link types. Our final Pearson correlation coefficient of .772 is to our knowledge the highest number reported in the literature, al-

Generality of *Category* links			
	+k	-k	=k
$k = 2$	0.760	0.685	0.462
$k = 4$	0.766	0.699	0.356
$k = 6$	**0.771**	0.729	0.334
$k = 8$	0.768	0.729	0.352
$k = 10$	0.768	0.720	0.352

Table 6: Pearson correlation on the (Lee et al., 2005) with random walks over only a subset of the edges in the Category link information (scores .410 when taking all edges). Note that factoring the graph by link generality can be very helpful to the walk.

Method	Text Sim
ESA@625	0.766
ESA@625+Walk Cat@+6	0.770
ESA@625+Walk Cat@+6 Inf@=2	0.772
Bag of words (Lee et al., 2005)	0.5
LDA (Lee et al., 2005)	0.60
ESA*	0.72

Table 7: Pearson correlation on the (Lee et al., 2005) dataset for our best sytems compared to previously reported numbers. ESA* is the score for raw ESA as reported number in (Gabrilovich and Markovitch, 2007).

beit only a small improvement over our ESA@625 score.

Despite the results obtained for text similarity, the best settings found for the Lee dataset did not translate to consistent improvements over the ESA baseline for Spearman rank correlation on the lexical similarity datasets. While our scores on the MC dataset of 30 word pairs did improve with the walk in roughly the same way as in Lee, no such improvements were found on the larger WordSim-353 data. On WordSim-353, our implementation of ESA scored 0.709 (versus Gabrilovich's reported ESA score of 0.75), and our walk on Cat@+6 showing no gain or loss. In contrast to the text similarity dataset, Infobox links were no longer helpful, bringing the correlation down to .699. We believe this is because Infobox links helped the most with entities, which are very rare in the WordSim-353 data, but are more common in the Lee dataset.

6 Discussion

Our results suggest that even with a simple dictionary-based approach, the graph of Wikipedia links can act as an effective resource for computing semantic relatedness. However, the dictionary approach alone was unable to reach the results of state-of-the-art models using Wikipedia (Gabrilovich and Markovitch, 2007; Milne and Witten, 2008) or using the same technique on WordNet (Hughes and Ramage, 2007; Agirre et al., 2009). Thus, it seems that the text of Wikipedia provides a stronger signal than the link structure. However, a pruned dictionary can improve the results of the dictionary based initialization, which indicates that some links are informative for semantic relatedness while others are not. The careful pruning, disambiguation and weighting functions presented in (Milne and Witten, 2008) are directions for future work.

The use of WordNet as a graph provided excellent results (Hughes and Ramage, 2007), close to those of ESA. In contrast with our dictionary-based initialization on Wikipedia, no pruning of dictionary or graph seem necessary to obtain high results with WordNet. One straightforward explanation is that Wikipedia is a noisy source of link information. In fact, both ESA and (Milne and Witten, 2008) use ad-hoc pruning strategies in order to obtain good results.

6.1 ESA and Walk Comparison

By using ESA to generate the teleport distribution, we were able to introduce small gains using the random walk. Because these gains were small, it is plausible that the walk introduces only modest changes from the initial ESA teleport distributions. To evaluate this, we examined the differences between the vector returned by ESA and distribution over the equivalent nodes in the graph after performing a random walk starting with that ESA vector.

For this analysis, we took all of the text entries used in this study, and generated two distributions over the Wikipedia graph, one using ESA@625, the other the result of performing a random walk starting at ESA@625. We generated a list of the concept nodes for both distributions, sorted in decreasing order by their associated scores. Starting from the beginning of both lists, we then counted the number of matched nodes until they disagreed on ordering, giving a simple view of

	Walk Type	Avg	Std	Max
MC	Cat@+6	12.1	7.73	35
	Cat@+6 Inf@=2	5.39	5.81	20
WordSim	Cat@+6	12.0	10.6	70
	Cat@+6 Inf@=2	5.74	7.78	54
Lee	Cat@+6	28.3	89.7	625
	Cat@+6 Inf@=2	4.24	14.8	103

Table 8: Statistics for first concept match length, by run and walk type.

Prob	Corr (Pruned)	Corr (All)
0.01	0.772	0.246
0.10	0.773	0.500
0.15	0.772	0.556
0.30	0.771	0.682
0.45	0.769	0.737
0.60	0.767	0.758
0.90	0.766	0.766
0.99	0.766	0.766

Table 9: Return probability vs. correlation, on textual similarity data (Lee et al., 2005).

how the walk perturbed the strongest factors in the graphs. We performed this for both the best performing walk models (ESA@625+Walk Cat@+6 and ESA@625+Walk Cat@+6 Inf@=2) against ESA@625. Results are given in Table 8.

