A Semantic Similarity Approach to Paraphrase Detection

Samuel Fernando and Mark Stevenson
Department of Computer Science
University of Sheffield
Sheffield, S1 4DP, UK
{s.fernando, m.stevenson}@shef.ac.uk

Abstract

This paper presents a novel approach to the problem of paraphrase identification. Although paraphrases often make use of synonymous or near synonymous terms, many previous approaches have either ignored or made limited use of information about similarities between word meanings. We present an algorithm for paraphrase identification which makes extensive use of word similarity information derived from WordNet (Fellbaum, 1998). The approach is evaluated using the Microsoft Research Paraphrase Corpus (Dolan et al., 2004), a standard resource for this task, and found to outperform previously published methods.

1 Introduction

Paraphrase is defined as the restatement (or reuse) of text giving the meaning in another form. Paraphrase identification is important for Information Extraction, Machine Translation, Information Retrieval and automatic identification of copyright infringement (Clough et al., 2002).

Many previous approaches to paraphrase detection (Clough et al., 2002; Qiu et al., 2006; Zhang and Patrick, 2005) have relied on purely lexical based matching techniques: the similarity between two candidate texts is computed as a function of the number of matching sequences of tokens between the texts. These methods will fail to identify the similarity between sentences which use different, but synonymous, words to convey the same meaning. Consider the following examples, paraphrased from recent on-line news sources:

S1. “The Iraqi Foreign Minister warned of disastrous consequences if Turkey launched an invasion of Iraq”

S2. “Iraq has warned that a Turkish incursion would have disastrous results”

A simple lexical matching comparison of the two sentences would fail to take into account words with very similar meanings such as ‘consequences’ and ‘results’ or ‘invasion’ and ‘incursion’.

Methods for determining the similarity of a pair of words are readily available, including several techniques based on WordNet (e.g. (Leacock and Chodorow, 1998; Wu and Palmer, 1994; Resnik, 1995)). Mihalcea et al. (2006) described a method for paraphrase detection which made use of these methods.

This paper presents a novel method, the matrix similarity approach, which was originally proposed for Information Extraction (Stevenson and Greenwood, 2005). This approach also uses semantic similarity metrics to find the similarity of two text segments, but a key difference is that all word-to-word similarities are taken into account, not just the maximal similarities between the sentences as in the method proposed in (Mihalcea et al., 2006). The hypothesis here is that taking into account all similarities in this way improves performance.

The outline for the rest of the paper is as follows. Section 2 describes some of the previous approaches to paraphrase identification and their limitations. The approach proposed here is described in Section 3. Section 4 gives a brief description of the Microsoft Research Paraphrase Corpus which is used for evaluation. Section 5 presents the results of this evaluation which is also compared against the performance of previous approaches. Conclusions and suggestions for future work are presented in Section 6.

2 Previous Approaches

The approach developed in (Qiu et al., 2006) uses a two-phase process. The first phase identifies the common *information nuggets* or key semantic content units in each sentence. These are then paired off. If any extraneous information nuggets remain then the significance of these are judged. If the sentences contain no unpaired nuggets or all unpaired nuggets are insignificant then a positive classification is given. The nuggets are predicate argument tuples which are compared using a simple lexical matching technique.

The main idea in the approach proposed by (Zhang and Patrick, 2005) is to create canonicalized forms of the sentences so that texts with similar meaning are more likely to be transformed into the same surface texts than those with different meaning. Once the text is transformed in this way simple lexical matching techniques are used to compare the transformed text.

In contrast, a recent approach that goes beyond simple lexical matching is described in (Mihalcea et al., 2006). Word-to-word similarity measures and a word specificity measure are used to estimate the semantic similarity of the sentence pairs.

The following scoring function was used:

$$sim(T_1, T_2) = \frac{1}{2}\left(\frac{\sum_{w \in \{T_1\}}(maxSim(w,T_2) * idf(w))}{\sum_{w \in \{T_1\}} idf(w)} + \frac{\sum_{w \in \{T_2\}}(maxSim(w,T_1) * idf(w))}{\sum_{w \in \{T_2\}} idf(w)}\right) \quad (1)$$

where $maxSim(w, T)$ is the maximum similarity score found between word w and words in T according to a word-to-word similarity measure, and $idf(w)$ is the inverse document frequency of the word. A threshold of 0.5 was used for classification: a score above the threshold was classified as paraphrase otherwise as not paraphrase.

Mihalcea et al. (2006) experimented with a number of measures for computing similarities between words: knowledge-based metrics which made use of WordNet and corpus-based measures.

