ACL-05

Empirical Modeling of Semantic Equivalence and Entailment

Proceedings of the Workshop

30 June 2005
University of Michigan
Ann Arbor, Michigan, USA

Production and Manufacturing by
Omnipress Inc.
Post Office Box 7214
Madison, WI 53707-7214

Sponsorship gratefully received from
Microsoft Research
One Microsoft Way
Redmond, Washington 98052, USA

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
75 Paterson Street, Suite 9
New Brunswick, NJ 08901
USA
Tel: +1-732-342-9100
Fax: +1-732-342-9339
acl@aclweb.org

Introduction

The last few years have seen a surge in interest in modeling techniques aimed at measuring semantic equivalence and entailment, with work on paraphrase acquisition/generation, WordNet-based expansion, distributional similarity, supervised learning of semantic variability in information extraction, and the identification of patterns in template-based QA. Being able to identify when two strings ”mean the same thing” or that one entails the other are crucial abilities for a broad range of NLP-related applications, ranging from question answering to summarization.

These proceedings contain a rich variety of papers centered on the problem of modeling semantic overlap between linguistic strings. This is a difficult problem space, encompassing issues of lexical choice, syntactic alternation, semantic inference, and reference/discourse structure.

We were pleased by the strong level of interest in the workshop, which resulted in a number of high-quality submissions. Each paper was blind-reviewed by 2-3 members of the Program Committee, and we were forced to make some difficult choices in determining the final schedule.

This workshop is intended to bring together people working on empirical, application-independent approaches to the practical problems of semantic inference. While different applications face similar underlying semantic problems, these problems have typically been addressed in an application-specific manner. In the absence of a generic evaluation framework, it is difficult to compare semantic methods that were developed for different applications. We are particularly hopeful that the workshop will help foster discussion around common datasets and evaluation strategies that will help guide future work in this area.

We would like to express our deepest gratitude to the hard-working members of the program committee. We’d also like to thank Mirella Lapata, Jason Eisner, Philipp Koehn, and Dragomir Radev for their organizational help.

We hope you enjoy this workshop!

Bill Dolan and Ido Dagan

Organizers:

Bill Dolan, Microsoft Research
Ido Dagan, Bar Ilan University

Program Committee:

Srinivas Bangalore, AT&T Research
Regina Barzilay, MIT
Chris Brockett, Microsoft Research
Pascale Fung, Hong Kong University of Science and Technology
Oren Glickman, Bar Ilan University
Cyril Goutte, Xerox Research Centre Europe
Ed Hovy, ISI
Kentaro Inui, Nara Institute of Science and Technology
Dekang Lin, University of Alberta
Daniel Marcu, ISI
Kathy McKeown, Columbia University
Dan Moldovan, University of Texas at Dallas
Chris Quirk, Microsoft Research
Maarten de Rijke, University of Amsterdam
Hinrich Schuetze, University of Stuttgart
Satoshi Sekine, New York University
Peter Turney, National Research Council of Canada

Invited Speaker:

Dan Roth, University of Illinois at Urbana-Champaign

Table of Contents

Conference Program

Thursday, June 30, 2005

9:00–9:15 Opening Remarks by Bill Dolan and Ido Dagan

9:15–9:40 *Classification of Semantic Relations by Humans and Machines*
Erwin Marsi and Emiel Krahmer

9:40–10:05 *The Distributional Similarity of Sub-Parses*
Julie Weeds, David Weir and Bill Keller

10:05–10:30 *Measuring the Semantic Similarity of Texts*
Courtney Corley and Rada Mihalcea

10:30–11:00 Break

11:00–11:25 *Training Data Modification for SMT Considering Groups of Synonymous Sentences*
Hideki Kashioka

11:25–12:25 Invited Talk by Dan Roth, University of Illinois at Urbana-Champaign

12:25–2:00 Lunch

2:00–2:25 *Recognizing Paraphrases and Textual Entailment Using Inversion Transduction Grammars*
Dekai Wu

2:25–2:50 *Local Textual Inference: Can it be Defined or Circumscribed?*
Annie Zaenen, Lauri Karttunen and Richard Crouch

2:50–3:15 *Discovering Entailment Relations Using "Textual Entailment Patterns"*
Fabio Massimo Zanzotto, Maria Teresa Pazienza and Marco Pennacchiotti

3:15–3:40 *A Probabilistic Setting and Lexical Coocurrence Model for Textual Entailment*
Oren Glickman and Ido Dagan

3:40–4:00 Break

Thursday, June 30, 2005 (continued)

4:00–4:25 *Generating an Entailment Corpus from News Headlines*
John Burger and Lisa Ferro

4:25–4:50 *Definition and Analysis of Intermediate Entailment Levels*
Roy Bar-Haim, Idan Szpecktor and Oren Glickman

4:50–5:40 Panel Discussion

Classification of semantic relations by humans and machines *

Erwin Marsi and **Emiel Krahmer**
Communication and Cognition
Tilburg University, The Netherlands
{e.c.marsi, e.j.krahmer}@uvt.nl

Abstract

This paper addresses the classification of semantic relations between pairs of sentences extracted from a Dutch parallel corpus at the word, phrase and sentence level. We first investigate the performance of human annotators on the task of manually aligning dependency analyses of the respective sentences and of assigning one of five semantic relations to the aligned phrases (equals, generalizes, specifies, restates and intersects). Results indicate that humans can perform this task well, with an F-score of .98 on alignment and an F-score of .95 on semantic relations (after correction). We then describe and evaluate a combined alignment and classification algorithm, which achieves an F-score on alignment of .85 (using EuroWordNet) and an F-score of .80 on semantic relation classification.

1 Introduction

An automatic method that can determine how two sentences relate to each other in terms of **semantic overlap** or **textual entailment** (e.g., (Dagan and Glickman, 2004)) would be a very useful thing to have for robust natural language applications. A summarizer, for instance, could use it to extract the most informative sentences, while a question-answering system – to give a second example – could use it to select potential answer string (Punyakanok et al., 2004), perhaps preferring more specific answers over more general ones. In general, it is very useful to know whether some sentence S is more specific (entails) or more general than (is entailed by) an alternative sentence S', or whether the two sentences express essentially the same information albeit in a different way (paraphrasing).

Research on automatic methods for recognizing semantic relations between sentences is still relatively new, and many basic issues need to be resolved. In this paper we address two such related issues: (1) to what extent can human annotators label semantic overlap relations between words, phrases and sentences, and (2) what is the added value of linguistically informed analyses.

It is generally assumed that pure string overlap is not sufficient for recognizing semantic relations; and that using some form of syntactic analysis may be beneficial (e.g., (Herrera et al., 2005), (Vanderwende et al., 2005)). Our working hypothesis is that semantic overlap at the word and phrase levels may provide a good basis for deciding the semantic relation between sentences. Recognising semantic relations between sentences then becomes a two-step procedure: first, the words and phrases in the respective sentences need to be aligned, after which the relations between the pairs of aligned words and phrases should be labeled in terms of semantic relations.

Various alignment algorithms have been developed for data-driven approaches to machine translation (e.g. (Och and Ney, 2000)). Initially work focused on word-based alignment, but more and more work is also addressing alignment at the higher levels (substrings, syntactic phrases or trees), e.g., (Meyers et al., 1996), (Gildea, 2003). For our purposes, an additional advantage of aligning syntactic structures is that it keeps the alignment feasible (as the number of arbitrary substrings that may be aligned grows exponentially to the number of words

*This work was carried out within the IMIX-IMOGEN (Interactive Multimodal Output Generation) project, sponsored by the Netherlands Organization of Scientific Research (NWO).

Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, pages 1–6,
Ann Arbor, June 2005.

in the sentence). Here, following (Herrera et al., 2005) and (Barzilay, 2003), we will align sentences at the level of **dependency structures**. In addition, we will label the alignments in terms of five basic semantic relations to be defined below. We will perform this task both manually and automatically, so that we can address both of the issues raised above.

Section 2 describes a monolingual parallel corpus consisting of two Dutch translations, and formalizes the alignment-classification task to be performed. In section 3 we report the results on alignment, first describing interannotator agreement on this task and then the results on automatic alignment. In section 4, then, we address the semantic relation classification; again, first describing interannotator results, followed by results obtained using memory-based machine learning techniques. We end with a general discussion.

2 Corpus and Task definition

2.1 Corpus

We have developed a **parallel monolingual corpus** consisting of two different Dutch translations of the French book "Le petit prince" (*the little prince*) by Antoine de Saint-Exupéry (published 1943), one by Laetitia de Beaufort-van Hamel (1966) and one by Ernst Altena (2000). For our purposes, this proved to be a good way to quickly find a large enough set of related sentence pairs, which differ semantically in interesting and subtle ways. In this work, we used the first five chapters, with 290 sentences and 3600 words in the first translation, and 277 sentences and 3358 words in the second translation. The texts were automatically tokenized and split into sentences, after which errors were manually corrected. Corresponding sentences from both translations were manually aligned; in most cases this was a one-to-one mapping, but occasionally a single sentence in one translation mapped onto two or more sentences in the other: this occurred 23 times in all five chapters. Next, the **Alpino** parser for Dutch (e.g., (Bouma et al., 2001)) was used for part-of-speech tagging and lemmatizing all words, and for assigning a dependency analysis to all sentences. The POS labels indicate the major word class (e.g. *verb*, *noun*, *adj*, and *adv*). The dependency relations hold between tokens and are identical to those used in the Spoken Dutch Corpus. These include dependencies such as *head/subject*, *head/modifier* and *coordination/conjunction*. If a full parse could not be obtained, Alpino produced partial analyses collected under a single root node. Errors in lemmatization, POS tagging, and syntactic dependency parsing were not subject to manual correction.

2.2 Task definition

The task to be performed can be described informally as follows: given two dependency analyses, align those nodes that are semantically related. More precisely: For each node v in the dependency structure for a sentence S, we define $\text{STR}(v)$ as the substring of all tokens under v (i.e., the composition of the tokens of all nodes reachable from v). An alignment between sentences S and S' pairs nodes from the dependency graphs for both sentences. Aligning node v from the dependency graph D of sentence S with node v' from the graph D' of S' indicates that there is a semantic relation between $\text{STR}(v)$ and $\text{STR}(v')$, that is, between the respective substrings associated with v and v'. We distinguish five potential, mutually exclusive, relations between nodes (with illustrative examples):

1. v **equals** v' iff $\text{STR}(v)$ and $\text{STR}(v')$ are literally identical (abstracting from case). Example: "a small and a large boa-constrictor" equals "a large and a small boa-constrictor";
2. v **restates** v' iff $\text{STR}(v)$ is a paraphrase of $\text{STR}(v')$ (same information content but different wording). Example: "a drawing of a boa-constrictor snake" restates "a drawing of a boa-constrictor";
3. v **specifies** v' iff $\text{STR}(v)$ is more specific than $\text{STR}(v')$. Example: "the planet B 612" specifies "the planet";
4. v **generalizes** v' iff $\text{STR}(v')$ is more specific than $\text{STR}(v)$. Example: "the planet" generalizes "the planet B 612";
5. v **intersects** v' iff $\text{STR}(v)$ and $\text{STR}(v')$ share some informational content, but also each express some piece of information not expressed in the other. Example: "Jupiter and Mars" intersects "Mars and Venus"

Figure 1 shows an example alignment with semantic relations between the dependency structures of

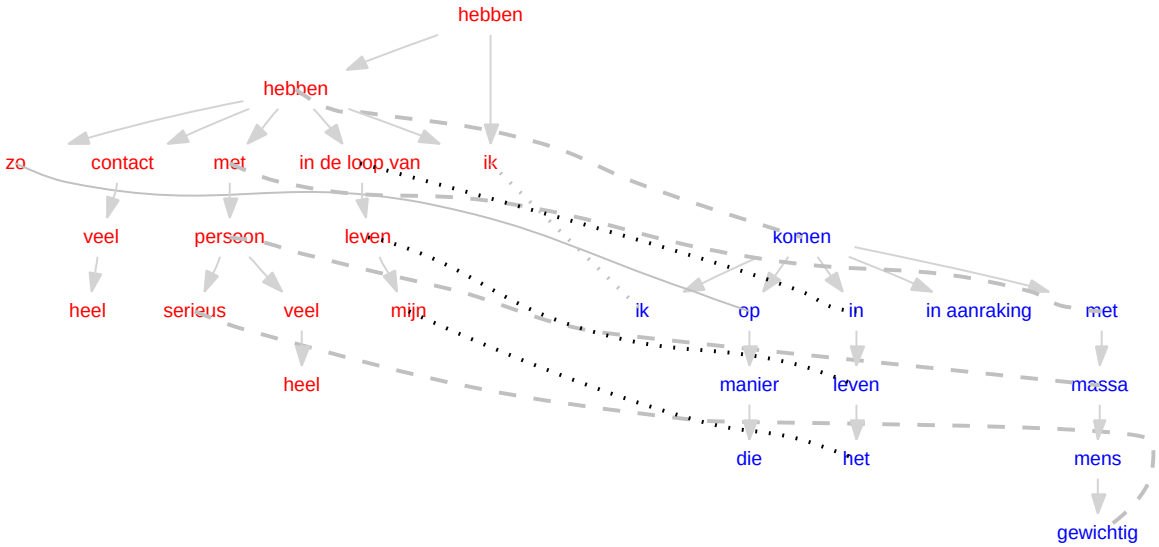

Figure 1: Dependency structures and alignment for the sentences *Zo heb ik in de loop van mijn leven heel veel contacten gehad met heel veel serieuze personen.* (lit. 'Thus have I in the course of my life very many contacts had with very many serious persons') and *Op die manier kwam ik in het leven met massa's gewichtige mensen in aanraking.*. (lit. 'In that way came I in the life with mass-of weighty/important people in touch'). The alignment relations are *equals* (dotted gray), *restates* (solid gray), *specifies* (dotted black), and *intersects* (dashed gray). For the sake of transparency, dependency relations have been omitted.

two sentences. Note that there is an intuitive relation with entailment here: both *equals* and *restates* can be understood as mutual entailment (i.e., if the root nodes of the analyses corresponding S and S' stand in an equal or restate relation, S entails S' and S' entails S), if S *specifies* S' then S also entails S' and if S *generalizes* S' then S is entailed by S'.

In remainder of this paper, we will distinguish two aspects of this task: **alignment** is the subtask of pairing related nodes – or more precise, pairing the token strings corresponding to these nodes; **classification of semantic relations** is the subtask of labeling these alignments in terms of the five types of semantic relations.

2.3 Annotation procedure

For creating manual alignments, we developed a special-purpose annotation tool which shows, side by side, two sentences, as well as their respective dependency graphs. When the user clicks on a node v in the graph, the corresponding string ($\text{STR}(v)$) is shown at the bottom. The tool enables the user to manually construct an alignment graph on the basis of the respective dependency graphs. This is done by focusing on a node in the structure for one sentence, and then selecting a corresponding node (if possible) in the other structure, after which the user can select the relevant alignment relation. The tool offers additional support for folding parts of the graphs, highlighting unaligned nodes and hiding dependency relation labels.

All text material was aligned by the two authors. They started with annotating the first ten sentences of chapter one together in order to get a feel for the task. They continued with the remaining sentences from chapter one individually (35 sentences and 521 in the first translation, and 35 sentences and 481 words in the second translation). Next, both annotators discussed annotation differences, which triggered some revisions in their respective annotation. They also agreed on a single consensus annotation. Interannotator agreement will be discussed in the next two sections. Finally, each author annotated two additional chapters, bringing the total to five.

3 Alignment

3.1 Interannotator agreement

Interannotator agreement was calculated in terms of precision, recall and F-score (with $\beta = 1$) on aligned

	(A_1, A_2)	$(A_{1'}, A_{2'})$	$(A_c, A_{1'})$	$(A_c, A_{2'})$
#real:	322	323	322	322
#pred:	312	321	323	321
#correct:	293	315	317	318
precision:	.94	.98	.98	.99
recall:	.91	.98	.98	.99
F-score:	.92	.98	.98	.99

Table 1: Interannotator agreement with respect to alignment between annotators 1 and 2 before (A_1, A_2) and after $(A_{1'}, A_{2'})$ revision , and between the consensus and annotator 1 $(A_c, A_{1'})$ and annotator 2 $(A_c, A_{2'})$ respectively.

node pairs as follows:

$$precision = \mid A_{real} \cap A_{pred} \mid / \mid A_{pred} \mid \quad (1)$$

$$recall = \mid A_{real} \cap A_{pred} \mid / \mid A_{real} \mid \quad (2)$$

$$F\text{-}score = \ (2 \times prec \times rec) \ / \ (prec + rec) \quad (3)$$

where A_{real} is the set of all real alignments (the reference or golden standard), A_{pred} is the set of all predicted alignments, and $A_{pred} \cap A_{real}$ is the set all correctly predicted alignments. For the purpose of calculating interannotator agreement, one of the annotations (A_1) was considered the 'real' alignment, the other (A_2) the 'predicted'. The results are summarized in Table 1 in column (A_1, A_2).[1]

As explained in section 2.3, both annotators revised their initial annotations. This improved their agreement, as shown in column $(A_{1'}, A_{2'})$. In addition, they agreed on a single consensus annotation (A_c). The last two columns of Table 1 show the results of evaluating each of the revised annotations against this consensus annotation. The F-score of .98 can therefore be regarded as the upper bound on the alignment task.

