The Distributional Similarity of Sub-Parses

Julie Weeds, David Weir and Bill Keller
Department of Informatics
University of Sussex
Brighton, BN1 9QH, UK
{juliewe, davidw, billk}@sussex.ac.uk

Abstract

This work explores computing distributional similarity between sub-parses, i.e., fragments of a parse tree, as an extension to general lexical distributional similarity techniques. In the same way that lexical distributional similarity is used to estimate lexical semantic similarity, we propose using distributional similarity between sub-parses to estimate the semantic similarity of phrases. Such a technique will allow us to identify paraphrases where the component words are not semantically similar. We demonstrate the potential of the method by applying it to a small number of examples and showing that the paraphrases are more similar than the non-paraphrases.

1 Introduction

An expression is said to *textually entail* another expression if the meaning of the second expression can be inferred from the meaning of the first. For example, the sentence "London is an English city," textually entails the sentence "London is in England." As discussed by Dagan et al. (2005) in their introduction to the first Recognising Textual Entailment Challenge, identifying textual entailment can be seen as a subtask of a variety of other natural language processing (NLP) tasks. For example, Question Answering (QA) can be cast as finding an answer which is entailed by the proposition in the question. Other identified tasks include summarization, paraphrasing, Information Extraction (IE), Information Retrieval (IR) and Machine Translation (MT).

The Natural Habitats (NatHab) project[1] (Weeds et al., 2004; Owen et al., 2005) provides an interesting setting in which to study paraphrase and textual entailment recognition as a tool for natural language understanding. The aim of the project is to enable non-technical users to configure their pervasive computing environments. They do this by stating *policies* in natural language which describe how they wish their environment to behave. For example, a user, who wishes to restrict the use of their colour printer to the printing of colour documents, might have as a policy, "Never print black-and-white documents on my colour printer." Similarly, a user, who wishes to be alerted by email when their mobile phone battery is low, might have as a policy, "If my mobile phone battery is low then send me an email." The natural language understanding task is to interpret the user's utterance with reference to a set of policy templates and an ontology of services (e.g. *print*) and concepts (e.g. *document*). The use of policy templates and an ontology restricts the number of possible meanings that a user can express. However, there is still considerable variability in the way these policies can be expressed. Simple variations on the theme of the second policy above include, "Send me an email whenever my mobile phone battery is low," and "If the charge on my mobile phone is low then email me." Our approach is to tackle the interpretation problem by identifying parts of expressions that are paraphrases of those expressions whose interpretation with respect to the ontology is more directly encoded. Here, we investigate extending distributional similarity methods from words to sub-parses.

The rest of this paper is organised as follows. In Section 2 we discuss the background to our work. We consider the limitations of an approach based on lexical similarity and syntactic templates, which motivates us to look directly at the similarity of larger units. In Section 3, we introduce our proposed approach, which is to measure the distributional similarity of sub-parses. In Section 4, we consider examples from the Pascal Textual Entailment Challenge

[1] http://www.informatics.susx.ac.uk/projects/nathab/

Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, pages 7–12,
Ann Arbor, June 2005. ©2005 Association for Computational Linguistics

Datasets[2] (Dagan et al., 2005) and demonstrate empirically how similarity can be found between corresponding phrases when parts of the phrases cannot be said to be similar. In Section 5, we present our conclusions and directions for further work.

2 Background

One well-studied approach to the identification of paraphrases is to employ a lexical similarity function. As noted by Barzilay and Elhadad (2003), even a lexical function that simply computes word overlap can accurately select paraphrases. The problem with such a function is not in the accuracy of the paraphrases selected, but in its low recall. One popular way of improving recall is to relax the requirement for words in each sentence to be identical in form, to being identical or similar in meaning. Methods to find the semantic similarity of two words can be broadly split into those which use lexical resources, e.g., WordNet (Fellbaum, 1998), and those which use a distributional similarity measure (see Weeds (2003) for a review of distributional similarity measures). Both Jijkoun and deRijke (2005) and Herrara et al. (2005) show how such a measure of lexical semantic similarity might be incorporated into a system for recognising textual entailment between sentences.

Previous work on the NatHab project (Weeds et al., 2004) used such an approach to extend lexical coverage. Each of the user's uttered words was mapped to a set of candidate words in a core lexicon[3], identified using a measure of distributional similarity. For example, the word *send* is used when talking about printing or about emailing, and a good measure of lexical similarity would identify both of these conceptual services as candidates. The best choice of candidate was then chosen by optimising the match between grammatical dependency relations and paths in the ontology over the entire sentence. For example, an indirect-object relation between the verb *send* and a printer can be mapped to the path in the ontology relating a print request to its target printer.