3 Semantic Matrix Approach

While the approach proposed by (Mihalcea et al., 2006) outperforms simpler lexical matching techniques, it is still limited by the fact that only the most similar word for the other sentence is taken into account. There are cases where this can hinder the identification of paraphrases, such as the following example:

S3. “Iraq has warned that a full-scale Turkish incursion attacking Kurdish rebel bases in northern Iraq would have disastrous results”

S4. “The Iraqi Foreign Minister warned of disastrous consequences if Turkey launched a major invasion of Iraq to strike at Kurdish rebels ”

The approach proposed by (Mihalcea et al., 2006) may find for the verb ‘attack’ in sentence S3 the most similar word to be the verb ‘strike’. However it would then not take into account that ‘invasion’ also has a high similarity to the word attack.

(Stevenson and Greenwood, 2005) describe an Information Extraction system which attempts to identify relevant information by searching for paraphrases of sentences known to contain relevant information. Their approach uses a *similarity matrix* to calculate the similarity between a pair of vectors representing Information Extraction patterns. This approach can be adapted to the problem of paraphrase identification in a straightforward way and has the advantage that all similarity scores between all word pairs between the sentences are taken into account.

This approach operates by representing each sentence as a binary vector (with elements equal to 1 if a word is present and 0 otherwise), $\vec{a}$ and $\vec{b}$. The similarity between these sentences can be computed using the following formula:

$$sim(\vec{a}, \vec{b}) = \frac{\vec{a} W \vec{b}^T}{|\vec{a}||\vec{b}|} \quad (2)$$

where W is a semantic similarity matrix containing information about the similarity of word pairs.

Formally, each element w_{ij} in W represents the similarity of words p_i and p_j according to some lexical similarity measure. If this measure is symmetric, i.e. $w_{ij} = w_{ji}$, then the matrix is also symmetric. Diagonal elements represent self similarity and should consequently have the greatest values. The actual values denoting the similarity between pattern elements are acquired using existing lexical similarity metrics (see Section 3.1). Since experiments with document specificity weightings (such as *tf-idf*) had shown that using these factors actually reduced performance no such weighting factor was used here.

The measure in Equation 2 is similar to the cosine metric, commonly used to determine the similarity of documents in the vector space model approach to Information Retrieval.[1]

Figure 1 shows a sample similarity matrix for the sentences (A) "The dog sat on the mat" and (B) "The mutt sat on the rug". Using the metric shown in formula 2 the similarity between these two sentences is 0.9. If the similarity matrix was not used (and the sentences compared by considering only the proportion of content words they have in common) their similarity would be just 0.33. (The self-similarity of sentences (A) and (B) is 1.)

	dog	mat	mutt	rug	sat
dog	1	0	0.8	0	0
mat	0	1	0	0.9	0
mutt	0.8	0	1	0	0
rug	0	0.9	0	1	0
sat	0	0	0	0	1

Figure 1: Sample similarity matrix showing similarity scores for content words from two sentences.

3.1 Computing Lexical Similarity

It is important to choose appropriate values for the elements of W. We made use of the work that has been carried out on computing lexical similarity (Banerjee and Pedersen, 2003; Wu and Palmer, 1994; Resnik, 1995).[2] This research has concentrated on developing methods for determining the similarity of pairs of lexical items, often using the WordNet hierarchy (Fellbaum, 1998).

We experimented with six WordNet similarity metrics to populate the similarity matrix. Five of these are similarity metrics which use only information about the *is-a* hierarchy to determine the similarity of the concepts. The remaining metric (*lesk*) is strictly speaking a *relatedness* metric since it uses additional information apart from hypernymy to measure the similarity of the two concepts.

The *lesk* metric (Banerjee and Pedersen, 2003) measures the overlap between the glosses of the two concepts and also concepts directly related via relations such as hypernyms and meronyms.

The *lch* metric (Leacock and Chodorow, 1998) determines the similarity of two nodes by finding the path length between them in the *is-a* hierarchy. The similarity is computed as:

$$sim_{lch} = -log\frac{N_p}{2D} \qquad (3)$$

where N_p is the distance between the nodes and D is the maximum depth in the *is-a* taxonomy.

The *wup* metric (Wu and Palmer, 1994) computes the similarity of the nodes as a function of the path length from the least common subsumer (LCS) of the nodes. Given two concept nodes C_1 and C_2 in a *is-a* hierarchy the *LCS* is defined as the most specific node which both share as an ancestor. For example if C_1 was 'car' and C_2 was 'boat' then the LCS would be 'vehicle'. The similarity between nodes C_1 and C_2 is:

$$sim_{wup} = \frac{2 \times depth(LCS(C_1, C_2))}{depth(C_1) + depth(C_2)} \qquad (4)$$

where $depth(C)$ is the depth of concept C in the WordNet hierarchy.