3.2 Automatic alignment

Our tree alignment algorithm is based on the dynamic programming algorithm in (Meyers et al., 1996), and similar to that used in (Barzilay, 2003). It calculates the match between each node in dependency tree D against each node in dependency tree D'. The score for each pair of nodes only depends on the similarity of the words associated with the nodes and, recursively, on the scores of the best matching pairs of their descendants. The node similarity function relies either on identity of the lemmas or on synonym, hyperonym, and hyponym relations between them, as retrieved from EuroWordNet.

Automatic alignment was evaluated with the consensus alignment of the first chapter as the gold standard. A baseline was constructed by aligning those nodes which stand in an *equals* relation to each other, i.e., a node v in D is aligned to a node v' in D' iff $\text{STR}(v) = \text{STR}(v')$. This baseline already achieves a relatively high score (an F-score of .56), which may be attributed to the nature of our material: the translated sentence pairs are relatively close to each other and may show a sizeable amount of literal string overlap. In order to test the contribution of synonym and hyperonym information for node matching, performance is measured with and without the use of EuroWordNet. The results for automatic alignment are shown in Table 2. In comparison with the baseline, the alignment algorithm without use of EuroWordnet loses a few points on precision, but improves a lot on recall (a 200% increase), which in turn leads to a substantial improvement on the overall F-score. The use of EurWordNet leads to a small increase (two points) on both precision and recall, and thus to small increase in F-score. However, in comparison with the gold standard human score for this task (.95), there is clearly room for further improvement.

4 Classification of semantic relations

4.1 Interannotator agreement

In addition to alignment, the annotation procedure for the first chapter of *The little prince* by two annotators (cf. section 2.3) also involved labeling of the semantic relation between aligned nodes. Interannotator agreement on this task is shown Table 3, before and after revision. The measures are *weighted* precision, recall and F-score. For instance, the precision is the weighted sum of the separate precision scores for each of the five relations. The table also shows the κ-score. The F-score of .97 can be regarded as the upper bound on the relation labeling task. We think these numbers indicate that the classification of semantic relations is a well defined task which can be accomplished with a high level of interannotator agreement.

[1]Note that since there are no classes, we can not calculate change agreement rethe $Kappa$ statistic.

Alignment :	*Prec* :	*Rec* :	*F-score:*
baseline	.87	.41	.56
algorithm without wordnet	.84	.82	.83
algorithm with wordnet	.86	.84	.85

Table 2: Precision, recall and F-score on automatic alignment

	(A_1, A_2)	$(A_{1'}, A_{2'})$	$(A_c, A_{1'})$	$(A_c, A_{2'})$
precision:	.86	.96	.98	.97
recall:	.86	.95	.97	.97
F-score:	.85	.95	.97	.97
κ:	.77	.92	.96	.96

Table 3: Interannotator agreement with respect to semantic relation labeling between annotators 1 and 2 before (A_1, A_2) and after $(A_{1'}, A_{2'})$ revision , and between the consensus and annotator 1 $(A_c, A_{1'})$ and annotator 2 $(A_c, A_{2'})$ respectively.

4.2 Automatic classification

For the purpose of *automatic* semantic relation labeling, we approach the task as a classification problem to be solved by machine learning. Alignments between node pairs are classified on the basis of the lexical-semantic relation between the nodes, their corresponding strings, and – recursively – on previous decisions about the semantic relations of daughter nodes. The input features used are:

- a boolean feature representing string identity between the strings corresponding to the nodes
- a boolean feature for each of the five semantic relations indicating whether the relation holds for at least one of the daughter nodes;
- a boolean feature indicating whether at least one of the daughter nodes is *not* aligned;
- a categorical feature representing the lexical semantic relation between the nodes (i.e. the lemmas and their part-of-speech) as found in EuroWordNet, which can be *synonym*, *hyperonym*, or *hyponym*.[2]

To allow for the use of previous decisions, the nodes of the dependency analyses are traversed in a bottom-up fashion. Whenever a node is aligned, the classifier assigns a semantic label to the alignment. Taking previous decisions into account may cause a proliferation of errors: wrong classification of daughter nodes may in turn cause wrong classification of the mother node. To investigate this risk, classification experiments were run both with and without (i.e. using the annotation) previous decisions.

[2]These three form the bulk of all relations in Dutch EuroWordnet. Since no word sense disambiguation was involved, we simply used all word senses.

	Prec :	*Rec* :	*F-score:*
equals	$.93 \pm .06$	$.95 \pm .04$	$.94 \pm .02$
restates	$.56 \pm .08$	$.78 \pm .04$	$.65 \pm .05$
specifies	$n.a.$	0	$n.a.$
generalizes	$.19 \pm .06$	$.37 \pm .09$	$.24 \pm .05$
intersects	$n.a.$	0	$n.a.$
Combined:	$.62 \pm .01$	$.70 \pm .02$	$.64 \pm .02$

Table 4: Average precision, recall and F-score (and SD) over all 5 folds on automatic classification of semantic relations

Since our amount of data is limited, we used a memory-based classifier, which – in contrast to most other machine learning algorithms – performs no abstraction, allowing it to deal with productive but low-frequency exceptions typically occurring in NLP tasks(Daelemans et al., 1999). All memory-based learning was performed with TiMBL, version 5.1 (Daelemans et al., 2004), with its default settings (overlap distance function, gain-ratio feature weighting, $k = 1$).

The five first chapters of *The little prince* were used to run a 5-fold cross-validated classification experiment. The first chapter is the consensus alignment and relation labeling, while the other four were done by one out of two annotators. The alignments to be classified are those from to the *human* alignment. The baseline of always guessing *equals* – the majority class – gives a precision of 0.26, a recall of 0.51, and an F-score of 0.36. Table 4 presents the results broken down to relation type. The combined F-score of 0.64 is almost twice the baseline score. As expected, the highest score goes to *equals*, followed by a reasonable score on *restates*. Performance on the other relation types is rather poor, with even no predictions of *specifies* and *intersects* at all.

Faking perfect previous decisions by using the annotation gives a considerable improvement, as shown in Table 5, especially on *specifies*, *generalizes* and *intersects*. This reveals that the proliferation of classification errors is indeed a problem that should be addressed.

	$Prec:$	$Rec:$	*F-score:*
equals	$.99 \pm .02$	$.97 \pm .02$	$.98 \pm .01$
restates	$.65 \pm .04$	$.82 \pm .04$	$.73 \pm .03$
specifies	$.60 \pm .12$	$.48 \pm .10$	$.53 \pm .09$
generalizes	$.50 \pm .11$	$.52 \pm .10$	$.50 \pm .09$
intersects	$.69 \pm .27$	$.35 \pm .12$	$.46 \pm .16$
Combined:	$.82 \pm .02$	$.81 \pm .02$	$.80 \pm .02$

Table 5: Average precision, recall and F-score (and SD) over all 5 folds on automatic classification of semantic relations without using previous decisions.

In sum, these results show that automatic classification of semantic relations is feasible and promising – especially when the proliferation of classification errors can be prevented – but still not nearly as good as human performance.

5 Discussion and Future work

This paper presented an approach to detecting semantic relations at the word, phrase and sentence level on the basis of dependency analyses. We investigated the performance of human annotators on the tasks of manually aligning dependency analyses and of labeling the semantic relations between aligned nodes. Results indicate that humans can perform this task well, with an F-score of $.98$ on alignment and an F-score of $.92$ on semantic relations (after revision). We also described and evaluated automatic methods addressing these tasks: a dynamic programming tree alignment algorithm which achieved an F-score on alignment of $.85$ (using lexical semantic information from EuroWordNet), and a memory-based semantic relation classifier which achieved F-scores of $.64$ and $.80$ with and without using real previous decisions respectively.

One of the issues that remains to be addressed in future work is the effect of parsing errors. Such errors were not corrected, but during manual alignment, we sometimes found that substrings could not be properly aligned because the parser had failed to identify them as syntactic constituents. As far as classification of semantic relations is concerned, the proliferation of classification errors is an issue that needs to be solved. Classification performance may be further improved with additional features (e.g. phrase length information), optimization, and more data. Also, we have not yet tried to combine automatic alignment and classification. Yet another point concerns the type of text material. The sentence pairs from our current corpus are relatively close, in the sense that both translations more or less convey the same information. Although this seems a good starting point to study alignment, we intend to continue with other types of text material in future work. For instance, in extending our work to the actual output of a QA system, we expect to encounter sentences with far less overlap.

References

R. Barzilay. 2003. *Information Fusion for Multidocument Summarization*. Ph.D. Thesis, Columbia University.

G. Bouma, G. van Noord, and R. Malouf. 2001. Alpino: Wide-coverage computational analysis of Dutch. In *Computational Linguistics in The Netherlands 2000*, pages 45–59.

W. Daelemans, A. Van den Bosch, and J. Zavrel. 1999. Forgetting exceptions is harmful in language learning. *Machine Learning, Special issue on Natural Language Learning*, 34:11–41.

W. Daelemans, J. Zavrel, K. Van der Sloot, and A. van den Bosch. 2004. TiMBL: Tilburg memory based learner, version 5.1, reference guide. ILK Technical Report 04-02, Tilburg University.

I. Dagan and O. Glickman. 2004. Probabilistic textual entailment: Generic applied modelling of language variability. In *Learning Methods for Text Understanding and Mining*, Grenoble.

D. Gildea. 2003. Loosely tree-based alignment for machine translation. In *Proceedings of the 41st Annual Meeting of the ACL*, Sapporo, Japan.

J. Herrera, A. Pe nas, and F. Verdejo. 2005. Textual entailment recognition based on dependency analysis and wordnet. In *Proceedings of the 1st. PASCAL Recognision Textual Entailment Challenge Workshop*. Pattern Analysis, Statistical Modelling and Computational Learning, PASCAL.

A. Meyers, R. Yangarber, and R. Grisham. 1996. Alignment of shared forests for bilingual corpora. In *Proceedings of 16th International Conference on Computational Linguistics (COLING-96)*, pages 460–465, Copenhagen, Denmark.

F.J. Och and H. Ney. 2000. Statistical machine translation. In *EAMT Workshop*, pages 39–46, Ljubljana, Slovenia.

V. Punyakanok, D. Roth, and W. Yih. 2004. Natural language inference via dependency tree mapping: An application to question answering. *Computational Linguistics*, 6(9).

L. Vanderwende, D. Coughlin, and W. Dolan. 2005. What syntax can contribute in entailment task. In *Proceedings of the 1st. PASCAL Recognision Textual Entailment Challenge Workshop*, Southampton, U.K.

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.

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.

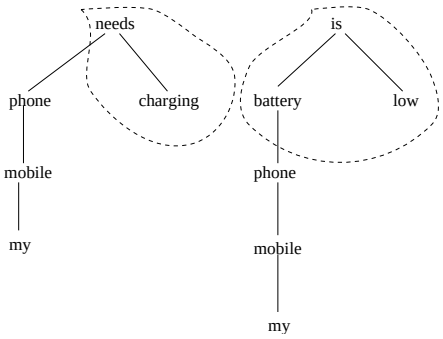

Figure 1: Parse trees for "my mobile phone needs charging" and "my mobile phone battery is low"

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.

Measuring the Semantic Similarity of Texts

Courtney Corley and **Rada Mihalcea**
Department of Computer Science
University of North Texas
{corley,rada}@cs.unt.edu

Abstract

This paper presents a knowledge-based method for measuring the semantic-similarity of texts. While there is a large body of previous work focused on finding the semantic similarity of concepts and words, the application of these word-oriented methods to text similarity has not been yet explored. In this paper, we introduce a method that combines word-to-word similarity metrics into a text-to-text metric, and we show that this method outperforms the traditional text similarity metrics based on lexical matching.

1 Introduction

Measures of text similarity have been used for a long time in applications in natural language processing and related areas. One of the earliest applications of text similarity is perhaps the vectorial model in information retrieval, where the document most relevant to an input query is determined by ranking documents in a collection in reversed order of their *similarity* to the given query (Salton and Lesk, 1971). Text similarity has been also used for relevance feedback and text classification (Rocchio, 1971), word sense disambiguation (Lesk, 1986), and more recently for extractive summarization (Salton et al., 1997b), and methods for automatic evaluation of machine translation (Papineni et al., 2002) or text summarization (Lin and Hovy, 2003).

The typical approach to finding the similarity between two text segments is to use a simple lexical matching method, and produce a similarity score based on the number of lexical units that occur in both input segments. Improvements to this simple method have considered stemming, stop-word removal, part-of-speech tagging, longest subsequence matching, as well as various weighting and normalization factors (Salton et al., 1997a). While successful to a certain degree, these lexical matching similarity methods fail to identify the *semantic* similarity of texts. For instance, there is an obvious similarity between the text segments *I own a dog* and *I have an animal*, but most of the current text similarity metrics will fail in identifying any kind of connection between these texts. The only exception to this trend is perhaps the latent semantic analysis (LSA) method (Landauer et al., 1998), which represents an improvement over earlier attempts to use measures of semantic similarity for information retrieval (Voorhees, 1993), (Xu and Croft, 1996). LSA aims to find similar terms in large text collections, and measure similarity between texts by including these additional related words. However, to date LSA has not been used on a large scale, due to the complexity and computational cost associated with the algorithm, and perhaps also due to the "black-box" effect that does not allow for any deep insights into why some terms are selected as similar during the singular value decomposition process.

In this paper, we explore a knowledge-based method for measuring the semantic similarity of texts. While there are several methods previously proposed for finding the semantic similarity of words, to our knowledge the application of these word-oriented methods to text similarity has not been yet explored. We introduce an algorithm

Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, pages 13–18,
Ann Arbor, June 2005.

that combines the word-to-word similarity metrics into a text-to-text semantic similarity metric, and we show that this method outperforms the simpler lexical matching similarity approach, as measured in a paraphrase identification application.

2 Measuring Text Semantic Similarity

Given two input text segments, we want to automatically derive a score that indicates their similarity at *semantic* level, thus going beyond the simple lexical matching methods traditionally used for this task. Although we acknowledge the fact that a comprehensive metric of text semantic similarity should take into account the relations between words, as well as the role played by the various entities involved in the interactions described by each of the two texts, we take a first rough cut at this problem and attempt to model the semantic similarity of texts as a function of the semantic similarity of the component words. We do this by combining metrics of word-to-word similarity and language models into a formula that is a potentially good indicator of the semantic similarity of the two input texts.

2.1 Semantic Similarity of Words

There is a relatively large number of word-to-word similarity metrics that were previously proposed in the literature, ranging from distance-oriented measures computed on semantic networks, to metrics based on models of distributional similarity learned from large text collections. From these, we chose to focus our attention on six different metrics, selected mainly for their observed performance in natural language processing applications, e.g. malapropism detection (Budanitsky and Hirst, 2001) and word sense disambiguation (Patwardhan et al., 2003), and for their relatively high computational efficiency.

We conduct our evaluation using the following word similarity metrics: Leacock & Chodorow, Lesk, Wu & Palmer, Resnik, Lin, and Jiang & Conrath. Note that all these metrics are defined between concepts, rather than words, but they can be easily turned into a word-to-word similarity metric by selecting for any given pair of words those two meanings that lead to the highest concept-to-concept similarity. We use the WordNet-based implementation of these metrics, as available in the WordNet::Similarity package (Patwardhan et al., 2003). We provide below a short description for each of these six metrics.

The **Leacock & Chodorow** (Leacock and Chodorow, 1998) similarity is determined as:

$$Sim_{lch} = -\log \frac{length}{2 * D} \quad (1)$$

where $length$ is the length of the shortest path between two concepts using node-counting, and D is the maximum depth of the taxonomy.

The **Lesk** similarity of two concepts is defined as a function of the overlap between the corresponding definitions, as provided by a dictionary. It is based on an algorithm proposed in (Lesk, 1986) as a solution for word sense disambiguation.

The **Wu and Palmer** (Wu and Palmer, 1994) similarity metric measures the depth of the two concepts in the WordNet taxonomy, and the depth of the least common subsumer (LCS), and combines these figures into a similarity score:

$$Sim_{wup} = \frac{2 * depth(LCS)}{depth(concept_1) + depth(concept_2)} \quad (2)$$

The measure introduced by **Resnik** (Resnik, 1995) returns the information content (IC) of the LCS of two concepts:

$$Sim_{res} = IC(LCS) \quad (3)$$

where IC is defined as:

$$IC(c) = -\log P(c) \quad (4)$$

and $P(c)$ is the probability of encountering an instance of concept c in a large corpus.