As well as lexical variation, our previous work (Weeds et al., 2004) allowed a certain amount of syntactic variation via its use of grammatical dependencies and policy templates. For example, the passive "paraphrase" of a sentence can be identified by comparing the sets of grammatical dependency relations produced by a shallow parser such as the RASP parser (Briscoe and Carroll, 1995). In other words, by looking at grammatical dependency relations, we can identify that "John is liked by Mary," is a paraphrase of "Mary likes John," and not of "John likes Mary." Further, where there is a limited number of styles of sentence, we can manually identify and list other templates for matches over the trees or sets of dependency relations. For example, "If C1 then C2" is the same as "C2 if C1".

However, the limitations of this approach, which combines lexical variation, grammatical dependency relations and template matching, become increasingly obvious as one tries to scale up. As noted by Herrera (2005), similarity at the word level is not required for similarity at the phrasal level. For example, in the context of our project, the phrases "if my mobile phone needs charging" and "if my mobile phone battery is low" have the same intended meaning but it is not possible to obtain the second by making substitutions for similar words in the first. It appears that "X needs charging" and "battery (of X) is low" have roughly similar meanings without their component words having similar meanings. Further, this does not appear to be due to either phrase being non-compositional. As noted by Pearce (2001), it is not possible to substitute similar words within non-compositional collocations. In this case, however, both phrases appear to be compositional. Words cannot be substituted between the two phrases because they are composed in different ways.

3 Proposal

Recently, there has been much interest in finding words which are distributionally similar e.g., Lin (1998), Lee (1999), Curran and Moens (2002), Weeds (2003) and Geffet and Dagan (2004). Two words are said to be distributionally similar if they appear in similar contexts. For example, the two words *apple* and *pear* are likely to be seen as the objects of the verbs *eat* and *peel*, and this adds to their distributional similarity. The Distributional Hypothesis (Harris, 1968) proposes a connection between distributional similarity and semantic similarity, which is the basis for a large body of work on automatic thesaurus construction using distributional similarity methods (Curran and Moens, 2002; Weeds, 2003; Geffet and Dagan, 2004).

Our proposal is that just as words have distributional similarity which can be used, with at least some success, to estimate semantic similarity, so do larger units of expression. We propose that the unit of interest is a sub-parse, i.e., a fragment (connected subgraph) of a parse tree, which can range in size from a single word to the parse for the entire sen-

[2] http://www.pascal-network.org/Challenges/RTE/

[3] The core lexicon lists a canonical word form for each concept in the ontology.

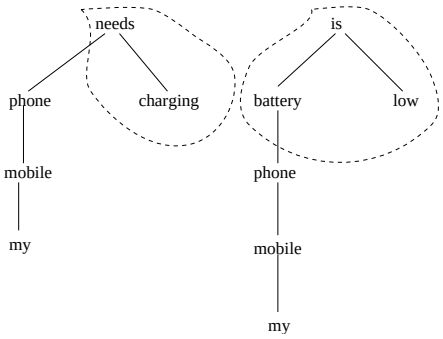

Figure 1: Parse trees for "my mobile phone needs charging" and "my mobile phone battery is low"

tence. Figure 1 shows the parses for the clauses, "my mobile phone needs charging," and "my mobile phone battery is low" and highlights the fragments ("needs charging" and "battery is low") for which we might be interested in finding similarity.

In our model, we define the features or contexts of a sub-parse to be the grammatical relations between any component of the sub-parse and any word outside of the sub-parse. In the example above, both sub-parses would have features based on their grammatical relation with the word *phone*. The level of granularity at which to consider grammatical relations remains a matter for investigation. For example, it might turn out to be better to distinguish between all types of dependent or, alternatively, it might be better to have a single class which covers all dependents. We also consider the parents of the sub-parse as features. In the example, "Send me an email if my mobile phone battery is low," this would be that the sub-parse modifies the verb *send* i.e., it has the feature, <mod-of, send>.

Having defined these models for the unit of interest, the sub-parse, and for the context of a sub-parse, we can build up co-occurrence vectors for sub-parses in the same way as for words. A co-occurrence vector is a conglomeration (with frequency counts) of all of the co-occurrences of the target unit found in a corpus. The similarity between two such vectors or descriptions can then be found using a standard distributional similarity measure (see Weeds (2003)).