The *res* metric (Resnik, 1995) uses the information content of the LCS of the two concepts:

$$sim_{res} = IC(LCS(C_1, C_2)) \qquad (5)$$

[1] The cosine metric for a pair of vectors is given by the calculation $\frac{a.b}{|a||b|}$. Substituting the matrix multiplication in the numerator of Equation 2 for the dot product of vectors $\vec{a}$ and $\vec{b}$ would give the cosine metric. Note that taking the dot product of a pair of vectors is equivalent to multiplying by the identity matrix, i.e. $\vec{a}.\vec{b} = \vec{a}I\vec{b}^{T}$. Under our interpretation of the similarity matrix, W, this equates to saying that lexical items are identical to themselves but not similar to anything else.

[2] We made use of the implementations of these measures which are available in the `WordNet::Similarity` package (Pedersen et al., 2004)

The information content of a node is an estimate of how informative the concept is, with frequently occurring concepts deemed to have low information content and rarely occurring concepts deemed to have high information content. Formally the information content of a concept c is defined as:

$$IC(c) = -logP(c) \tag{6}$$

where $P(c)$ is the probability of finding c in a large corpus.

The *lin* metric (Lin, 1998) builds on the *resnik* measure by normalising using the information content of the two nodes themselves:

$$sim_{lin} = \frac{2 * IC(LCS(C_1, C_2))}{IC(C_1) + IC(C_2)} \tag{7}$$

The *jcn* metric (Jiang and Conrath, 1997) uses the same information combined in a different way: $sim_{jcn} =$

$$\frac{1}{IC(C_1) + IC(C_2) + 2 \times IC(LCS(C_1, C_2))} \tag{8}$$

The *lin* and *wup* measures return a score between 0 and 1, where 1 indicates the highest possible similarity score for a pair of words and 0 that they are completely dissimilar. The remaining measures return scores in a variety of ranges which were normalised by dividing by the maximum possible score.

Since all senses of words are being compared sometimes spurious similarities may result, with an overinflated score using the WordNet similarity metric. To help avoid this we set a threshold of 0.8 on each of the lexical similarity scores, if a similarity metric is below this value for particular word pair then a value of 0 is entered in the appropriate position in the similarity matrix.

4 The Microsoft Research Paraphrase Corpus

The MSRPC[3] (Dolan et al., 2004) comprises 5801 candidate paraphrase sentence pairs which have been obtained from Web news sources. The sentence pairs have been marked by human judges with a binary classification determining if they are in fact paraphrases or not. The data is unbalanced in that 67% of the pairs are positive examples and only 33% is negative. The data have been arbitrarily split into a training set containing 4076 examples and a test set containing 1725 examples. This train/test partition has been observed by all the approaches evaluated here.

4.1 Threshold for paraphrases

A strict definition of a paraphrase would insist on the two candidate texts having identical meanings. However the creators of the MSRPC found this strict definition would limit paraphrase pairs to be virtually identical string copies of each other. Such pairs provide interesting data for analysing minor syntactic and lexical alternations but the authors of the MSRPC wanted to capture more interesting and complex differences. This led to them relaxing the definition of a paraphrase from ‘full bidirectional entailment’ to ‘mostly bidirectional entailments’ (Dolan and Brockett, 2005). Additional weighting is given to the principal actors and events described. The key message of the guidelines for the annotators of the corpus state that in order to constitute a paraphrase sentence pairs should describe the same event and contain the same important information about that event.

Thus the following example from the MSRPC is a clear positive example of a paraphrase since S5 and S6 are virtually semantically identical:

S5. “Amrozi accused his brother, whom he called ‘the witness’, of deliberately distorting his evidence.”

S6. “Referring to him as only ‘the witness’, Amrozi accused his brother of deliberately distorting his evidence. ”

A more difficult example to classify is the following:

S7. “The former wife of rapper Eminem has been electronically tagged after missing two court appearances.”

S8. “After missing two court appearances *in a cocaine possession case*, Eminem’s ex-wife has been placed under electronic house arrest.”

[3] Available from `http://research.microsoft.com/nlp/msr_paraphrase.htm`

Here the sentences seem to be describing the same event but S8 contains some key information (shown in italics) that is not present in S7, so should therefore be classified as a negative example (not a paraphrase).