The next measure we use in our experiments is the metric introduced by **Lin** (Lin, 1998), which builds on Resnik's measure of similarity, and adds a normalization factor consisting of the information content of the two input concepts:

$$Sim_{lin} = \frac{2 * IC(LCS)}{IC(concept_1) + IC(concept_2)} \quad (5)$$

Finally, the last similarity metric we consider is **Jiang & Conrath** (Jiang and Conrath, 1997), which returns a score determined by:

$$Sim_{jnc} = \frac{1}{IC(concept_1) + IC(concept_2) - 2 * IC(LCS)} \quad (6)$$

2.2 Language Models

In addition to the semantic similarity of words, we also want to take into account the *specificity* of words, so that we can give a higher weight to a semantic matching identified between two very specific words (e.g. *collie* and *sheepdog*), and give less importance to the similarity score measured between generic concepts (e.g. *go* and *be*). While the specificity of words is already measured to some extent by their depth in the semantic hierarchy, we are reinforcing this factor with a corpus-based measure of word specificity, based on distributional information learned from large corpora.

Language models are frequently used in natural language processing applications to account for the distribution of words in language. While word frequency does not always constitute a good measure of word importance, the distribution of words across an entire collection can be a good indicator of the *specificity* of the words. Terms that occur in a few documents with high frequency contain a greater amount of discriminatory ability, while terms that occur in numerous documents across a collection with a high frequency have inherently less meaning to a document. We determine the *specificity* of a word using the inverse document frequency introduced in (Sparck-Jones, 1972), which is defined as the total number of documents in the corpus, divided by the total number of documents that include that word. In the experiments reported in this paper, we use the British National Corpus to derive the document frequency counts, but other corpora could be used to the same effect.

2.3 Semantic Similarity of Texts

Provided a measure of semantic similarity between words, and an indication of the word specificity, we combine them into a measure of text semantic similarity, by pairing up those words that are found to be most similar to each other, and weighting their similarity with the corresponding specificity score.

We define a *directional* measure of similarity, which indicates the semantic similarity of a text segment T_i *with respect to* a text segment T_j. This definition provides us with the flexibility we need to handle applications where the directional knowledge is useful (e.g. entailment), and at the same time it gives us the means to handle bidirectional similarity through a simple combination of two unidirectional metrics.

For a given pair of text segments, we start by creating *sets* of open-class words, with a separate set created for nouns, verbs, adjectives, and adverbs. In addition, we also create a set for cardinals, since numbers can also play an important role in the understanding of a text. Next, we try to determine pairs of similar words across the sets corresponding to the same open-class in the two text segments. For nouns and verbs, we use a measure of semantic similarity based on WordNet, while for the other word classes we apply lexical matching[1].

For each noun (verb) in the set of nouns (verbs) belonging to one of the text segments, we try to identify the noun (verb) in the other text segment that has the highest semantic similarity ($maxSim$), according to one of the six measures of similarity described in Section 2.1. If this similarity measure results in a score greater than 0, then the word is added to the set of similar words for the corresponding word class WS_{pos}[2]. The remaining word classes: adjectives, adverbs, and cardinals, are checked for lexical similarity with their counter-parts and included in the corresponding word class set if a match is found.

The similarity between the input text segments T_i and T_j is then determined using a scoring function that combines the word-to-word similarities and the word specificity:

$$sim(T_i, T_j)_{T_i} = \frac{\sum_{pos}(\sum_{\mathbf{w}_k \in \{WS_{pos}\}} (maxSim(\mathbf{w}_k) * idf_{\mathbf{w}_k}))}{\sum_{\mathbf{w}_k \in \{T_{i_{pos}}\}} idf_{\mathbf{w}_k}} \quad (7)$$

This score, which has a value between 0 and 1, is a measure of the directional similarity, in this case computed with respect to T_i. The scores from both directions can be combined into a bidirectional similarity using a simple average function:

$$sim(T_i, T_j) = \frac{sim(T_i, T_j)_{T_i} + sim(T_i, T_j)_{T_j}}{2} \quad (8)$$

[1]The reason behind this decision is the fact that most of the semantic similarity measures apply only to nouns and verbs, and there are only one or two relatedness metrics that can be applied to adjectives and adverbs.

[2]All similarity scores have a value between 0 and 1. The similarity threshold can be also set to a value larger than 0, which would result in tighter measures of similarity.

Text Segment 1: The jurors were taken into the courtroom in groups of 40 and asked to fill out a questionnaire.

- Set_{NN} = {juror, courtroom, group, questionnaire}
  Set_{VB} = {be, take, ask, fill}
  Set_{RB} = {out}
  Set_{CD} = {40}

Text Segment 2: About 120 potential jurors were being asked to complete a lengthy questionnaire.

- Set_{NN} = {juror, questionnaire}
  Set_{VB} = {be, ask, complete}
  Set_{JJ} = {potential, lengthy}
  Set_{CD} = {120}

Figure 1: Two text segments and their corresponding word class sets

3 A Walk-Through Example

We illustrate the application of the text similarity measure with an example. Given two text segments, as shown in Figure 1, we want to determine a score that reflects their semantic similarity. For illustration purposes, we restrict our attention to one measure of word-to-word similarity, the **Wu & Palmer** metric.

First, the text segments are tokenized, part-of-speech tagged, and the words are inserted into their corresponding word class sets. The sets obtained for the given text segments are illustrated in Figure 1.

Starting with each of the two text segments, and for each word in its word class sets, we determine the most similar word from the corresponding set in the other text segment. As mentioned earlier, we seek a WordNet-based semantic similarity for nouns and verbs, and only lexical matching for adjectives, adverbs, and cardinals. The word semantic similarity scores computed starting with the first text segment are shown in Table 3.

Text 1	Text 2	maxSim	IDF
jurors	jurors	1.00	5.80
courtroom	jurors	0.30	5.23
questionnaire	questionnaire	1.00	3.57
groups	questionnaire	0.29	0.85
were	were	1.00	0.09
taken	asked	1.00	0.28
asked	asked	1.00	0.45
fill	complete	0.86	1.29
out	–	0	0.06
40	–	0	1.39

Table 1: Wu & Palmer word similarity scores for computing text similarity with respect to text 1

Next, we use equation 7 and determine the semantic similarity of the two text segments with respect to text 1 as 0.6702, and with respect to text 2 as 0.7202. Finally, the two figures are combined into a bidirectional measure of similarity, calculated as 0.6952 based on equation 8.

Although there are a few words that occur in both text segments (e.g. *juror*, *questionnaire*), there are also words that are not identical, but closely related, e.g. *courtroom* found similar to *juror*, or *fill* which is related to *complete*. Unlike traditional similarity measures based on lexical matching, our metric takes into account the semantic similarity of these words, resulting in a more precise measure of text similarity.

4 Evaluation

To test the effectiveness of the text semantic similarity metric, we use this measure to automatically identify if two text segments are paraphrases of each other. We use the Microsoft paraphrase corpus (Dolan et al., 2004), consisting of 4,076 training pairs and 1,725 test pairs, and determine the number of correctly identified paraphrase pairs in the corpus using the text semantic similarity measure as the only indicator of paraphrasing. In addition, we also evaluate the measure using the PASCAL corpus (Dagan et al., 2005), consisting of 1,380 test–hypothesis pairs with a directional entailment (580 development pairs and 800 test pairs).

For each of the two data sets, we conduct two evaluations, under two different settings: (1) An unsupervised setting, where the decision on what constitutes a paraphrase (entailment) is made using a constant similarity threshold of 0.5 across all experiments; and (2) A supervised setting, where the optimal threshold and weights associated with various similarity metrics are determined through learning on training data. In this case, we use a voted perceptron algorithm (Freund and Schapire, 1998)[3].

We evaluate the text similarity metric built on top of the various word-to-word metrics introduced in Section 2.1. For comparison, we also compute three baselines: (1) A random baseline created by randomly choosing a true or false value for each text pair; (2) A lexical matching baseline, which only

[3]Classification using this algorithm was determined optimal empirically through experiments.

counts the number of matching words between the two text segments, while still applying the weighting and normalization factors from equation 7; and (3) A vectorial similarity baseline, using a cosine similarity measure as traditionally used in information retrieval, with *tf.idf* term weighting. For comparison, we also evaluated the corpus-based similarity obtained through LSA; however, the results obtained were below the lexical matching baseline and are not reported here.

For paraphrase identification, we use the bidirectional similarity measure, and determine the similarity with respect to each of the two text segments in turn, and then combine them into a bidirectional similarity metric. For entailment identification, since this is a directional relation, we only measure the semantic similarity with respect to the *hypothesis* (the text that is entailed).

We evaluate the results in terms of accuracy, representing the number of correctly identified true or false classifications in the test data set. We also measure precision, recall and F-measure, calculated with respect to the *true* values in each of the test data sets.

Tables 2 and 3 show the results obtained in the unsupervised setting, when a text semantic similarity larger than 0.5 was considered to be an indicator of paraphrasing (entailment). We also evaluate a metric that combines all the similarity measures using a simple average, with results indicated in the $Combined$ row.

The results obtained in the supervised setting are shown in Tables 4 and 5. The optimal combination of similarity metrics and optimal threshold are now determined in a learning process performed on the training set. Under this setting, we also compute an additional baseline, consisting of the most frequent label, as determined from the training data.

Metric	Acc.	Prec.	Rec.	F
Semantic similarity (knowledge-based)				
J & C	0.683	0.724	0.846	0.780
L & C	0.680	0.724	0.838	0.777
Lesk	0.680	0.724	0.838	0.777
Lin	0.679	0.717	0.855	0.780
W & P	0.674	0.722	0.831	0.773
Resnik	0.672	0.725	0.815	0.768
Combined	**0.688**	0.741	0.817	0.777
Baselines				
LexMatch	0.661	0.722	0.798	0.758
Vectorial	0.654	0.716	0.795	0.753
Random	0.513	0.683	0.500	0.578

Table 2: Text semantic similarity for paraphrase identification (unsupervised)

Metric	Acc.	Prec.	Rec.	F
Semantic similarity (knowledge-based)				
J & C	0.573	0.543	0.908	0.680
L & C	0.569	0.543	0.870	0.669
Lesk	0.568	0.542	0.875	0.669
Resnik	0.565	0.541	0.850	0.662
Lin	0.563	0.538	0.878	0.667
W & P	0.558	0.534	0.895	0.669
Combined	**0.583**	0.561	0.755	0.644
Baselines				
LexMatch	0.545	0.530	0.795	0.636
Vectorial	0.528	0.525	0.588	0.555
Random	0.486	0.486	0.493	0.489

Table 3: Text semantic similarity for entailment identification (unsupervised)

5 Discussion and Conclusions

For the task of paraphrase recognition, incorporating semantic information into the text similarity measure increases the likelihood of recognition significantly over the random baseline and over the lexical matching baseline. In the unsupervised setting, the best performance is achieved using a method that combines several similarity metrics into one, for an overall accuracy of 68.8%. When learning is used to find the optimal combination of metrics and optimal threshold, the highest accuracy of 71.5% is obtained by combining the similarity metrics and the lexical matching baseline together.

For the entailment data set, although we do not explicitly check for entailment, the directional similarity computed for textual entailment recognition does improve over the random and lexical matching baselines. Once again, the combination of similarity metrics gives the highest accuracy, measured at 58.3%, with a slight improvement observed in the supervised setting, where the highest accuracy was measured at 58.9%. Both these figures are competitive with the best results achieved during the PASCAL entailment evaluation (Dagan et al., 2005).

Although our method relies on a bag-of-words approach, as it turns out the use of measures of *semantic* similarity improves significantly over the traditional lexical matching metrics[4]. We are nonetheless

[4]The improvement of the combined semantic similarity metric over the simpler lexical matching measure was found to be statistically significant in all experiments, using a paired t-test ($p < 0.001$).

Metric	Acc.	Prec.	Rec.	F
Semantic similarity (knowledge-based)				
Lin	0.702	0.706	0.947	0.809
W & P	0.699	0.705	0.941	0.806
L & C	0.699	0.708	0.931	0.804
J & C	0.699	0.707	0.935	0.805
Lesk	0.695	0.702	0.929	0.800
Resnik	0.692	0.705	0.921	0.799
Combined	**0.715**	0.723	0.925	0.812
Baselines				
LexMatch	0.671	0.693	0.908	0.786
Vectorial	0.665	0.665	1.000	0.799
Most frequent	0.665	0.665	1.000	0.799

Table 4: Text semantic similarity for paraphrase identification (supervised)

Metric	Acc.	Prec.	Rec.	F
Semantic similarity (knowledge-based)				
L & C	0.583	0.573	0.650	0.609
W & P	0.580	0.570	0.648	0.607
Resnik	0.579	0.572	0.628	0.598
Lin	0.574	0.568	0.620	0.593
J & C	0.575	0.566	0.643	0.602
Lesk	0.573	0.566	0.633	0.597
Combined	**0.589**	0.579	0.650	0.612
Baselines				
LexMatch	0.568	0.573	0.530	0.551
Most frequent	0.500	0.500	1.000	0.667
Vectorial	0.479	0.484	0.645	0.553

Table 5: Text semantic similarity for entailment identification (supervised)

aware that a bag-of-words approach ignores many of important relationships in sentence structure, such as dependencies between words, or roles played by the various arguments in the sentence. Future work will consider the investigation of more sophisticated representations of sentence structure, such as first order predicate logic or semantic parse trees, which should allow for the implementation of more effective measures of text semantic similarity.

References

A. Budanitsky and G. Hirst. 2001. Semantic distance in wordnet: An experimental, application-oriented evaluation of five measures. In *Proceedings of the NAACL Workshop on WordNet and Other Lexical Resources*, Pittsburgh, June.

I. Dagan, O. Glickman, and B. Magnini. 2005. The PASCAL recognising textual entailment challenge. In *Proceedings of the PASCAL Workshop*.

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

Y. Freund and R.E. Schapire. 1998. Large margin classification using the perceptron algorithm. In *Proceedings of the 11th Annual Conference on Computational Learning Theory*, pages 209–217, New York, NY. ACM Press.

J. Jiang and D. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of the International Conference on Research in Computational Linguistics*, Taiwan.

T. K. Landauer, P. Foltz, and D. Laham. 1998. Introduction to latent semantic analysis. *Discourse Processes*, 25.

C. Leacock and M. Chodorow. 1998. Combining local context and WordNet sense similiarity for word sense disambiguation. In *WordNet, An Electronic Lexical Database*. The MIT Press.

M.E. Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the SIGDOC Conference 1986*, Toronto, June.

C.Y. Lin and E.H. Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of Human Language Technology Conference (HLT-NAACL 2003)*, Edmonton, Canada, May.

D. Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning*, Madison, WI.

K. Papineni, S. Roukos, T. Ward, and W. Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 2002)*, Philadelphia, PA, July.

S. Patwardhan, S. Banerjee, and T. Pedersen. 2003. Using measures of semantic relatedness for word sense disambiguation. In *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*, Mexico City, February.

P. Resnik. 1995. Using information content to evaluate semantic similarity. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, Montreal, Canada.

J. Rocchio, 1971. *Relevance feedback in information retrieval*. Prentice Hall, Ing. Englewood Cliffs, New Jersey.

G. Salton and M.E. Lesk, 1971. *Computer evaluation of indexing and text processing*, pages 143–180. Prentice Hall, Ing. Englewood Cliffs, New Jersey.

G. Salton, , and A. Bukley. 1997a. Term weighting approaches in automatic text retrieval. In *Readings in Information Retrieval*. Morgan Kaufmann Publishers, San Francisco, CA.

G. Salton, A. Singhal, M. Mitra, and C. Buckley. 1997b. Automatic text structuring and summarization. *Information Processing and Management*, 2(32).

K. Sparck-Jones. 1972. A statistical interpretation of term specificity and its applicatoin in retrieval. *Journal of Documentation*, 28(1):11–21.

E. Voorhees. 1993. Using wordnet to disambiguate word senses for text retrieval. In *Proceedings of the 16th annual international ACM SIGIR conference*, Pittsburgh, PA.

Z. Wu and M. Palmer. 1994. Verb semantics and lexical selection. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, Las Cruces, New Mexico.

J. Xu and W. B. Croft. 1996. Query expansion using local and global document analysis. In *Proceedings of the 19th annual international ACM SIGIR conference*, Zurich, Switzerland.