The use of distributional evidence for larger units than words is not new. Szpektor et al. (2004) automatically identify *anchors* in web corpus data. Anchors are lexical elements that describe the context of a sentence and if words are found to occur with the same set of anchors, they are assumed to be paraphrases. For example, the anchor set {Mozart, 1756} is a known anchor set for verbs with the meaning "born in". However, this use of distributional evidence requires both anchors, or contexts, to occur simultaneously with the target word. This differs from the standard notion of distributional similarity which involves finding similarity between co-occurrence vectors, where there is no requirement for two features or contexts to occur simulultaneously.

Our work with distributional similarity is a generalisation of the approach taken by Lin and Pantel (2001). These authors apply the distributional similarity principle to *paths* in a parse tree. A path exists between two words if there are grammatical relations connecting them in a sentence. For example, in the sentence "John found a solution to the problem," there is a path between "found" and "solution" because solution is the direct object of found. Contexts of this path, in this sentence, are then the grammatical relations <ncsubj, John> and <iobj, problem> because these are grammatical relations associated with either end of the path. In their work on QA, Lin and Pantel restrict the grammatical relations considered to two "slots" at either end of the path where the word occupying the slot is a noun. Co-occurrence vectors for paths are then built up using evidence from multiple occurrences of the paths in corpus data, for which similarity can then be calculated using a standard metric (e.g., Lin (1998)). In our work, we extend the notion of distributional similarity from linear paths to trees. This allows us to compute distributional similarity for any part of an expression, of arbitrary length and complexity (although, in practice, we are still limited by data sparseness). Further, we do not make any restrictions as to the number or types of the grammatical relation contexts associated with a tree.

4 Empirical Evidence

Practically demonstrating our proposal requires a source of paraphrases. We first looked at the MSR paraphrase corpus (Dolan et al., 2004) since it contains a large number of sentences close enough in meaning to be considered paraphrases. However, inspection of the data revealed that the lexical overlap between the pairs of paraphrasing sentences in this corpus is very high. The average word overlap (i.e., the proportion of exactly identical word forms) calculated over the sentences paired by humans in the training set is 0.70, and the lowest overlap[4] for such sentences is 0.3. This high word overlap makes this a poor source of examples for us, since we wish to study similarity between phrases which do not share semantically similar words.

[4] A possible reason for this is that candidate sentences were first identified automatically.

Consequently, for our purposes, the Pascal Textual Entailment Recognition Challenge dataset is a more suitable source of paraphrase data. Here the average word overlap between textually entailing sentences is 0.39 and the lowest overlap is 0. This allows us to easily find pairs of sub-parses which do not share similar words. For example, in paraphrase pair id.19, we can see that "reduce the risk of diseases" entails "has health benefits". Similarly in pair id.20, "may keep your blood glucose from rising too fast" entails "improves blood sugar control," and in id.570, "charged in the death of" entails "accused of having killed."

In this last example there is semantic similarity between the words used. The word *charged* is semantically similar to *accused*. However, it is not possible to swap the two words in these contexts since we do not say "charged of having killed." Further, there is an obvious semantic connection between the words *death* and *killed*, but being different parts of speech this would be easily missed by traditional distributional methods.

Consequently, in order to demonstrate the potential of our method, we have taken the phrases "reduce the risk of diseases", "has health benefits", "charged in the death of" and "accused of having killed", constructed corpora for the phrases and their components and then computed distributional similarity between pairs of phrases and their respective components. Under our hypotheses, paraphrases will be more similar than non-paraphrases and there will be no clear relation between the similarity of phrases as a whole and the similarity of their components.

We now discuss corpus construction and distributional similarity calculation in more detail.

4.1 Corpus Construction

In order to compute distributional similarity between sub-parses, we need to have seen a large number of occurrences of each sub-parse. Since data sparseness rules out using traditional corpora, such as the British National Corpus (BNC), we constructed a corpus for each phrase by mining the web. We also constructed a similar corpus for each component of each phrase. For example, for phrase 1, we constructed corpora for "reduce the risk of diseases", "reduce" and "the risk of diseases". We do this in order to avoid only have occurrences of the components in the context of the larger phrase. Each corpus was constructed by sending the phrase as a quoted string to Altavista. We took the returned list of URLs (up to the top 1000 where more than 1000 could be returned), removed duplicates and then downloaded the associated files. We then searched the files for the lines containing the relevant string and added each of these to the corpus file for that phrase. Each corpus file was then parsed using the RASP parser (version 3.β) ready for feature extraction.