As expected, adding edges to the random walk increases the amount of change from the graph, as initialized by ESA. A cursory examination of the distributions also revealed a number of outliers with extremely high match lengths: these were likely due to the fact that the selected edge types were already extremely specialized. Thus for a number of concept nodes, it is likely they did not have any outbound edges at all.

Having established that the random walk does indeed have an impact on the ESA vectors, the next question is if changes via graph walk are consistently helpful. To answer this, we compared the performance of the walk on the (Lee et al., 2005) dataset for probabilities at selected values, using the best link pruned Wikipedia graph (ESA@625+Walk Cat@+6 Inf@=2), and using all of the available edges in the graph for comparison. Here, a lower probability means the distribution spreads out further into the graph, compared to higher values, where the distribution varies only slightly from the ESA vector. Results are given in Table 9. Performance for the pruned graph improves as the return probability decreases, with larger changes introduced by the graph walk resulting in better scores, whereas using all available links decreases performance. This reinforces the notion that Wikipedia links are indeed noisy, but that within a selected edge subset, making use of all information via the random walk indeed results in gains.

7 Conclusion

This paper has demonstrated that performing random walks with Personalized PageRank over the Wikipedia graph is a feasible and potentially fruitful means of computing semantic relatedness for words and texts. We have explored two methods of initializing the teleport vector: a dictionary-based method and a method based on ESA, the current state-of-the-art technique. Our results show the importance of pruning the dictionary, and for Wikipedia link structure, the importance of both categorizing by anchor type and comparative generality. We report small improvements over the state-of-the-art on (Lee et al., 2005) using ESA as a teleport vector and a limited set of links from Wikipedia category pages and infoboxes.

In future work, we plan to explore new ways to construct nodes, edges, and dictionary entries when constructing the Wikipedia graph and dictionary. We believe that finer grained methods of graph construction promise to improve the value of the Wikipedia link structure. We also plan to further investigate the differences between WordNet and Wikipedia and how these may be combined, from the perspective of graph and random walk techniques. A public distribution of software used for these experiments will also be made available.[5]

Acknowledgements

The authors would like to thank Michael D. Lee and Brandon Pincombe for access to their textual similarity dataset, and the reviewers for their helpful comments. Eneko Agirre performed part of the work while visiting Stanford, thanks to a grant from the Science Ministry of Spain.

[5]Please see `http://nlp.stanford.edu/software` and `http://ixa2.si.ehu.es/ukb`

References

E. Agirre and A. Soroa. 2009. Personalizing pagerank for word sense disambiguation. In *Proceedings of 14th Conference of the European Chapter of the Association for Computational Linguistics*, Athens, Greece.

E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Pasça, and A. Soroa. 2009. A study on similarity and relatedness using distributional and WordNet-based approaches. In *Proceedings of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies*, Boulder, USA.

S Auer, C Bizer, G Kobilarov, J Lehmann, R Cyganiak, and Z Ives. 2008. Dbpedia: A nucleus for a web of open data. In *Proceedings of 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference (ISWC+ASWC 2007)*, pages 722–735.

S. Brin and L. Page. 1998. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1-7).

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

R. C. Bunescu and M. Pasca. 2006. Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of 11th Conference of the European Chapter of the Association for Computational Linguistics*.

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

E. Gabrilovich and S. Markovitch. 2007. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07)*.

E. Gabrilovich and S. Markovitch. 2009. Wikipedia-based semantic interpretation. *Journal of Artificial Intelligence Research*, 34:443–498.

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 *Proceedings of EMNLP-CoNLL*, pages 581–589.

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

M. D. Lee, B. Pincombe, and M. Welsh. 2005. An empirical evaluation of models of text document similarity. In *Proceedings of the 27th Annual Conference of the Cognitive Science Society*, pages 1254–1259, Mahwah, NJ. Erlbaum.

D. Milne and I.H. Witten. 2008. An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. In *Proceedings of the first AAAI Workshop on Wikipedia and Artifical Intellegence (WIKIAI'08)*, Chicago, I.L.

M. Strube and S.P. Ponzetto. 2006. Wikirelate! computing semantic relatedness using Wikipedia. In *Proceedings of the 21st National Conference on Artificial Intelligence*, pages 1419–1424.