The concept of what exactly is to be considered important information is difficult to pin down, which means even for humans this is an ambiguous and subjective task. However the inter-rater agreement amongst the human judges was an impressively high 84%, which can be considered as an upper bound for the accuracy that could be obtained using automatic methods.

5 Experiments

The approach described in Section 3 was evaluated against the Microsoft Research Paraphrase Corpus. We experimented with the six different lexical similarity metrics (described in Section 3.1) for populating the similarity matrix. Most of the similarity metrics are limited to comparing words with the same part-of-speech so the corpus was tagged using TreeTagger (Schmid, 1994). All WordNet word senses were considered when finding the similarity between words. For each similarity metric the training part of the MSRPC was used to find the classification threshold for the similarity score which maximized accuracy.

Following (Mihalcea et al., 2006), two baselines were computed: *random* simply makes a random decision (i.e. "paraphrase" or "not paraphrase") for each candidate paraphrase and *vector-based*, a more informed baseline inspired by Information Retrieval, in which each sentence is represented by a vector with tf-idf weighting and the cosine metric used to compare them. Standard metrics were evaluated for each of the novel methods on the MSRPC test corpus: accuracy, precision, recall and F-1 measure. These are defined as follows:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

where TP are true positives, TN are true negatives, FN are false negatives and FP are false positives.

$$precision = \frac{TP}{TP + FP} \tag{10}$$

$$recall = \frac{TP}{TP + FN} \tag{11}$$

$$F = \frac{2 \times precision \times recall}{precision + recall} \tag{12}$$

Results for the semantic similarity approaches are shown in Table 1 for each WordNet similarity metric. Results from previous approaches (described in Section 2) are also shown in the table: the semantic similarity method (Mihalcea et al., 2006), a dissimilarity significance classifier approach (Qiu et al., 2006) and a text canonicalization metric (Zhang and Patrick, 2005).

The matrix similarity approach outperforms both baselines for all six of the similarity measures used in these experiments. It can also be seen that, with the exception of the *wup* measure, all of the similarity metrics outperform the previously reported methods in terms of classifier accuracy.

Metric	Acc.	Prec.	Rec.	F
matrixJcn	**74.1**	**75.2**	91.3	**82.4**
matrixLch	73.9	74.8	91.6	82.3
matrixLesk	72.9	73.5	92.6	82.0
matrixLin	73.7	74.2	92.5	**82.4**
matrixRes	72.2	73.8	90.4	81.2
matrixWup	71.6	75.2	85.4	80.0
Mihalcea, 2006	70.3	69.6	**97.7**	81.3
Qiu, 2006	72.0	72.5	93.4	81.6
Zhang, 2005	71.9	74.3	88.2	80.7
random	51.3	68.3	50.0	57.8
vector-based	65.4	71.6	79.5	75.3

Table 1: Results of novel methods and previously published work.

It is worth noting that the results quoted in Table 1 for (Mihalcea et al., 2006) are the best scores recorded for that method which was achieved by using the average similarity score from the six similarity measures described in Section 3.1. (Mihalcea et al., 2006) also tested versions of their approach using each of these measures alone. The accuracy figures for the approach presented by (Mihalcea et al., 2006) using a single similarity measure were compared against the equivalent figure using the approach presented here. It was found that the matrix similarity approach was significantly better (two-tailed paired t-test, $p < 0.01$) demonstrating that this method makes more effective use of the information available from WordNet.

The superiority of the *jcn* measure is consistent with other evaluations of the WordNet similarity measures (Budanitsky and Hirst, 2006).

6 Conclusion and Future Work

This paper presented a novel approach to the problem of paraphrase identification. Our method makes use of WordNet-based lexical similarity measures applied differently from previous approaches. The system was evaluated on the Microsoft Research Paraphrase Corpus and found to outperform previously reported approaches.

Section 4 noted that the inter-rater agreement figures for the Microsoft Research Paraphrase Corpus was 84%, suggesting that there is room for further refinement of paraphrase identification techniques. Various refinements to the matrix similarity approach presented here could be explored. For example, in addition to using WordNet-based similarity measures, (Mihalcea et al., 2006) also experiment with two corpus-based measures of lexical similarity: pointwise mutual information (Manning and Schütze, 1999) and Latent Semantic Analysis (Dumais, 2004). Also, a common criticism of WordNet is that the sense distinctions it contains are too fine-grained. The WordNet-based lexical similarity measures we use do not attempt to find the correct sense for words in candidate paraphrases but choose the highest similarity scores for a word pair. It is possible that Word Sense Disambiguation could be used to identify the correct senses, thereby leading to more accurate estimates for lexical similarities, although this may not be practical given the current state-of-the-art performance for Word Sense Disambiguation systems (Agirre et al., 2007).