Training Data Modification for SMT

Considering Groups of Synonymous Sentences

Hideki KASHIOKA
Spoken Language Communication Research Laboratories, ATR
2-2-2 Hikaridai, Keihanna Science City
Kyoto, 619-0288, Japan
hideki.kashioka@atr.jp

Abstract

Generally speaking, statistical machine translation systems would be able to attain better performance with more training sets. Unfortunately, well-organized training sets are rarely available in the real world. Consequently, it is necessary to focus on modifying the training set to obtain high accuracy for an SMT system. If the SMT system trained the translation model, the translation pair would have a low probability when there are many variations for target sentences from a single source sentence. If we decreased the number of variations for the translation pair, we could construct a superior translation model. This paper describes the effects of modification on the training corpus when consideration is given to synonymous sentence groups. We attempt three types of modification: compression of the training set, replacement of source and target sentences with a selected sentence from the synonymous sentence group, and replacement of the sentence on only one side with the selected sentence from the synonymous sentence group. As a result, we achieve improved performance with the replacement of source-side sentences.

1 Introduction

Recently, many researchers have focused their interest on statistical machine translation (SMT) systems, with particular attention given to models and decoding algorithms. The quantity of the training corpus has received less attention, although of course the earlier reports do address the quantity issue. In most cases, the larger the training corpus becomes, the higher accuracy is achieved. Usually, the quantity problem of the training corpus is discussed in relation to the size of the training corpus and system performance; therefore, researchers study line graphs that indicate the relationship between accuracy and training corpus size.

On the other hand, needless to say, a single sentence in the source language can be used to translate several sentences in the target language. Such various possibilities for translation make MT system development and evaluation very difficult. Consequently, here we employ multiple references to evaluate MT systems like BLEU (Papineni et al., 2002) and NIST (Doddington, 2002). Moreover, such variations in translation have a negative effect on training in SMT because when several sentences of input-side language are translated into the exactly equivalent output-side sentences, the probability of correct translation decreases due to the large number of possible pairs of expressions. Therefore, if we can restrain or modify the training corpus, the SMT system might achieve high accuracy.

As an example of modification, different output-side sentences paired with the exactly equivalent input-side sentences are replaced with one target sentence. These sentence replacements are required for synonymous sentence sets. Kashioka (2004) discussed synonymous sets of sentences. Here, we employ a method to group them as a way of modifying the training corpus for use with SMT. This paper focuses on how to control the corpus while giving consideration to synonymous sentence groups.

Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, pages 19–24,
Ann Arbor, June 2005.

2 Target Corpus

In this paper, we use a multilingual parallel corpus called BTEC (Takezawa et al., 2002) for our experiments. BTEC was used in IWSLT (Akiba et al., 2004). This parallel corpus is a collection of Japanese sentences and their translations into English, Korean and Chinese that are often found in phrase books for foreign tourists. These parallel sentences cover a number of situations (e.g., hotel reservations, troubleshooting) for Japanese going abroad, and most of the sentences are rather short. Since the scope of its topics is quite limited, some very similar sentences can be found in the corpus, making BTEC appropriate for modification with compression or replacement of sentences. We use only a part of BTEC for training data in our experiments. The training data we employ contain 152,170 Japanese sentences, with each sentence combined with English and Chinese translations. In Japanese, each sentence has 8.1 words on average, and the maximum sentence length is 150 words. In English, each sentence contains an average of 7.4 words, with a maximum sentence length of 117 words. In Chinese, each sentence has an average of 6.7 words and maximum length of 122 words. Some sentences appear twice or more in the training corpus. In total, our data include 94,268 different Japanese sentences, 87,061 different Chinese sentences, and 91,750 different English sentences. Therefore, there are some sentence pairs that consist of exactly the same sentence in one language but a different sentence in another language, as Fig. 1 shows. This relationship can help in finding the synonymous sentence group.

The test data contain 510 sentences from different training sets in the BTEC. Each source sentence in the test data has 15 target sentences for evaluations. For the evaluation, we do not use any special process for the grouping process. Consequently, our results can be compared with those of other MT systems.

S1 ⇔ T1
S2 ⇔ T1
S1 ⇔ T2
S3 ⇔ T1

Figure 1. Sample sentence pairs

3 Modification Method

When an SMT system learns the translation model, variations in the translated sentences of the pair are critical for determining whether the system obtains a good model. If the same sentence appears twice in the input-side language and these sentences form pairs with two different target sentences in the output-side language, then broadly speaking the translation model defines almost the same probability for these two target sentences.

In our model, the translation system features the ability to generate an output sentence with some variations; however, for the system to generate the most appropriate output sentence, sufficient information is required. Thus, it is difficult to prepare a sufficiently large training corpus.

3.1 Synonymous Sentence Group

Kashioka (2004) reported two steps for making a synonymous sentence group. The first is a concatenation step, and the second is a decomposition step. In this paper, to form a synonymous sentence group, we performed only the concatenation step, which has a very simple idea. When the expression "Exp_A_1" in language A is translated into the expressions "Exp_B_1, Exp_B_2, ..., Exp_B_n" in language B, that set of expressions form one synonymous group. Furthermore, when the sentence "Exp_A_2" in language A is translated into the sentences "Exp_B_1, Exp_$B_{n+1,}$ ..., Exp_B_m" in language B, "Exp_B_1, Exp_$B_{n+1,}$ ..., Exp_B_m ($n < m$)" form one synonymous group. In this situation, "Exp_A_1" and "Exp_A_2" form a synonymous group because both "Exp_A_1" and "Exp_A_2" have a relationship with the translation pairs of "Exp_B_1." Thus, "Exp_$A_{1,}$ Exp_A_2" in language A and "Exp_$B_{1,}$ ..., Exp_B_m" in language B form a synonymous group. If other language information is available, we can extend this synonymous group using information on translation pairs for other languages.

In this paper, we evaluate an EJ/JE system and a CJ/JC system, and our target data include three languages, i.e., Japanese, English, and Chinese. We make synonymous sentence groups in two different environments. One is a group using Japanese and English data, and other is a group that uses Japanese and Chinese data.

The JE group contained 72,808 synonymous sentence groups, and the JC group contained 83,910 synonymous sentence groups as shown in Table 1.

	# of Groups	# of Sent per Group
JE	72,808	2.1
JC	83,910	1.8

Table 1 Statistics used in BTEC data

3.2 Modification

We prepared the three types of modifications for training data.

1. Compress the training corpus based on the synonymous sentence group (Fig. 2).
2. Replace the input and output sides' sentences with the selected sentence, considering the synonymous sentence group (Fig. 3).
3. Replace one side's sentences with a selected sentence, considering the synonymous sentence group (Figs. 4, 5).

We describe these modifications in more detail in the following subsections.

3.2.1 Modification with Compression

Here, a training corpus is constructed with several groups of synonymous sentences. Then, each group keeps only one pair of sentences and the other pairs are removed from each group, thereby decreasing the total number of sentences and narrowing the variation of expressions. Figure 2 shows an example of modification in this way. In the figure, S1, S2, and S3 indicate the input-side sentences while T1 and T2 indicate the output-side sentences. The left-hand side box shows a synonymous sentence group in the original training corpus, where four sentence pairs construct one synonymous sentence group. The right-hand side box shows a part of the modified training corpus. In this case, we keep the S1 and T1 sentences, and this resulting pair comprises a modified training corpus.

The selection of what sentences to keep is an important issue. In our current experiment, we select the most frequent sentence in each side's language from within each group. In Fig. 2, S1 appeared twice, while S2 and S3 appeared only once in the input-side language. As for the output-side language, T1 appeared three times and T2 appeared once. Thus, we keep the pair consisting of S1 and T1. When attempting to separately select the most frequent sentence in each language, we may not find suitable pairs in the original training corpus; however, we can make a new pair with the extracted sentences for the modified training corpus.

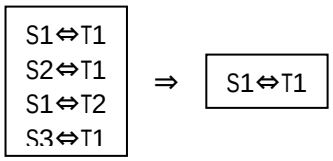

Figure 2. Modification sample for compression

3.2.2 Modification of replacing the sentences of both sides

In the compression stage, the total number of sentences in the modified training corpus is decreased, and it is clear that fewer sentences in the training corpus leads to diminished accuracy. In order to make a comparison between the original training corpus and a modified training corpus with the same number of sentences, we extract one pair of sentences from each group, and each pair appears in the modified training corpus in the same number of sentences. Figure 3 shows an example of this modification. The original training data are the same as in Fig. 2. Then we extract S1 and T1 by the same process from each side with this group, and replacing all of the input-side sentences with S1 in this group. The output side follows the same process. In this case, the modified training corpus consists of four pairs of S1 and T1.

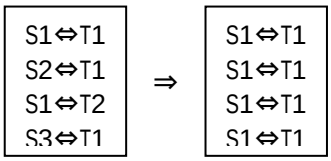

Figure 3. Sample modifications for replacement of both sentences

3.2.3 Modification to replace only one side's sentence

With the previous two modifications, the language variations in both sides decrease. Next, we propose the third modification, which narrows the range of one side's variations.

The sentences of one side are replaced with the selected sentence from that group. The sentence for replacement is selected by following the same process used in the previous modifications. As a result, two modified training corpora are available

as shown in Figs. 4 and 5. Figure 4 illustrates the output side's decreasing variation, while Fig. 5 shows the input side's decreasing variation.

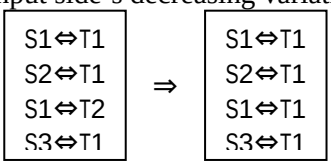

Figure 4. Modification example of replacing the output side's sentence

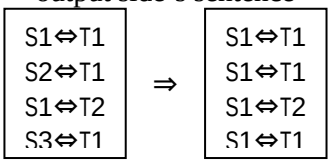

Figure 5. Modification example of replacing the input side's sentence

4 SMT System and Evaluation method

In this section, we describe the SMT systems used in these experiments. The SMT systems' decoder is a graph-based decoder (Ueffing et al., 2002; Zhang et al., 2004). The first pass of the decoder generates a word-graph, a compact representation of alternative translation candidates, using a beam search based on the scores of the lexicon and language models. In the second pass, an *A** search traverses the graph. The edges of the word-graph, or the phrase translation candidates, are generated by the list of word translations obtained from the inverted lexicon model. The phrase translations extracted from the Viterbi alignments of the training corpus also constitute the edges. Similarly, the edges are also created from dynamically extracted phrase translations from the bilingual sentences (Watanabe and Sumita, 2003). The decoder used the IBM Model 4 with a trigram language model and a five-gram part-of-speech language model. Training of the IBM model 4 was implemented by the GIZA++ package (Och and Ney, 2003). All parameters in training and decoding were the same for all experiments. Most systems with this training can be expected to achieve better accuracy when we run the parameter tuning processes. However, our purpose is to compare the difference in results caused by modifying the training corpus.

We performed experiments for JE/EJ and JC/CJ systems and four types of training corpora:

1) Original BTEC corpus;
2) Compressed BTEC corpus (see 3.2.1);
3) Replace both languages (see 3.2.2);
4) Replace one side language (see 3.2.3)
   4-1) replacement on the input side
   4-2) replacement on the output side.

For the evaluation, we use BLEU, NIST, WER, and PER as follows:

BLEU: A weighted geometric mean of the n-gram matches between test and reference sentences multiplied by a brevity penalty that penalizes short translation sentences.

NIST: An arithmetic mean of the n-gram matches between test and reference sentences multiplied by a length factor, which again penalizes short translation sentences.

mWER (Niessen et al., 2000): Multiple reference word-error rate, which computes the edit distance (minimum number of insertions, deletions, and substitutions) between test and reference sentences.

mPER: Multiple reference position-independent word-error rate, which computes the edit distance without considering the word order.

5 Experimental Results

In this section, we show the experimental results for the JE/EJ and JC/CJ systems.

5.1 EJ/JE-system-based JE group

Tables 2 and 3 show the evaluation results for the EJ/JE system.

EJ	BLEU	NIST	mWER	mPER
Original	0.36	3.73	0.55	0.51
Compress	0.47	5.83	0.47	0.44
Replace Both	0.42	5.71	0.50	0.47
Replace J.	0.44	2.98	0.60	0.58
Replace E.	**0.48**	**6.05**	**0.44**	**0.41**

Table 2. Evaluation results for EJ System

JE	BLEU	NIST	mWER	mPER
Original	0.46	3.96	0.52	0.49
Compress	0.53	8.53	**0.42**	**0.38**
Replace Both	0.49	8.10	0.46	0.41
Replace J.	**0.54**	**8.64**	**0.42**	**0.38**
Replace E.	0.51	6.10	0.52	0.49

Table 3. Evaluation results for JE system

Modification of the training data is based on the synonymous sentence group with the JE pair.

The EJ system performed at 0.55 in mWER with the original data set, and the system replacing the Japanese side achieved the best performance of 0.44 in mWER. The system then gained 0.11 in mWER. On the other hand, the system replacing the English side lost 0.05 in mWER. The mPER score also indicates a similar result. For the BLEU and NIST scores, the system replacing the Japanese side also attained the best performance.

The JE system attained a score of 0.52 in mWER with the original data set, while the system with English on the replacement side gave the best performance of 0.42 in mWER, a gain of 0.10. On the other hand, the system with Japanese on the replacement side showed no change in mWER, and the case of compression achieved good performance. The ratios of mWER and mPER are nearly the same for replacing Japanese. Thus, in both directions replacement of the input-side language derives a positive effect for translation modeling.

5.2 CJ/JC system-based JC group

Tables 4 and 5 show the evaluation results for the EJ/JE system based on the group with a JC language pair.

CJ	BLEU	NIST	mWER	mPER
Original	0.51	6.22	0.41	0.38
Compress	0.52	**6.43**	0.43	0.40
Replace both	**0.53**	5.99	***0.40***	***0.37***
Replace J.	0.50	5.98	0.41	0.39
Replace C.	0.51	6.22	0.41	0.38

Table 4. Evaluation results for CJ based on the JC language pair

JC	BLEU	NIST	mWER	mPER
Original	0.56	**8.45**	**0.38**	**0.34**
Compress	0.55	8.22	0.41	0.36
Replace both	0.56	8.32	0.39	0.35
Replace J.	0.56	8.25	0.40	0.36
Replace C.	**0.57**	8.33	**0.38**	0.35

Table 5. Evaluation results for JC based on the JC language pair

The CJ system achieved a score of 0.41 in mWER with the original data set, with the other cases similar to the original; we could not find a large difference among the training corpus modifications. Furthermore, the JC system performed at 0.38 in mWER with the original data, although the other cases' results were not as good. These results seem unusual considering the EJ/JE system, indicating that they derive from the features of the Chinese part of the BTEC corpus.

6 Discussion

Our EJ/JE experiment indicated that the system with input-side language replacement achieved better performance than that with output-side language replacement. This is a reasonable result because the system learns the translation model with fewer variations for input-side language.

In the experiment on the CJ/JC system based on the JC group, we did not provide an outline of the EJ/JE system due to the features of BTEC. Initially, BTEC data were created from pairs of Japanese and English sentences in the travel domain. Japanese-English translation pairs have variation as shown in Fig. 1. However, when Chinese data was translated, BTEC was controlled so that the same Japanese sentence has only one Chinese sentence. Accordingly, there is no variation in Chinese sentences for the pair with the same Japanese sentence. Therefore, the original training data would be similar to the situation of replacing Chinese. Moreover, replacing the Japanese data was almost to the same as replacing both sets of data. Considering this feature of the training corpus, i.e. the results for the CJ/JC system based on the group with JC language pairs, there are few differences between keeping the original data and replacing the Chinese data, or between replacing both side's data and replacing only the Japanese data. These results demonstrate the correctness of the hypothesis that reducing the input side's language variation makes learning models more effective.

Currently, our modifications only roughly process sentence pairs, though the process of making groups is very simple. Sometimes a group may include sentences or words that have slightly different meanings, such as. *fukuro* (bag), *kamibukuro* (paper bag), *shoppingu baggu* (shopping bag), *tesagebukuro* (tote bag), and *biniiru bukuro* (plastic bag). In this case if we select *tesagebukuro* from the Japanese side and "paper bag" from the English side, we have an incorrect word pair in the translation model. To handle such a problem, we would have to arrange a method to select the sen-

tences from a group. This problem is discussed in Imamura et al. (2003). As one solution to this problem, we borrowed the measures of literalness, context freedom, and word translation stability in the sentence-selection process.

In some cases, the group includes sentences with different meanings, and this problem was mentioned in Kashioka (2004). In an attempt to solve the problem, he performed a secondary decomposition step to produce a synonymous group. However, in the current training corpus, each synonymous group before the decomposition step is small, so there would not be enough difference for modifications after the decomposition step.

The replacement of a sentence could be called paraphrasing. Shimohata et al. (2004) reported a paraphrasing effect in MT systems, where if each group would have the same meaning, the variation in the phrases that appeared in the other groups would reduce the probability. Therefore, considering our results in light of their discussion, if the training corpus could be modified with the module for paraphrasing in order to control phrases, we could achieve better performance.