Phrase	Types	Tokens
reduce the risk of diseases	156	389
reduce	3652	14082
the risk of diseases	135	947
has health benefits	340	884
has	3709	10221
health benefits	143	301
charged in the death of	624	1739
charged in	434	1011
the death of	348	1440
accused of having killed	88	173
accused of	679	1760
having killed	569	1707

Table 1: Number of feature types and tokens extracted for each Phrase

4.2 Computing Distributional Similarity

First, a feature extractor is run over each parsed corpus file to extract occurrences of the sub-parse and their features. The feature extractor reads in a template for each phrase in the form of dependency relations over lemmas. It checks each sentence parse against the template (taking care that the same word form is indeed the same occurrence of the word in the sentence). When a match is found, the other grammatical relations[5] for each word in the sub-parse are output as features. When the sub-parse is only a word, the process is simplified to finding grammatical relations containing that word.

The raw feature file is then converted into a co-occurrence vector by counting the occurrences of each feature type. Table 1 shows the number of feature types and tokens extracted for each phrase. This shows that we have extracted a reasonable number of features for each phrase, since distributional similarity techniques have been shown to work well for words which occur more than 100 times in a given corpus (Lin, 1998; Weeds and Weir, 2003).

We then computed the distributional similarity between each co-occurrence vector using the α-skew divergence measure (Lee, 1999). The α-skew divergence measure is an approximation to the Kullback-Leibler (KL) divergence meassure between two distributions p and q:

$$D(p||q) = \sum_x p(x) log \frac{p(x)}{q(x)}$$

[5]We currently retain all of the distinctions between grammatical relations output by RASP.

The α-skew divergence measure is designed to be used when unreliable maximum likelihood estimates (MLE) of probabilities would result in the KL divergence being equal to ∞. It is defined as:

$$dist_\alpha(q, r) = D(r||\alpha.q + (1 - \alpha).r)$$

where $0 \leq \alpha \leq 1$. We use $\alpha = 0.99$, since this provides a close approximation to the KL divergence measure. The result is a number greater than or equal to 0, where 0 indicates that the two distributions are identical. In other words, a smaller distance indicates greater similarity.

The reason for choosing this measure is that it can be used to compute the distance between any two co-occurrence vectors independent of any information about other words. This is in contrast to many other measures, e.g., Lin (1998), which use the co-occurrences of features with other words to compute a weighting function such as mutual information (MI) (Church and Hanks, 1989). Since we only have corpus data for the target phrases, it is not possible for us to use such a measure. However, the α-skew divergence measure has been shown (Weeds, 2003) to perform comparably with measures which use MI, particularly for lower frequency target words.

4.3 Results

The results, in terms of α-skew divergence scores between pairs of phrases, are shown in Table 2. Each set of three lines shows the similarity score between a pair of phrases and then between respective pairs of components. In the first two sets, the phrases are paraphrases whereas in the second two sets, the phrases are not.

From the table, there does appear to be some potential in the use of distributional similarity between sub-parses to identify potential paraphrases. In the final two examples, the paired phrases are not semantically similar, and as we would expect, their respective distributional similarities are less (i.e., they are further apart) than in the first two examples.

Further, we can see that there is no clear relation between the similarity of two phrases and the similarity of respective components. However in 3 out of 4 cases, the similarity between the phrases lies between that of their components. In every case, the similarity of the phrases is less than the similarity of the verbal components. This might be what one would expect for the second example since the components "charged in" and "accused of" are semantically similar. However, in the first example, we would have expected to see that the similarity between "reduce the risk of diseases" and "has health benefits" to be greater than either pair of components, which it is not. The reason for this is not clear from just these examples. However, possibilities include the distributional similarity measure used, the features selected from the corpus data and a combination of both. It may be that single words tend to exhibit greater similarity than phrases due to their greater relative frequencies. As a result, it may be necessary to factor in the length or frequency of a sub-parse into distributional similarity calculations or comparisons thereof.

Phrase 1	Phrase 2	Dist.
reduce the risk of diseases	has health benefits	5.28
reduce	has	4.95
the risk of diseases	health benefits	5.58
charged in the death of	accused of having killed	5.07
charged in	accused of	4.86
the death of	having killed	6.16
charged in the death of	has health benefits	6.04
charged in	has	5.54
the death of	health benefits	4.70
reduce the risk of diseases	accused of having killed	6.09
reduce	accused of	5.77
the risk of diseases	having killed	6.31

Table 2: α-skew divergence scores between pairs of phrases

5 Conclusions and Further Work

In conclusion, it is clear that components of phrases do not need to be semantically similar for the encompassing phrases to be semantically similar. Thus, it is necessary to develop techniques which estimate the semantic similarity of two phrases directly rather than combining similarity scores calculated for pairs of words.