The matrix similarity approach can be related to kernel methods which have been applied successfully for document classification (Basili et al., 2005) and word sense disambiguation (Gliozzo et al., 2005). A possible line of future work would be to experiment with a kernelised version of the matrix similarity approach.

The work here has focused on the use of lexical similarity for paraphrase detection, essentially using a bag-of-words model. Others, such as (Wang and Neumann, 2007), have used syntactic analysis for paraphrase detection and another interesting line of future research would be to explore whether it could be combined with the approach presented here.

References

Eneko Agirre, Lluis Marquez, and Richard Wicentowski, editors. 2007. *Proceedings of the 4th International Workshop on Semantic Evaluations*, Prague, Czech Republic.

Satanjeev Banerjee and Ted Pedersen. 2003. Extended gloss overlaps as a measure of semantic relatedness. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, pages 805–810.

Roberto Basili, Marco Cammisa, and Alessandro Moschitti. 2005. Effective use of WordNet semantics via kernel-based learning. In *Proceedings of the 9th Conference on Computational Natural Language Learning, Ann Arbor (MI), USA*.

Alexander Budanitsky and Graeme Hirst. 2006. Evaluating WordNet-based Measures of Lexical Semantic Relatedness. *Computational Linguistics*, 32(1):13–47.

Paul Clough, Robert Gaizauskas, Scott Piao, and Yorick Wilks. 2002. METER: MEasuring TExt Reuse. In *Proceedings of the 40th Anniversary Meeting for the Association for Computational Linguistics (ACL-02)*, pages 152–159, Pennsylvania, PA.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *The 3rd International Workshop on Paraphrasing (IWP2005)*.

Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised Construction of Large Paraphrase Corpora: Exploiting Massively Parallel News Sources. In *COLING '04: Proceedings of the 20th international conference on Computational Linguistics*, page 350, Morristown, NJ, USA. Association for Computational Linguistics.

Susan Dumais. 2004. Latent semantic analysis. *Annual Review of Information Science and Technology (ARIST)*, 38(4):189–230.

Christine Fellbaum. 1998. *WordNet – An Electronic Lexical Database*. MIT Press.

Alfio Gliozzo, Claudio Giuliano, and Carlo Strapparava. 2005. Domain Kernels for Word Sense Disambiguation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*.

Jay J. Jiang and David W. Conrath. 1997. Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. *Proceedings of the International Conference on Research in Computational Linguistics*.

Claudia Leacock and Martin Chodorow. 1998. Combining local context and WordNet similarity for word sense identification. In Christine Fellbaum, editor, *WordNet – An Electronic Lexical Database*. MIT Press.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *ICML '98: Proceedings of the Fifteenth International Conference on Machine Learning*, pages 296–304, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Christopher D. Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts.

Rada Mihalcea, Courtney Corley, and Carlo Strapparava. 2006. Corpus-based and Knowledge-based Measures of Text Semantic Similarity. In *Proceedings of the American Association for Artificial Intelligence (AAAI 2006)*, Boston, July.

Ted Pedersen, Siddarth Patwardhan, and Jason Michelizzi. 2004. WordNet::Similarity - Measuring the Relatedness of Concepts. In *Proceedings of the Nineteenth National Conference on Artificial Intelligence (AAAI-04), 2004*, pages 1024–1025, San Jose, CA.

Long Qiu, Min-Yen Kan, and Tat-Seng Chua. 2006. Paraphrase recognition via dissimilarity significance classification. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 18–26, Sydney, Australia, July. Association for Computational Linguistics.

Philip Resnik. 1995. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In *International Joint Conference on AI*, pages 448–453.

Helmut Schmid. 1994. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *International Conference on New Methods in Language Processing*, Manchester, UK.

Mark Stevenson and Mark A. Greenwood. 2005. A Semantic Approach to IE Pattern Induction. In *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 379–386, Morristown, NJ, USA. Association for Computational Linguistics.

Rui Wang and Günter Neumann. 2007. Recognizing textual entailment using sentence similarity based on dependency tree skeletons. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 36–41, Prague, June. Association for Computational Linguistics.

Zhibiao Wu and Martha Palmer. 1994. Verb semantics and lexical selection. In *32nd. Annual Meeting of the Association for Computational Linguistics*, pages 133 –138, New Mexico State University, Las Cruces, New Mexico.

Yitao Zhang and Jon Patrick. 2005. Paraphrase identification by text canonicalization. In *Proceedings of the Australasian Language Technology Workshop 2005*, pages 160–166, Sydney, Australia, December.