7 Conclusion

This paper described the modification of a training set based on a synonymous sentence group for a statistical machine translation system in order to attain better performance. In an EJ/JE system, we confirmed a positive effect by replacing the input-side language. Because the Chinese data was specific in our modification, we observed an inconclusive result for the modification in the CJ/JC system based on the synonymous sentence group with a JC language pair. However, there was still some effect on the characteristics of the training corpus. In this paper, the modifications of the training set are based on the synonymous sentence group, and we replace the sentence with rough processing. If we paraphrased the training set and controlled the phrase pair, we could achieve better performance with the same training set.

Acknowledgements

This research was supported in part by the National Institute of Information and Communications Technology.

References

Yasuhiro AKIBA, Marcello FEDERICO, Noriko KANDO, Hiromi NAKAIWA, Michael PAUL, and Jun'ichi TSUJII, 2004. *Overview of the IWSLT04 Evaluation Campaign,* In *Proc. of IWSLT04*, 1 – 12.

George Doddington. 2002. *Automatic evaluation of machine translation quality using n-gram co-occurrence statistics*. In *Proceedings of the HLT Conference*, San Diego, California.

Kenji Imamura, Eiichiro Sumita, and Yuji Matsumoto, 2003. *Automatic Construction of Machine Translation Knowledge Using Translation Literalness*, in *Proc. of EACL* 2003, 155 – 162.

Hideki Kashioka, 2004. *Grouping Synonymous Sentences from a Parallel Corpus.* In *Proc. of LREC 2004*, 391 - 394.

Sonja Niessen, Franz J. Och, Gregor Leusch, and Hermann Ney. 2000. *An evaluation tool for machine translation: Fast evaluation for machine translation research.* In *Proc.of LREC 2000*, 39 – 45.

Franz Josef Och and Hermann Ney. 2003. *A systematic comparison of various statistical alignment models. Computational Linguistics*, 29(1):19 - 51.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. *Bleu: a method for automatic evaluation of machine translation.* In *Proc. of ACL 2002*, 311–318.

Mitsuo Shimohata, Eiichiro Sumita, and Yuji Matsumoto, 2004. *Building a Paraphrase Corpus for Speech Translation.* In *Proc. of LREC 2004*, 1407 - 1410.

Toshiyuki Takezawa, Eiichiro Sumita, Fumiaki Sugaya, Hirofumi Yamamoto, and Seiichi Yamamoto. 2002. *Toward a broad-coverage bilingual corpus for speech translation of travel conversations in the real world*, In *Proc. of LREC 2002*, 147–152.

Nicola Ueffing, Franz Josef Och, and Hermann Ney. 2002. *Generation of word graphs in statistical machine translation.* In *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP02)*, 156 – 163.

Taro Watanabe and Eiichiro Sumita. 2003. *Example-based decoding for statistical machine translation.* In *Machine Translation Summit IX*, 410 – 417.

Ruiqiang Zhang, Genichiro Kikui, Hirofumi Yamamoto, Frank Soong, Taro Watanabe and Wai Kit Lo, 2004. A Unified Approach in Speech-to-Speech Translation: Integrating Features of Speech recognition and Machine Translation, In Proc. of COLING 2004, 1168 - 1174.

Recognizing Paraphrases and Textual Entailment using Inversion Transduction Grammars

Dekai Wu[1]

Human Language Technology Center
HKUST
Department of Computer Science
University of Science and Technology, Clear Water Bay, Hong Kong
dekai@cs.ust.hk

Abstract

We present first results using paraphrase as well as textual entailment data to test the language universal constraint posited by Wu's (1995, 1997) Inversion Transduction Grammar (ITG) hypothesis. In machine translation and alignment, the ITG Hypothesis provides a strong inductive bias, and has been shown empirically across numerous language pairs and corpora to yield both efficiency and accuracy gains for various language acquisition tasks. Monolingual paraphrase and textual entailment recognition datasets, however, potentially facilitate closer tests of certain aspects of the hypothesis than bilingual parallel corpora, which simultaneously exhibit many irrelevant dimensions of cross-lingual variation. We investigate this using simple generic Bracketing ITGs containing no language-specific linguistic knowledge. Experimental results on the MSR Paraphrase Corpus show that, even in the absence of any thesaurus to accommodate lexical variation between the paraphrases, an uninterpolated average precision of at least 76% is obtainable from the Bracketing ITG's structure matching bias alone. This is consistent with experimental results on the Pascal Recognising Textual Entailment Challenge Corpus, which show surpisingly strong results for a number of the task subsets.

1 Introduction

The *Inversion Transduction Grammar* or *ITG* formalism, which historically was developed in the context of translation and alignment, hypothesizes strong expressiveness restrictions that constrain paraphrases to vary word order only in certain allowable nested permutations of arguments (Wu, 1997). The ITG Hypothesis has been more extensively studied across different languages, but newly available paraphrase datasets provide intriguing opportunities for meaningful analysis of the ITG Hypothesis in a monolingual setting.

The strong inductive bias imposed by the ITG Hypothesis has been repeatedly shown empirically to yield both efficiency and accuracy gains for numerous language acquisition tasks, across a variety of language pairs and tasks. For example, Zens and Ney (2003) show that ITG constraints yield significantly better alignment coverage than the constraints used in IBM statistical machine translation models on both German-English (Verbmobil corpus) and French-English (Canadian Hansards corpus). Zhang and Gildea (2004) find that unsupervised alignment using Bracketing ITGs produces significantly lower Chinese-English alignment error rates than a syntactically supervised tree-to-string model (Yamada and Knight, 2001). With regard to translation rather than alignment accuracy, Zens *et al.* (2004) show that decoding under ITG constraints yields significantly lower word error rates and BLEU scores than the IBM constraints.

We are conducting a series of investigations motivated by the following observation: the empirically demonstrated suitability of ITG paraphrasing constraints across languages should hold, if anything, even more strongly in the monolingual case. The monolingual case allows in some sense closer testing of various implications of the ITG hypothesis, without irrelevant dimensions of variation arising from other cross-lingual phenomena.

Asymmetric textual entailment recognition (RTE) datasets, in particular the Pascal Recognising Textual Entailment Challenge Corpus (Dagan *et al.*, 2005), provide testbeds that abstract over many tasks, including information retrieval, comparable documents, reading comprehension, question answering, information extraction, machine translation, and paraphrase acquisition.

At the same time, the emergence of paraphrasing datasets presents an opportunity for complementary experiments on the task of recognizing symmetric bidirectional entailment rather than asymmetric directional entailment. In particular, for this study we employ the MSR Paraphrase Corpus (Quirk *et al.*, 2004).

[1]The author would like to thank the Hong Kong Research Grants Council (RGC) for supporting this research in part through grants RGC6083/99E, RGC6256/00E, and DAG03/04.EG09, and Marine Carpuat and Yihai Shen for invaluable assistance in preparing the datasets and stoplist.

Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, pages 25–30,
Ann Arbor, June 2005.

2 Inversion Transduction Grammars

Formally, ITGs can be defined as the restricted subset of syntax-directed transduction grammars or SDTGs Lewis and Stearns (1968) where all of the rules are either of *straight* or *inverted* orientation. Ordinary SDTGs allow any permutation of the symbols on the right-hand side to be specified when translating from the input language to the output language. In contrast, ITGs only allow two out of the possible permutations. If a rule is straight, the order of its right-hand symbols must be the same for both language. On the other hand, if a rule is inverted, then the order is left-to-right for the input language and right-to-left for the output language. Since inversion is permitted at any level of rule expansion, a derivation may intermix productions of either orientation within the parse tree. The ability to compose multiple levels of straight and inverted constituents gives ITGs much greater expressiveness than might seem at first blush.

A simple example may be useful to fix ideas. Consider the following pair of parse trees for sentence translations:

> [[[The Authority]$_{\text{NP}}$ [will [[be accountable]$_{\text{VV}}$ [to [the [[Financial Secretary]$_{\text{NN}}$]$_{\text{NNN}}$]$_{\text{NP}}$]$_{\text{PP}}$]$_{\text{VP}}$]$_{\text{VP}}$]$_{\text{SP}}$.]$_{\text{S}}$
>
> [[[管理局]$_{\text{NP}}$ [将会 [[向 [[[财政 司]$_{\text{NN}}$]$_{\text{NNN}}$]$_{\text{NP}}$]$_{\text{PP}}$ [负责]$_{\text{VV}}$]$_{\text{VP}}$]$_{\text{VP}}$]$_{\text{SP}}$ 。]$_{\text{S}}$

Even though the order of constituents under the inner VP is inverted between the languages, an ITG can capture the common structure of the two sentences. This is compactly shown by writing the parse tree together for both sentences with the aid of an $\langle\rangle$ angle bracket notation marking parse tree nodes that instantiate rules of inverted orientation:

> [[[The/ε Authority/管 理 局]$_{\text{NP}}$ [will/将 会 ⟨[be/ε accountable/负 责]$_{\text{VV}}$ [to/向 [the/ε [[Financial/财政Secretary/司]$_{\text{NN}}$]$_{\text{NNN}}$]$_{\text{NP}}$]$_{\text{PP}}$ ⟩$_{\text{VP}}$]$_{\text{VP}}$]$_{\text{SP}}$./。]$_{\text{S}}$

In a weighted or stochastic ITG (SITG), a weight or a probability is associated with each rewrite rule. Following the standard convention, we use a and b to denote probabilities for syntactic and lexical rules, respectively. For example, the probability of the rule NN $\stackrel{0.4}{\rightarrow}$ [A N] is $a_{\text{NN}\rightarrow\text{[A N]}} = 0.4$. The probability of a lexical rule A $\stackrel{0.001}{\rightarrow}$ x/y is $b_{\text{A}}(x, y) = 0.001$. Let W_1, W_2 be the vocabulary sizes of the two languages, and $\mathcal{N} = \{A_1, \ldots, A_N\}$ be the set of nonterminals with indices $1, \ldots, N$.

Wu (1997) also showed that ITGs can be equivalently be defined in two other ways. First, ITGs can be defined as the restricted subset of SDTGs where all rules are of rank 2. Second, ITGs can also be defined as the restricted subset of SDTGs where all rules are of rank 3.

Polynomial-time algorithms are possible for various tasks including translation using ITGs, as well as bilingual parsing or *biparsing*, where the task is to build the highest-scored parse tree given an input bi-sentence.

For present purposes we can employ the special case of Bracketing ITGs, where the grammar employs only one single, undistinguished "dummy" nonterminal category for any non-lexical rule. Designating this category A, a Bracketing ITG has the following form (where, as usual, lexical transductions of the form $A \rightarrow e/f$ may possibly be singletons of the form $A \rightarrow e/\epsilon$ or $A \rightarrow \epsilon/f$).

$$\begin{aligned} A &\rightarrow [AA] \\ A &\rightarrow \langle AA \rangle \\ A &\rightarrow \epsilon, \epsilon \\ A &\rightarrow e_1/f_1 \\ &\ldots \\ A &\rightarrow e_{\text{i}}/f_{\text{j}} \end{aligned}$$

The simplest class of ITGs, *Bracketing ITGs*, are particularly interesting in applications like paraphrasing, because they impose ITG constraints in language-independent fashion, and in the simplest case do not require any language-specific linguistic grammar or training. In Bracketing ITGs, the grammar uses only a single, undifferentiated non-terminal (Wu, 1995). The key modeling property of Bracketing ITGs that is most relevant to paraphrase recognition is that they assign strong preference to candidate paraphrase pairs in which nested constituent subtrees can be recursively aligned with a minimum of constituent boundary violations. Unlike language-specific linguistic approaches, however, the shape of the trees are driven in unsupervised fashion by the data. One way to view this is that the trees are hidden explanatory variables. This not only provides significantly higher robustness than more highly constrained manually constructed grammars, but also makes the model widely applicable across languages in economical fashion without a large investment in manually constructed resources.

Moreover, for reasons discussed by Wu (1997), ITGs possess an interesting intrinsic combinatorial property of permitting roughly up to four arguments of any frame to be transposed freely, but not more. This matches suprisingly closely the preponderance of linguistic verb frame theories from diverse linguistic traditions that all allow up to four arguments per frame. Again, this property emerges naturally from ITGs in language-independent fashion, without any hardcoded language-specific knowledge. This further suggests that ITGs should do well at picking out paraphrase pairs where the order of up to four arguments per frame may vary freely between the two strings. Conversely, ITGs should do well at rejecting pairs where (1) too many words in one sentence

find no correspondence in the other, (2) frames do not nest in similar ways in the candidate sentence pair, or (3) too many arguments must be transposed to achieve an alignment—all of which would suggest that the sentences probably express different ideas.

As an illustrative example, in common similarity models, the following pair of sentences (found in actual data arising in our experiments below) would receive an inappropriately high score, because of the high lexical similarity between the two sentences:

> Chinese president Jiang Zemin arrived in Japan today for a landmark state visit .
>
> 江泽民 将 是 到 日本 做 国事访问 的 首位 中国 国家 主席 .
> *(Jiang Zemin will be the first Chinese national president to pay a state vist to Japan.)*

However, the ITG based model is sensitive enough to the differences in the constituent structure (reflecting underlying differences in the predicate argument structure) so that our experiments show that it assigns a low score. On the other hand, the experiments also show that it successfully assigns a high score to other candidate bisentences representing a true Chinese translation of the same English sentence, as well as a true English translation of the same Chinese sentence.

We investigate a model for the paraphrase recognition problem that employ simple generic Bracketing ITGs. The experimental results show that, even in the absence of any thesaurus to accommodate lexical variation between the two strings, the Bracketing ITG's structure matching bias alone produces a significant improvement in average precision.

3 Scoring Method

All words of the vocabulary are included among the lexical transductions, allowing exact word matches between the two strings of any candidate paraphrase pair.

Each candidate pair of the test set was scored via the ITG biparsing algorithm, which employs a dynamic programming approach as follows.Let the input English sentence be $\mathbf{e}_1, \ldots, \mathbf{e}_T$ and the corresponding input Chinese sentence be $\mathbf{c}_1, \ldots, \mathbf{c}_V$. As an abbreviation we write $\mathbf{e}_{s..t}$ for the sequence of words $\mathbf{e}_{s+1}, \mathbf{e}_{s+2}, \ldots, \mathbf{e}_t$, and similarly for $\mathbf{c}_{u..v}$; also, $\mathbf{e}_{s..s} = \epsilon$ is the empty string. It is convenient to use a 4-tuple of the form $q = (s, t, u, v)$ to identify each node of the parse tree, where the substrings $\mathbf{e}_{s..t}$ and $\mathbf{c}_{u..v}$ both derive from the node q. Denote the nonterminal label on q by $\ell(q)$. Then for any node $q = (s, t, u, v)$, define

$$\delta_q(i) = \delta_{stuv}(i) = \max_{\text{subtrees of} q} P[\text{subtree of} q, \ell(q) = i, i \stackrel{*}{\Rightarrow} \mathbf{e}_{s..t}/\mathbf{c}_{u..v}]$$

as the maximum probability of any derivation from i that successfully parses both $\mathbf{e}_{s..t}$ and $\mathbf{c}_{u..v}$. Then the best parse of the sentence pair has probability $\delta_{0,T,0,V}(\mathrm{S})$.

The algorithm computes $\delta_{0,T,0,V}(\mathrm{S})$ using the following recurrences. Note that we generalize argmax to the case where maximization ranges over multiple indices, by making it vector-valued. Also note that $[\,]$ and $\langle\rangle$ are simply constants, written mnemonically. The condition $(S-s)(t-S)+(U-u)(v-U) \neq 0$ is a way to specify that the substring in one but not both languages may be split into an empty string ϵ and the substring itself; this ensures that the recursion terminates, but permits words that have no match in the other language to map to an ϵ instead.