Our approach is to find the distributional similarity of the sub-parses associated with phrases by extending general techniques for finding lexical distributional similarity. We have illustrated this method for examples, showing how data sparseness can be overcome using the web.

We have shown that finding the distributional similarity between phrases, as outlined here, may have potential in identifying paraphrases. In our examples, the distributional similarities of paraphrases was higher than non-paraphrases. However, obviously, more extensive evaluation of the technique is required before drawing more definite conclusions.

In this respect, we are currently in the process of developing a gold standard set of similar phrases from the Pascal Textual Entailment Chal-

lenge dataset. This task is not trivial since, even though pairs of sentences are already identified as potential paraphrases, it is still necessary to extract pairs of phrases which convey roughly the same meaning. This is because 1) some pairs of sentences are almost identical in word content and 2) some pairs of sentences are quite distant in meaning similarity. Further, it is also desirable to classify extracted pairs of paraphrases as to whether they are lexical, syntactic, semantic or inferential in nature. Whilst lexical (e.g. "to gather" is similar to "to collect") and syntactic (e.g. "Cambodian sweatshop" is equivalent to "sweatshop in Cambodia") are of interest, our aim is to extend lexical techniques to the semantic level (e.g. "X won presidential election" is similar to "X became president"). Once our analysis is complete, the data will be used to evaluate variations on the technique proposed herein and also to compare it empirically to other techniques such as that of Lin and Pantel (2001).

References

Regina Barzilay and Noemie Elhadad. 2003. Sentence alignment for monolingual comparable corpora. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP2003)*, pages 25–33, Sapporo, Japan.

Edward Briscoe and John Carroll. 1995. Developing and evaluating a probabilistic lr parser of part-of-speech and punctuation labels. In *4th ACL/SIGDAT International Workshop on Parsing Technologies*, pages 48–58.

Kenneth W. Church and Patrick Hanks. 1989. Word association norms, mutual information and lexicography. In *Proceedings of the 27th Annual Conference of the Association for Computational Linguistics (ACL-1989)*, pages 76–82.

James R. Curran and Marc Moens. 2002. Improvements in automatic thesaurus extraction. In *ACL-SIGLEX Workshop on Unsupervised Lexical Acquisition*, Philadelphia.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Proceedings of the Recognising Textual Entailment Challenge 2005*.

Bill Dolan, Chris Brockett, and Chris Quirk. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th International Conference on Computational Linguistics*, Geneva.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Maayan Geffet and Ido Dagan. 2004. Feature vector quality and distributional similarity. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING-2004)*, pages 247–253, Geneva.

Zelig S. Harris. 1968. *Mathematical Structures of Language*. Wiley, New York.

Jesus Herrera, Anselmo Penas, and Felisa Verdejo. 2005. Textual entailment recognition based on dependency analysis and wordnet. In *Proceedings of the Recognising Textual Entailment Challenge 2005*, April.

Valentin Jijkoun and Maarten de Rijke. 2005. Recognising textual entailment using lexical similarity. In *Proceedings of the Recognising Textual Entailment Challenge 2005*, April.

Lillian Lee. 1999. Measures of distributional similarity. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL-1999)*, pages 23–32.

Dekang Lin and Patrick Pantel. 2001. Discovery of inference rules for question answering. *Natural Language Engineering*, 7(4):343–360.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics (COLING-ACL '98)*, pages 768–774, Montreal.

Tim Owen, Ian Wakeman, Bill Keller, Julie Weeds, and David Weir. 2005. Managing the policies of non-technical users in a dynamic world. In *IEEE Workshop on Policy in Distributed Systems*, Stockholm, Sweden, May.

Darren Pearce. 2001. Synonymy in collocation extraction. In *Proceedings of the NAACL Workshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, Carnegie Mellon University, Pittsburgh.

Idan Szpektor, Hristo Tanev, Ido Dagan, and Bonaventura Coppola. 2004. Scaling web-based acquisition of entailment relations. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP) 2004*, Barcelona.

Julie Weeds and David Weir. 2003. A general framework for distributional similarity. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-2003)*, Sapporo, Japan.

Julie Weeds, Bill Keller, David Weir, Tim Owen, and Ian Wakemna. 2004. Natural language expression of user policies in pervasive computing environments. In *Proceedings of OntoLex2004, LREC Workshop on Ontologies and Lexical Resources in Distributed Environments*, Lisbon, Portugal, May.

Julie Weeds. 2003. *Measures and Applications of Lexical Distributional Similarity*. Ph.D. thesis, Department of Informatics, University of Sussex.