1. **Initialization**

$$\begin{aligned}
\delta_{t-1,t,v-1,v}(i) &= b_i(\mathbf{e}_t/\mathbf{c}_v), & 1 \le t \le T \\
& & 1 \le v \le V \\
\delta_{t-1,t,v,v}(i) &= b_i(\mathbf{e}_t/\epsilon), & 1 \le t \le T \\
& & 0 \le v \le V \\
\delta_{t,t,v-1,v}(i) &= b_i(\epsilon/\mathbf{c}_v), & 0 \le t \le T \\
& & 1 \le v \le V
\end{aligned}$$

2. **Recursion** For all i, s, t, u, v such that $\left\{ \begin{array}{c} 1 \le i \le N \\ 0 \le s < t \le T \\ 0 \le u < v \le V \\ t-s+v-u > 2 \end{array} \right.$

$$\begin{aligned}
\delta_{stuv}(i) &= \max[\delta^{[]}_{stuv}(i), \delta^{\langle\rangle}_{stuv}(i)] \\
\theta_{stuv}(i) &= \left\{ \begin{array}{ll} [\,] & \text{if } \delta^{[]}_{stuv}(i) \ge \delta^{\langle\rangle}_{stuv}(i) \\ \langle\rangle & \text{otherwise} \end{array} \right.
\end{aligned}$$

where

$$\delta^{[]}_{stuv}(i) = \max_{\substack{1 \le j \le N \\ 1 \le k \le N \\ s \le S \le t \\ u \le U \le v \\ (S-s)(t-S)+(U-u)(v-U) \neq 0}} a_{i \to [jk]}\, \delta_{sSuU}(j)\, \delta_{StUv}(k)$$

$$\begin{bmatrix} \iota^{[]}_{stuv}(i) \\ \kappa^{[]}_{stuv}(i) \\ \sigma^{[]}_{stuv}(i) \\ \upsilon^{[]}_{stuv}(i) \end{bmatrix} = \operatorname*{argmax}_{\substack{1 \le j \le N \\ 1 \le k \le N \\ s \le S \le t \\ u \le U \le v \\ (S-s)(t-S)+(U-u)(v-U) \neq 0}} a_{i \to [jk]}\, \delta_{sSuU}(j)\, \delta_{StUv}(k)$$

$$\delta^{\langle\rangle}_{stuv}(i) = \max_{\substack{1 \le j \le N \\ 1 \le k \le N \\ s \le S \le t \\ u \le U \le v \\ (S-s)(t-S)+(U-u)(v-U) \neq 0}} a_{i \to \langle jk \rangle}\, \delta_{sSUv}(j)\, \delta_{StuU}(k)$$

$$\begin{bmatrix} \iota^{\langle\rangle}_{stuv}(i) \\ \kappa^{\langle\rangle}_{stuv}(i) \\ \sigma^{\langle\rangle}_{stuv}(i) \\ \upsilon^{\langle\rangle}_{stuv}(i) \end{bmatrix} = \operatorname*{argmax}_{\substack{1 \le j \le N \\ 1 \le k \le N \\ s \le S \le t \\ u \le U \le v \\ (S-s)(t-S)+(U-u)(v-U) \neq 0}} a_{i \to \langle jk \rangle}\, \delta_{sSUv}(j)\, \delta_{StuU}(k)$$

3. **Reconstruction** Initialize by setting the root of the parse tree to $q_1 = (0, T, 0, V)$ and its nonterminal label to $\ell(q_1) = \text{S}$. The remaining descendants in the optimal parse tree are then given recursively for any $q = (s, t, u, v)$ by:

$$\textsc{Left}(q) = \begin{cases} \text{NIL} & \text{if } t-s+v-u \leq 2 \\ (s, \sigma_q^{[]}(\ell(q)), u, v_q^{[]}(\ell(q))) & \text{if } \theta_q(\ell(q)) = [\,] \\ (s, \sigma_q^{\langle\rangle}(\ell(q)), v_q^{\langle\rangle}(\ell(q))) & \text{if } \theta_q(\ell(q)) = \langle\rangle \end{cases}$$

$$\textsc{Right}(q) = \begin{cases} \text{NIL} & \text{if } t-s+v-u \leq 2 \\ (\sigma_q^{[]}(\ell(q)), t, v_q^{[]}(\ell(q)), v) & \text{if } \theta_q(\ell(q)) = [\,] \\ (\sigma_q^{\langle\rangle}(\ell(q)), t, u, v_q^{\langle\rangle}(\ell(q))) & \text{if } \theta_q(\ell(q)) = \langle\rangle \end{cases}$$

$$\ell(\textsc{Left}(q)) = \iota_q^{\theta_q(\ell(q))}(\ell(q))$$

$$\ell(\textsc{Right}(q)) = \kappa_q^{\theta_q(\ell(q))}(\ell(q))$$

As mentioned earlier, biparsing for ITGs can be accomplished efficiently in polynomial time, rather than the exponential time required for classical SDTGs. The result in Wu (1997) implies that for the special case of Bracketing ITGs, the time complexity of the algorithm is $\Theta\left(T^3V^3\right)$ where T and V are the lengths of the two sentences. This is a factor of V^3 more than monolingual chart parsing, but has turned out to remain quite practical for corpus analysis, where parsing need not be real-time.

The ITG scoring model can also be seen as a variant of the approach described by Leusch *et al.* (2003), which allows us to forego training to estimate true probabilities; instead, rules are simply given unit weights. The ITG scores can be interpreted as a generalization of classical Levenshtein string edit distance, where inverted block transpositions are also allowed. Even without probability estimation, Leusch *et al.* found excellent correlation with human judgment of similarity between translated paraphrases.

4 Experimental Results—Paraphrase Recognition

Our objective here was to isolate the effect of the ITG constraint bias. No training was performed with the available development sets. Rather, the aim was to establish foundational baseline results, to see in this first round of paraphrase recognition experiments what results could be obtained with the simplest versions of the ITG models.

The MSR Paraphrase Corpus test set consists of 1725 candidate paraphrase string pairs, each annotated for semantic equivalence by two or three human collectors. Within the test set, 66.5% of the examples were annotated as being semantically equivalent. The corpus was originally generated via a combination of automatic filtering methods, making it difficult to make specific claims about distributional neutrality, due to the arbitrary nature of the example selection process.

The ITG scoring model produced an uninterpolated average precision (also known as confidence weighted score) of 76.1%. This represents an improvement of roughly 10% over the random baseline. Note that this improvement can be achieved with no thesaurus or lexical similarity model, and no parameter training.

5 Experimental Results—Textual Entailment Recognition

The experimental procedure for the monolingual textual entailment recognition task is the same as for paraphrase recognition, except that one string serves as the Text and the other serves as the Hypothesis.

Results on the textual entailment recognition task are consistent with the above paraphrase recognition results. For the PASCAL RTE challenge datasets, across all subsets overall, the model produced a confidence-weighted score of 54.97% (better than chance at the 0.05 level). All examples were labeled, so precision, recall, and f-score are equivalent; the accuracy was 51.25%.

For the RTE task we also investigated a second variant of the model, in which a list of 172 words from a stoplist was excluded from the lexical transductions. The motivation for this model was to discount the effect of words such as "the" or "of" since, more often than not, they could be irrelevant to the RTE task.

Surprisingly, the stoplisted model produced worse results. The overall confidence-weighted score was 53.61%, and the accuracy was 50.50%. We discuss the reasons below in the context of specific subsets.

As one might expect, the Bracketing ITG models performed better on the subsets more closely approximating the tasks for which Bracketing ITGs were designed: comparable documents (CD), paraphrasing (PP), and information extraction (IE). We will discuss some important caveats on the machine translation (MT) and reading comprehension (RC) subsets. The subsets least close to the Bracketing ITG models are information retrieval (IR) and question answering (QA).

5.1 Comparable Documents (CD)

The CD task definition can essentially be characterized as recognition of noisy word-aligned sentence pairs. Among all subsets, CD is perhaps closest to the noisy word alignment task for which Bracketing ITGs were originally developed, and indeed produced the best results for both of the Bracketing ITG models. The basic model produced a confidence-weighted score of 79.88% (accuracy 71.33%), while the stoplisted model produced an essentially unchanged confidence-weighted score of 79.83%

(accuracy 70.00%).

The results on the RTE Challenge datasets closely reflect the larger-scale findings of Wu and Fung (2005), who demonstrate that an ITG based model yields far more accurate extraction of parallel sentences from quasi-comparable non-parallel corpora than previous state-of-the-art methods. Wu and Fung's results also use the evaluation metric of uninterpolated average precision (i.e., confidence-weighted score).

Note also that we believe the results here are artificially lowered by the absence of any thesaurus, and that significantly further improvements would be seen with the addition of a suitable thesaurus, for reasons discussed below under the MT subsection.

5.2 Paraphrase Acquisition (PP)

The PP task is also close to the task for which Bracketing ITGs were originally developed. For the PP task, the basic model produced a confidence-weighted score of 57.26% (accuracy 56.00%), while the stoplisted model produced a lower confidence-weighted score of 51.65% (accuracy 52.00%). Unlike the CD task, the greater importance of function words in determining equivalent meaning between paraphrases appears to cause the degradation in the stoplisted model.

The effect of the absence of a thesaurus is much stronger for the PP task as opposed to the CD task. Inspection of the datasets reveals much more lexical variation between paraphrases, and shows that cases where lexis does not vary are generally handled accurately by the Bracketing ITG models. The MT subsection below discusses why a thesaurus should produce significant improvement.

5.3 Information Extraction (IE)

The IE task presents a slight issue of misfit for the Bracketing ITG models, but yielded good results anyhow. The basic Bracketing ITG model attempts to align all words/collocations between the two strings. However, for the IE task in general, only a substring of the Text should be aligned to the Hypothesis, and the rest should be disregarded as "noise". We approximated this by allowing words to be discarded from the Text at little cost, by using parameters that impose only a small penalty on null-aligned words from the Text. (As a reasonable first approximation, this characterization of the IE task ignores the possibility of modals, negation, quotation, and the like in the Text.)

Despite the slight modeling misfit, the Bracketing ITG models produced good results for the IE subset. The basic model produced a confidence-weighted score of 59.92% (accuracy 55.00%), while the stoplisted model produced a lower confidence-weighted score of 53.63% (accuracy 51.67%). Again, the lower score of the stoplisted model appears to arise from the greater importance of function words in ensuring correct information extraction, as compared with the CD task.

5.4 Machine Translation (MT)

One exception to expectations is the machine translation subset, a task for which Bracketing ITGs were developed. The basic model produced a confidence-weighted score of 34.30% (accuracy 40.00%), while the stoplisted model produced a comparable confidence-weighted score of 35.96% (accuracy 39.17%).

However, the performance here on the machine translation subset cannot be directly interpreted, for two reasons.

First, the task as defined in the RTE Challenge datasets is not actually crosslingual machine translation, but rather evaluation of monolingual comparability between an automatic translation and a gold standard human translation. This is in fact closer to the problem of defining a good MT evaluation metric, rather than MT itself. Leusch *et al.* (2003 and personal communication) found that Bracketing ITGs as an MT evaluation metric show excellent correlation with human judgments.

Second, no translation lexicon or equivalent was used in our model. Normally in translation models, including ITG models, the translation lexicon accommodates lexical ambiguity, by providing multiple possible lexical choices for each word or collocation being translated. Here, there is no second language, so some substitute mechanism to accommodate lexical ambiguity would be needed.

The most obvious substitute for a translation lexicon would be a monolingual thesaurus. This would allow matching synonomous words or collocations between the Text and the Hypothesis. Our original thought was to incorporate such a thesaurus in collaboration with teams focusing on creating suitable thesauri, but time limitations prevented completion of these experiments. Based on our own prior experiments and also on Leusch *et al.*'s experiences, we believe this would bring performance on the MT subset to excellent levels as well.

5.5 Reading Comprehension (RC)

The reading comprehension task is similar to the information extraction task. As such, the Bracketing ITG model could be expected to perform well for the RC subset. However, the basic model produced a confidence-weighted score of just 49.37% (accuracy 47.14%), and the stoplisted model produced a comparable confidence-weighted score of 47.11% (accuracy 45.00%).

The primary reason for the performance gap between the RC and IE domains appears to be that RC is less news-oriented, so there is less emphasis on exact lexical choices such as named entities. This puts more weight on

the importance of a good thesaurus to recognize lexical variation. For this reason, we believe the addition of a thesaurus would bring performance improvements similar to the case of MT.

5.6 Information Retrieval (IR)

The IR task diverges significantly from the tasks for which Bracketing ITGs were developed. The basic model produced a confidence-weighted score of 43.14% (accuracy 46.67%), while the stoplisted model produced a comparable confidence-weighted score of 44.81% (accuracy 47.78%).

Bracketing ITGs seek structurally parallelizable substrings, where there is reason to expect some degree of generalization between the frames (heads and arguments) of the two substrings from a lexical semantics standpoint. In contrast, the IR task relies on unordered keywords, so the effect of argument-head binding cannot be expected to be strong.

5.7 Question Answering (QA)

The QA task is extremely free in the sense that questions can differ significantly from the answers in both syntactic structure and lexis, and can also require a significant degree of indirect complex inference using real-world knowledge. The basic model produced a confidence-weighted score of 33.20% (accuracy 40.77%), while the stoplisted model produced a significantly better confidence-weighted score of 38.26% (accuracy 44.62%).

Aside from adding a thesaurus, to properly model the QA task, at the very least the Bracketing ITG models would need to be augmented with somewhat more linguistic rules that include a proper model for *wh-* words in the Hypothesis, which otherwise cannot be aligned to the Text. In the Bracketing ITG models, the stoplist appears to help by normalizing out the effect of the *wh-* words.

6 Conclusion

The most serious omission in our experiments with Bracketing ITG models was the absence of any thesaurus model, allowing zero lexical variation between the two strings of a candidate paraphrase pair (or Text and Hypothesis, in the case of textual entailment recognition). This forced the models to rely entirely on the Bracketing ITG's inherent tendency to optimize structural match between hypothesized nested argument-head substructures. What we find highly interesting is the perhaps surprisingly large effect obtainable from this structure matching bias alone, which already produces good results on paraphrasing as well as a number of the RTE subsets.

We plan to remedy the absence of a thesaurus as the obvious next step. This can be expected to raise performance significantly on all subsets.

Wu and Fung (2005) also discuss how to obtain any desired tradeoff between precision and recall. This would be another interesting direction to pursue in the context of recognizing paraphrases or textual entailment.

Finally, using the development sets to train the parameters of the Bracketing ITG model would improve performance. It would only be feasible to tune a few basic parameters, however, given the small size of the development sets.

References

Ido Dagan, Oren Glickman, and Bernardo Magnini. The pascal recognising textual entailment challenge. In *PASCAL Proceedings of the First Challenge Workshop—Recognizing Textual Entailment*, pages 1–8, Southampton, UK, April 2005.

Gregor Leusch, Nicola Ueffing, and Hermann Ney. A novel string-to-string distance measure with applications to machine translation evaluation. In *Machine Translation Summit*, New Orleans, 2003.

P. M. Lewis and R. E. Stearns. Syntax-directed transduction. *Journal of the Association for Computing Machinery*, 15:465–488, 1968.

C. Quirk, C. Brockett, and W. B. Dolan. Monolingual machine translation for paraphrase generation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP-2004)*, Barcelona, June 2004. SIGDAT, Association for Computational Linguistics.

Dekai Wu and Pascale Fung. Inversion Transduction Grammar constraints for mining parallel sentences from quasi-comparable corpora. In *Forthcoming*, 2005.

Dekai Wu. An algorithm for simultaneously bracketing parallel texts by aligning words. In *33rd Annual Meeting of the Association for Computational Linguistics Conference (ACL-95)*, Cambridge, MA, Jun 1995. Association for Computational Linguistics.

Dekai Wu. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3), Sep 1997.

Kenji Yamada and Kevin Knight. A syntax-based statistical translation model. In *39th Annual Meeting of the Association for Computational Linguistics Conference (ACL-01)*, Toulouse, France, 2001. Association for Computational Linguistics.

Richard Zens and Hermann Ney. A comparative study on reordering constraints in statistical machine translation. pages 192–202, Hong Kong, August 2003.

Richard Zens, Hermann Ney, Taro Watanabe, and Eiichiro Sumita. Reordering constraints for phrase-based statistical machine translation. In *Proceedings of COLING*, Geneva, August 2004.

Hao Zhang and Daniel Gildea. Syntax-based alignment: Supervised or unsupervised? In *Proceedings of COLING*, Geneva, August 2004.

Local Textual Inference: can it be defined or circumscribed?

Annie Zaenen
Palo Alto Research Center
3333, Coyote Hill Road
Palo Alto, CA 94304
zaenen@parc.com

Lauri Karttunen
Palo Alto Research Center
3333, Coyote Hill Road
Palo Alto, CA 94304
karttunen@parc.com

Richard Crouch
Palo Alto Research Center
3333, Coyote Hill Road
Palo Alto, CA 94304
crouch@parc.com

Abstract

This paper argues that local textual inferences come in three well-defined varieties (entailments, conventional implicatures/presuppositions, and conversational implicatures) and one less clearly defined one, generally available world knowledge. Based on this taxonomy, it discusses some of the examples in the PASCAL text suite and shows that these examples do not fall into any of them. It proposes to enlarge the test suite with examples that are more directly related to the inference patterns discussed.

1 Introduction

The PASCAL initiative on "textual entailment" had the excellent idea of proposing a competition testing NLP systems on their ability to understand language separate from the ability to cope with world knowledge. This is obviously a welcome endeavor: NLP systems cannot be held responsible for knowledge of what goes on in the world but no NLP system can claim to "understand" language if it can't cope with textual inferences. The task also shies away from creative metaphorical or metonymic use of language and makes the assumption that referential assignments remain constant for entities that are described in the same way. These all seem good features of the proposal as it stands.

Looking at the challenge as it was put before the community, however, we feel that it might be useful to try to circumscribe more precisely what exactly should count as linguistic knowledge. In this paper we make a stab at this in the hope of getting a discussion going. For reasons that will become clear, we prefer to talk about TEXTUAL INFERENCES rather than about textual entailments when referring to the general enterprise. We first explicitate what we think should be covered by the term textual inferences, we then look at the PASCAL development suite in the light of our discussion and we conclude with a short proposal for extensions to the test suite.

Before even starting at this, a point of clarification needs to be made: the correspondence of a linguistic object to an object in the real world goes beyond what can be learned from the text itself. When somebody says or writes *The earth is flat* or *The king of France is bald* because (s)he is a liar or ill-informed, nothing in these linguistic expressions in themselves alerts us to the fact that they do not correspond to situations in the real world (we leave texts in which the author signals consciously or unconsiously that he is lying or fibbing out of consideration here.) What the text does is give us information about the stance its author takes vis-à-vis the events or states described.

It is thus useful to distinguish between two ingredients that go into determining the truth value of an utterance, one is the trustworthiness of the utterer and the other is the stance of the utterer vis-à-vis the truth of the content. The latter we will call the veridicity of the content. When we talk about textual inferences we are only interested in veridicity not in the truth which lies beyond what can be inferred from texts. Or, maybe more realistically, we assume a trustworthy author so that veridical statements are also true.

Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, pages 31–36,
Ann Arbor, June 2005.

2 Varieties of local textual inferences

Under this assumption of trustworthiness, semantics and pragmatics as practiced by philosophers and linguists can give us some insights that are of practical relevance. Work done in the last century has led researchers to distinguish between entailments, conventional implicatures and conversational implicatures. We describe these three classes of inferences and illustrate why the distinctions are important for NLP.

2.1 Entailments

The most uncontroversional textual inferences are those that can be made on the basis of what is asserted in a text. If the author makes the statement that *Tony Hall arrived in Baghdad on Sunday night*, then we can conclude that *Tony Hall was in Baghdad on Sunday night* (keeping referring expressions constant, as proposed in the PASCAL task). The second sentence is true when the first is true (assuming we are talking about the same Tony Hall, the same Baghdad and the same Sunday) just by virtue of what the words mean.

In simple examples such as that in (1)

(1) Bill murdered John.
Bill killed John.

one can go to a resource such as WordNet, look up *murder*, discover that it means *kill* with some further conditions. "Ontologies" or thesauruses typically order terms in a hierarchy that encodes a relation from less specific at the top of the hierarchy to more specific at the bottom. In simple clauses the replacement of a more specific term with a less specific one, ensures an upward monotonic relation between these sentences. As is well known this relation is inversed when the sentences are negated.[1]

(2) Bill didn't murder John.
does not entail *Bill didn't kill John.*

but

(3) Bill didn't kill John.
does entail *Bill didn't murder John.*

Monotonicity relations also hold when adjectival modification is introduced as in (4)

(4) Ames was a clever spy.
entails *Ames was a spy.*

Again negation reverses the entailment:

(5) Ames wasn't a spy.
entails *Ames wasn't a clever spy.*

Quantifiers, easily among the most intensively studied lexical items, also exhibit upward or downward monotonicity.[2] To give just one example:

(6) All companies have to file annual reports.
entails *All Fortune 500 companies have to file annual reports.*

but

(7) All companies have to file annual reports.
does not entail *All companies have to file annual reports to the* SEC.

The fact that there are both upwards monotonic and downwards monotonic expressions means that simple matching on an inclusion of relevant material cannot work as a technique to detect entailments. Upward monotone expressions preserve truth by leaving out material whereas downward monotone expressions don't: adding material to them can be truth preserving.[3]

Apart from a more specific/less specific relation, lexical items can establish a part-subpart relation between the events they describe. If we followed the first sentence in (1) by

(8) John died.

we would still have a lexical inference. In this case one in which the event described in the second sentence is a subpart of the event described in the first.

The investigation of entailments leads one to distinguish several types of lexical items that have predictable effects on meaning that can be exploited to discover sentences that are inferentially related (by real entailments in this case). Other examples are scope bearing elements (an aspect of meaning that often leads to ambiguities which are not always easily perceived) and perception reports.

[1] A sentence is downward monotonic iff it remains true when it is narrowed. A sentence is upward monotonic when it remains true when it is broadened.

[2] A quantifier Q is downward monotonic with respect to its restrictor ϕ iff $((Q\ \phi)\ \psi)$ remains true when the ϕ is narrowed, e.g. from *companies* to *Fortune 500 companies*. A quantifier Q is upward monotonic with respect to its scope ψ iff $((Q\ \phi)\ \psi)$ remains true when ψ is broadened, e.g. from *have to file reports to the* SCE to just *have to file reports*.

[3] Dagan and Glickman (2004) explore inferencing by syntactic pattern matching techniques but consider only upward monotonic expressions. Their proposal ensures loss of recall on downward monotonic expressions.

Two types of relations deserve special mention here because they are pervasive and they are at the borderline between linguistic and world knowledge: temporal relations and spatial relations. Whether knowing that Tuesday follows Monday or that there are leap years and non-leap years is linguistic knowledge or world knowledge might not be totally clear but it is clear that one wants this information to be part of what textual entailment can draw upon. The consequences in a Eucledian space of the place and movement of objects are similar. There is a rich set of entailment relations that builds on these temporal and spatial notions.

2.2 Conventional Implicatures[4]

Apart from making assertions, however, an author will often "conventionally implicate" certain things. We use here the term conventional implicature for what has been called by that name or labeled as (semantic) presupposition. Some of us have argued elsewhere there is no need for a distinction between these two notions (Karttunen and Peters, 1979) and that presupposition is a less felicitous term because it tends to be confused with "old information".

Traditionally these implications are not considered to be part of what makes the sentence true, but the author is COMMITTED to them and we consider them part of what textual inferences should be based on. We take this position because we think it is reasonable, for IE tasks, to assume that material that is conventionally implicated can be used in the same way as assertions, for instance, to provide answers to questions. When somebody says *Bill acknowledges that the earth is round*, we know something about the author's as well as Bill's beliefs in the matter, namely that the author is committed to the belief that the earth is round.

If all conventionally implied material were also discourse old information, this might not matter very much as the same information would be available elsewhere in the text, but often conventionally implied material is new information that is presented as not being under discussion. Conventional implicatures are a rich source of information for IE tasks because the material presented in them is supposed to be non-controversial. In newspapers and other information sources they are a favorite way to distinguish background knowledge, that the reader might have or not, without confusing it with what is newsworthy in the report at hand. A very common example of this, exploited in the PASCAL test suite, is the use of appositives. illustrated in the following example:

(9) The New York Times reported that Hanssen, who sold FBI secrets to the Russians, could face the death penalty.
Did Hanssen sell FBI reports to the Russians?
YES

From the perspective of IE tasks, the way conventional implicatures behave under negation is one reason to pay close attention to them. The following examples illustrate this:

(10) Kerry realized that Bush was right.
Bush was right.

(11) Kerry didn't realize that Bush was right.
Bush was right.

Other types of embedded clauses that are conventionally implicated are temporal adverbials (except those introduced by *before* or *until*. Other types of material that can introduce a conventional implicature are adverbial expressions such as *evidently* and simple adverbs such as *again* or *still*.

It is important to point out that the syntactic structure doesn't guide the interpretation here. Consider the following contrast:

(12) As the press reported, Ames was a successful spy.

conventionally implicates that Ames was a successful spy, but

(13) According to the press, Ames was a successful spy.

does not.

2.3 Conversational Implicatures[5]

Authors can be held responsible for more than just assertions and conventional implicatures. Conversational implicatures are another type of author commitment. A conversational implicature rests on the assumption that, in absence of evidence to the contrary, a collaborative author will say as much as she

[4]For more on conventional implicatures, see e.g. Karttunen and Peters (1979) and Potts (2005)

[5]For more on conversational implicatures, see e.g. Grice (1989) and Horn (2003)

knows. So if Sue says that she has four children, we tend to conclude that she has no more than four. This type of implicature can be destroyed without any contradiction arising: *He not only ate some of the cake, he ate all of it.* Within the context of a textual inference task such as that defined in the PASCAL initiative, it is clear that inferences based on conversational implicatures might be wrong: PASCAL doesn't give the context. In a more developed type of inference task, a distinction should be made between this type of inference and the ones we discussed earlier, but when inferencing is reduced to one sentence it seems more reasonable to take generalized conversational implicatures into account as bona fide cases of inferences (except of course if they are cancelled in the sentence itself, as in the example above).

(14) I had the time to read your paper.

conversationally implies that I read your paper. But it could be followed by *but I decided to go play tennis instead.*

(15) Some soldiers were killed.

conversationally implies *Not all soldiers were killed.* But it could be cancelled by *In fact we fear that all of them are dead.*

(16) He certainly has three children.

conversationally implies *He doesn't have more than three children* but it could be followed by *In fact he has five, three daughters and two sons.*

Apart from the general conversational implicatures, implicatures can also arise by virtue of something being said or not said in a particular context. If in a letter of recommendation, one praises the candidate's handwriting without saying anything about his intellectual abilities, this allows the reader to draw some conclusions. We assume here that this type of inference is not part of the PASCAL task, as too little context is given for it to be reliably calculated.

One might agree with the analysis of various sources of author commitment given above but be of the opinion that it doesn't matter because, given enough data, it will come out in the statistical wash. We doubt, however, that this will happen any time soon without some help: the semantic distinctions are rather subtle and knowing about them will help develop adequate features for statistical training.

It might also be thought that the generalizations that we need here can be reduced to syntactic distinctions. We don't have the space to show in great detail that this is not the case but some reflection on and experimentation with the examples given throughout this paper will convince the reader that this is not the cases. For instance, if one replaces the adjective *clever* with the equally good adjective *alleged* in (4) above, the entailment relation between the sentences doesn't hold anymore. Substituting *show* for *realize* in (11) has the same effect.

2.4 Some world knowledge?

In our mind this exhausts the ways in which an author can be held responsible for her writings on the basis of text internal elements. Textual inferences are based on textual material that is either an entailment of what is explicitly asserted, or material that conventionally or conversationally implied by the author. These inferences can be made solely on the basis of the way the meaning of the words and construction she uses are related to other words and constructions in the language. But even in a task that tries to separate out linguistic knowledge from world knowledge, it is not possible to avoid the latter completely. There is world knowledge that underlies just about everything we say or write: the societies we live in use a common view of time to describe events and rely on the assumptions of Euclidean geometry, leading to shared calendars and measurement systems. It would be impossible to separate these from linguistic knowledge. Then there is knowledge that is commonly available and static, e.g. that Baghdad is in Iraq. It seems pointless to us to exclude the appeal to such knowledge from the test suite but it would be good to define it more explicitly.

3 The PASCAL development suite.

We now discuss some of the PASCAL development set examples in the light of the discussion above and explain why we think some of them do not belong in a textual inference task. First a number of PASCAL examples are based on spelling variants or even spelling mistakes. While it is clear that coping with this type of situation is important for NLP applications we think they do not belong in a textual inference test bed. We first discuss a couple of examples

that we think should not have been in the test suite and then some that do not confirm to our view on inferencing but which might belong in a textual inference test suite.

3.1 Errors?

A problem arises with an example like the following:

(17) A farmer who was in contact with cows suffering from BSE – the so-called mad cow disease – has died from what is regarded as the human form of the disease.
Bovine spongiform encephalopathy is another name for the "mad cow disease".
TRUE

If one googles BSE, one finds that it is an abbreviation that can stand for many things, including the Bombay, Bulgarian, Baku or Bahrain Stock Exchange, Breast Self-Examination, and Brain Surface Extractor. To select the right alternative, one needs the knowledge that "bovine spongiform encephalopathy" is a name of a disease and the other competing BSE expansions are not.

The authors of the PASCAL test suite don't seem to allow for as much world knowledge when they mark the following relation as FALSE.

(18) "I just hope I don't become so blissful I become boring" – Nirvana leader Kurt Cobain said, giving meaning to his "Teen Spirit" coda, a denial.
"Smells Like Teen Spirit" is a song by Nirvana.
FALSE

Apparently, it is NOT OK to know that the Nirvana song "Smells like Teen Spirit" is often referred to as "Teen Spirit". But why should we then know that bovine spongiform encephalopathy is a disease?

The test suite also contains examples that can only be classified as plain errors. A couple of examples are the following:

(19) Green cards are becoming more difficult to obtain.
Green card is now difficult to receive.
TRUE

Something that is becoming more difficult can still be easy, if it starts out that way.

(20) Hippos do come into conflict with people quite often.
Hippopotamus attacks human.
TRUE

For somebody who knows a lot about hippos it might be reasonable to assume that a conflict is necessarily an attack but in general there is no inference: *conflict* is the less general term and *attack* the more specific one.

(21) A statement said to be from al Qaida claimed the terror group had killed one American and kidnapped another in Riyadh.
A U.S. citizen working in Riyadh has been kidnapped.
TRUE

This seems betray a rather implausible belief in the claims of al Qaida and while we are assuming that the author of the text is trustworthy, this assumption does not extend to the sources he invokes. In this case especially, the use of *claim* can be construed as indication the doubt of the author about the veracity of what the source says.

(22) Wal-Mart is being sued by a number of its female employees who claim they were kept out of jobs in management because they were women.
Wal-Mart is sued for sexual discrimination.
TRUE

A minute of reflection will make clear that here the relation between the two sentences involves quite a bit of specialized legal knowledge and goes beyond textual inferencing. How is *sexual discrimination* different from *sexual harassment*?

(23) South Korean's deputy foreign minister says his country won't change its plan to send 3000 soldiers to Iraq.
South Korea continues to send troops.
TRUE

We assume that in context the second sentence means that South Korea continues to plan to send troops but normally *continue* does not mean *continue to plan* and the first sentence certainly doesn't imply that South Korea has already sent troops. Here the way the test suite has been put together leads to odd results. A headline is paired up with a full sentence. Headlines are not meant to be understood completely out of context and it would be prudent to use them sparingly in inference tasks of the sort proposed here. We discuss other consequences of the way the test suite was constructed in the next subsection with examples that to our mind need some kind of accommodation.

3.2 Not a textual inference as such but . . .

There are a couple of examples such as the following in the test suite:

(24) The White House failed to act on the domestic threat from al Qaida prior to September 11, 2001.
White House ignored the threat of attack.
TRUE

Here there is no entailment either way and surely *fail to act* is not a synonym of *ignore*. The examples are due to the way the PASCAL test suite was put together. It was evidently at least in part developed by finding snippets of text that refer to the same event in different news sources; this is a fertile method for finding inferences but it will lead to the inclusion of some material that mixes factual description and various APPRECIATIONS of the described facts. For instance in (24) above, two different authors described what the White house did, putting a different spin on it. While the fact described in both cases was the same, the appreciations that the two renderings give, while both negative, are not equivalent. But although there is no legitimate inference for the sentences as a whole, they both entail that the White House did not act. Here the test suite is the victim of its self imposed constraints, namely that the relation has to be established between two sentences found in "real" text. We propose to give up this constraint.

Another maybe simpler illustration of the same problem is (25):

(25) The report catalogues 10 missed opportunities.
The report lists 10 missed opportunities.

Although *catalogue* and *list* do not have the same meaning, they may in some cases be used interchangeably because, again, there is a common entailment:

(26) According to the report, there were 10 missed opportunities.

One can conceive of a thesaurus where *catalogue* and *list* would have a low level common hypernym (in WordNet they don't) or a statistically inferred word class that would make the common entailment explicit, but that relation should not be confused with an inference between the two sentences in (25).

4 A proposal for some refinements

As the discussion above has shown, the way the test suite was put together leads sometimes to the inclusion of material that should not be there given the definition of the task. Most of the data that form the basis of PASCAL are extracted from different newspaper articles about the same event, often from the same newswire. This means that the information packaging is very similar, reducing the constructional and lexical range that can be used to express a same idea. This situation will not pertain in the more general setting of question answering and many types of paraphrases or inferences that would be useful for question answering in general will not be found or will be very rare in PASCAL-like suites.

We would propose to augment the types of pairs that one can get through the PASCAL extraction techniques with some that take the type of relations that we have discussed explicitly into account. It can be objected that this introduces a new level of artificiality by allowing made-up sentences but the separation of world knowledge from linguistic knowledge is in any case artificial. But it is necessary because we will not be able to solve the inferencing problem without slicing the task into manageable pieces.

Acknowledgments

This article was supported in part by the Advanced Research and Development Agency (ARDA) within the program for Advanced Question Answering for Intelligence (AQUAINT). Thanks to all the members of PARC's AQUAINT team.

References

Ido Dagan and Oren Glickman. 2004. Probabilistic textual entailment: Generic applied modeling of language variablity. In *Learning Methods for Text Understanding and Mining*, Grenoble, January.

Paul H. Grice. 1989. *Studies in the Way of Words*. Harvard University, Cambridge, MA.

Larry Horn. 2003. Implicature. In Horn and Ward, editors, *Handbook of Pragmatics*. Blackwell, Oxford.

Lauri Karttunen and Stanley Peters. 1979. Conventional implicature. In Choon-Kyu Oh and David A. Dinneen, editors, *Syntax and Semantics, Volume 11: Presupposition*, pages 1–56. Academic Press, New York.

Christopher Potts. 2005. *The Logic of Conventional Implicatures*. Oxford Studies in Theoretical Linguistics. Oxford University Press, Oxford.

Discovering entailment relations using *"textual entailment patterns"*

Fabio Massimo Zanzotto
DISCo, University of Milano-Bicocca,
Via Bicocca degli Arcimboldi 8, Milano, Italy,
zanzotto@disco.unimib.it

Maria Teresa Pazienza, Marco Pennacchiotti
DISP, University of Rome "Tor Vergata",
Viale del Politecnico 1, Roma, Italy,
{pennacchiotti, pazienza}@info.uniroma2.it

Abstract

In this work we investigate methods to enable the detection of a specific type of textual entailment (*strict* entailment), starting from the preliminary assumption that these relations are often clearly expressed in texts. Our method is a statistical approach based on what we call *textual entailment patterns*, prototypical sentences hiding entailment relations among two activities. We experimented the proposed method using the entailment relations of WordNet as test case and the web as corpus where to estimate the probabilities; obtained results will be shown.

1 Introduction

Textual entailment has been recently defined as a common solution for modelling language variability in different NLP tasks (Glickman and Dagan, 2004). Roughly, the problem is to recognise if a given textual expression, the *text* (t), entails another expression, the *hypothesis* (h). An example is determining whether or not "*Yahoo acquired Overture* (t) entails *Yahoo owns Overture* (h)". More formally, the problem of determining a textual entailment between t and h is to find a possibly graded truth value for the entailment relation $t \rightarrow h$.

Since the task involves natural language expressions, textual entailment has a more difficult nature with respect to logic entailment, as it hides two different problems: *paraphrase detection* and what can be called *strict entailment detection*. Generally, this task is faced under the simplifying assumption that the analysed text fragments represent *facts* (f_t for the ones in the text and f_h for those in the hypothesis) in an assertive or negative way. *Paraphrase detection* is then needed when the hypothesis h carries a fact f that is also in the target text t but is described with different words, e.g., *Yahoo acquired Overture* vs. *Yahoo bought Overture*. On the other hand, *strict entailment* emerges when target sentences carry different facts, $f_h \neq f_t$. The challenge here is to derive the truth value of the entailment $f_t \rightarrow f_h$. For example, a strict entailment is "*Yahoo acquired Overture* $\rightarrow$ *Yahoo owns Overture*". In fact, it does not depend on the possible paraphrasing between the two expressions but on an entailment of the two *facts* governed by *acquire* and *own*.

Whatever the form of textual entailment is, the real research challenge consists in finding a relevant number of *textual entailment prototype relations* such as "*X acquired Y* entails *X owns Y*" or "*X acquired Y* entails *X bought Y*" that can be used to recognise entailment relations. Methods for acquiring such textual entailment prototype relations are based on the assumption that specific facts are often repeated in possibly different linguistic forms. These forms may be retrieved using their *anchors*, generally nouns or noun phrases completely characterising specific facts. The retrieved text fragments are thus considered alternative expressions for the same fact. This supposed equivalence is then exploited to derive textual entailment prototype relations. For example, the specific fact *Yahoo bought Overture* is characterised by the two anchors

Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, pages 37–42,
Ann Arbor, June 2005.

{*Yahoo, Overture*}, that are used to retrieve in the corpus text fragments where they co-occur, e.g. *"Yahoo purchased Overture (July 2003)."*, *"Now that Overture is completely owned by Yahoo!..."*. These retrieved text fragments are then considered good candidate for paraphrasing *X bought Y*.

Anchor-based learning methods have been used to investigate many semantic relations ranging from very general ones as the *isa* relation in (Morin, 1999) to very specific ones as in (Ravichandran and Hovy, 2002) where paraphrases of question-answer pairs are searched in the web or as in (Szpektor et al., 2004) where a method to scan the web for searching textual entailment prototype relations is presented. These methods are mainly devoted to induce entailment pairs related to the first kind of textual entailment, that is, *paraphrasing* as their target is mainly to look for the same "fact" in different textual forms. Incidentally, these methods can come across strict entailment relations whenever specific anchors are used for both a fact f_t and a *strictly* entailed fact f_h.

In this work we will investigate specific methods to induce the second kind of textual entailment relations, that is, *strict* entailment. We will focus on entailment between verbs, due to the fact that verbs generally govern the meaning of sentences. The problem we are facing is to look for (or verify) entailment relations like $v_t \rightarrow v_h$ (where v_t is the text verb and v_h the hypothesis verb). Our approach is based on an intuition: strict entailment relations among verbs are often clearly expressed in texts. For instance the text fragment *"Player wins $50K in Montana Cash"* hides an entailment relation between two activities, namely *play* and *win*. If someone wins, he has first of all to play, thus, *win* $\rightarrow$ *play*. The idea exploits the existence of what can be called *textual entailment pattern*, a prototypical sentence hiding an entailment relation among two activities. In the abovementioned example the pattern instance *player win* subsumes the entailment relation "*win* $\rightarrow$ *play*".

In the following we will firstly describe in Sec. 2 our method to recognise entailment relations between verbs that uses: (1) the prior linguistic knowledge of these *textual entailment patterns* and (2) statistical models to assess stability of the implied relations in a corpus. Then, we will experiment our method by using the WordNet entailment relations as test cases and the web as corpus where to estimate the probabilities (Sec. 3). Finally we will draw some conclusions (Sec. 4).

2 The method

Discovering entailment relations within texts implies the understanding of two aspects: firstly, how these entailment relations are usually expressed and, secondly, when an entailment relation may be considered stable and commonly shared. Assessing the first aspect requires the investigation of which are the prototypical textual forms that describe entailment relations. We will call them *textual entailment patterns*. These patterns (analysed in Sec. 2.2) will enable the detection of *point-wise entailment assertions*, that is, candidate verb pairs that still need a further step of analysis in order to be considered true entailment expressions. In fact, some of these candidates may be not enough stable and commonly shared in the language to be considered true entailments. To better deal with this second aspect, methods for statistically analysing large corpora are needed (see later in Sec. 2.3).

The method we propose may be used in either: (1) *recognising* if entailment holds between two verbs, or, (2) *extracting* from a corpus C all the implied entailment relations. In *recognition*, given a verb pair, the related textual entailment expressions are derived as instances of the *textual entailment patterns* and, then, the statistical entailment indicators on a corpus C are computed to evaluate the stability of the relation. In *extraction*, the corpus C should be scanned to extract textual expressions that are instances of the textual entailment patterns. The resulting pairs are sorted according to the statistical entailment indicators and only the best ranked are retained as useful verb entailment pairs.

2.1 An intuition

Our method stems from an observation: verb logical subjects, as any verb role filler, have to satisfy specific preconditions as the theory of *selectional restrictions* suggests. Then, if in a given sentence a verb v has a specific logical subject x, its selectional restrictions imply that the subject has to satisfy some preconditions p, that is, $v(x) \rightarrow p(x)$. This can be read also as: if x has the property of doing the action

v this implies that x has the property p. For example, if the verb is *to eat*, the selectional restrictions of *eat* would imply, among other things, that its subject is an *animal*. If the precondition p is "having the property of doing an action a", the constraint may imply that the action v entails the action a, that is, $v \rightarrow a$.

As for selectional restriction acquisition, the previous observation can enable the use of corpora as enormous sources of candidate entailment relations among verbs. For example *"John McEnroe won the match..."* can contribute to the definition of the selectional restriction $win(x) \rightarrow human(x)$ (since *John McEnroe* is a *human*), as well as to the induction (or verification) of the entailment relation between *win* and *play*, since *John McEnroe* has the *property of playing*. However, as the example shows, classes relevant for acquiring selectional preferences may be more explicit than active properties useful to derive entailment relations (i.e., it is easier to derive that *John McEnroe* is a human than that he has the property of playing).

This limitation can be overcome when *agentive nouns* such as *runner* play subject roles in some sentences. Agentive nouns usually denote the "doer" or "performer" of some action a. This is exactly what is needed to make clearer the relevant property of the noun playing the logical subject role, in order to discover entailment. The action a will be the one entailed by the verb heading the sentence. For example, in *"the player wins"*, the action *play* evocated by the agentive noun *player* is entailed by *win*.

2.2 Textual entailment patterns

As observed for the *isa* relations in (Hearst, 1992) local and simple inter-sentential patterns may carry relevant semantic relations. As we saw in the previous section, this also happens for entailment relations. Our aim is thus to search for an initial set of textual patterns that describe possible linguistic forms expressing entailment relations between two verbs (v_t, v_h). By using these patterns, actual pointwise assertions of entailment can be detected or verified in texts. We call these prototypical patterns *textual entailment patterns*.

The idea described in Sec. 2.1 can be straightforwardly applied to generate textual entailment patterns, as it often happens that verbs can undergo an agentive nominalization (hereafter called *personification*), e.g., *play* vs. *player*. Whether or not an entailment relation between two verbs (v_t, v_h) holds according to some writer can be verified looking for sentences with expressions involving the agentive nominalization of the hypothesis verb v_h. Then, the procedure to verify if entailment between two verbs (v_t, v_h) holds in a point-wise assertion is: whenever it is possible to personify the hypothesis v_h, scan the corpus to detect the expressions where the personified hypothesis verb is the subject of a clause governed by the text verb v_t.

Given the two investigated verbs (v_t, v_h) we will refer to this first set of textual entailment patterns as *personified patterns* $\mathcal{P}_{pers}(v_t, v_h)$. This set will contain the following textual patterns:

$$
\begin{array}{l}
\mathcal{P}_{pers}(v_t, v_h) = \\
\{\text{“}pers(v_h)|_{number:sing} \quad v_t|_{person:third,tense:present}\text{”}, \\
\text{“}pers(v_h)|_{number:plur} \quad v_t|_{person:nothird,tense:present}\text{”}, \\
\text{“}pers(v_h)|_{number:sing} \quad v_t|_{tense:past}\text{”}, \\
\text{“}pers(v_h)|_{number:plur} \quad v_t|_{tense:past}\text{”}\}
\end{array}
$$

where $pers(v)$ is the noun deriving from the personification of the verb v and elements such as $l|_{f_1,...,f_N}$ are the tokens generated from lemmas l by applying constraints expressed via the features $f_1, ..., f_N$. For example, in the case of the verbs *play* and *win*, the related set of textual entailment expressions derived from the patterns will be $\mathcal{P}_{pers}(win, play)$ = { *"player wins"*, *"players win"*, *"player won"*, *"players won"* }. In the experiments hereafter described, the required verbal inflections (except personification) have been obtained using the publicly available morphological tools described in (Minnen et al., 2001) whilst simple heuristics have been used to personify verbs[1].

As the statistical measures introduced in the following section are those usually used for studying co-occurrences, two more sets of expressions, $\mathcal{F}_{pers}(v)$ and $\mathcal{F}(v)$, are needed to represent the single events in the pair. These are defined as:

$$
\begin{array}{ll}
\mathcal{F}_{pers}(v) = & \{\text{“}pers(v)|_{number:sing}\text{”}, \text{“}pers(v)|_{number:plur}\text{”}\} \\
\mathcal{F}(v) = & \{\text{“}v|_{person:third,tense:present}\text{”}, \\
 & \text{“}v|_{person:nothird,tense:present}\text{”}, \text{“}v|_{tense:past}\text{”}\}
\end{array}
$$

[1] Personification, i.e. agentive nominalization, has been obtained adding "-er" to the verb root taking into account possible special cases such as verbs ending in "-y". A form is retained as a correct personification if it is in WordNet.

2.3 Measures to estimate the entailment strength

The above textual entailment patterns define *point-wise* entailment assertions. In fact, if pattern instances are found in texts, the only conclusion that may be drawn is that someone (the author of the text) sustains the related entailment pairs. A sentence like *"**Painter draws** on old techniques but creates only decorative objects."* suggests that *painting* entails *drawing*. However, it may happen that these correctly detected entailments are accidental, that is, the detected relation is only valid for that given text. For example, the text fragment *"When a **painter discovers** this hidden treasure, other people are immediately struck by its beauty."* if taken in insulation suggests that *painting* entails *discovering*, but this is questionable. Furthermore, it may also happen that patterns detect wrong cases due to ambiguous expressions like *"**Painter draws** inspiration from forest, field"* where the sense of the verb *draw* is not the one expected.

In order to get rid of these wrong verb pairs, an assessment of point-wise entailment assertions over a corpus is needed to understand how much the derived entailment relations are shared and commonly agreed. This validation activity can be obtained by both analysing large textual collections and applying statistical measures relevant for the task.

Before introducing the statistical entailment indicators, some definitions are necessary. Given a corpus C containing samples, we will refer to the absolute frequency of a textual expression t in the corpus C with $f_C(t)$. The definition is easily extended to a set of expressions T as follows:

$$f_C(T) = \sum_{t \in T} f_C(t)$$

Given a pair v_t and v_h we may thus define the following *entailment strength indicators* $\mathcal{S}(v_t, v_h)$, related to more general statistical measures.

The first relevance indicator, $\mathcal{S}_f(v_t, v_h)$, is related to the probability of the textual entailment pattern as it is. This probability may be represented by the frequency, as the fixed corpus C makes constant the total number of pairs:

$$\mathcal{S}_f(v_t, v_h) = log_{10}(f_C(\mathcal{P}_{pers}(v_t, v_h)))$$

where logarithm is used to contrast the effect of the Zipf's law. This measure is often positively used in terminology extraction (e.g., (Daille, 1994)).

Secondly, another measure $\mathcal{S}_{mi}(v_t, v_h)$ related to point-wise mutual information (Fano, 1961) may be also used. Given the possibility of estimating the probabilities through maximum-likelihood principle, the definition is straightforward:

$$\mathcal{S}_{mi}(v_t, v_h) = log_{10} \frac{p(\mathcal{P}_{pers}(v_t, v_h))}{p(\mathcal{F}_{pers}(v_t))p(\mathcal{F}(v_h))}$$

where $p(x) = f_C(x)/f_C(.)$. The aim of this measure is to indicate the relatedness between two elements composing a pair. Mutual information has been positively used in many NLP tasks such as collocation analysis (Church and Hanks, 1989), terminology extraction (Damerau, 1993), and word sense disambiguation (Brown et al., 1991).

3 Experimental Evaluation

As many other corpus linguistic approaches, our entailment detection model relies partially on some linguistic prior knowledge (the expected structure of the searched collocations, i.e., the *textual entailment patterns*) and partially on some probability distribution estimation. Only a positive combination of both these two ingredients can give good results when applying (and evaluating) the model.

The aim of the experimental evaluation is then to understand, on the one side, if the proposed *textual entailment patterns* are useful to detect entailment between verbs and, on the other, if a statistical measure is preferable with respect to the other. We will here evaluate the capability of our method to *recognise* entailment between given pairs of verbs.

We carried out the experiments using the web as the corpus C where to estimate our two textual entailment measures ($\mathcal{S}_f$ and $\mathcal{S}_{mi}$) and Google^{TM} as a count estimator. The findings described in (Keller and Lapata, 2003) seem to suggest that count estimations we need in the present study over *Subject-Verb* bigrams are highly correlated to corpus counts.

As test bed we used existing resources: a non trivial set of controlled verb entailment pairs is in fact contained in WordNet (Miller, 1995). There, the entailment relation is a semantic relation defined at the synset level, standing in the verb subhierarchy. Each

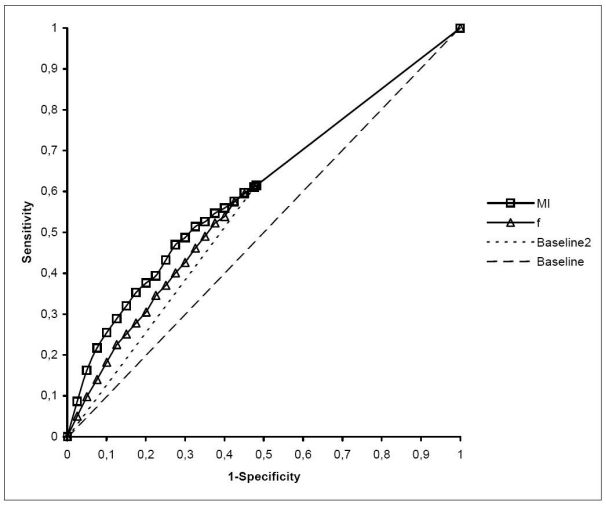

Figure 1: ROC curves

pair of synsets (S_t, S_h) is an oriented entailment relation between S_t and S_h. WordNet contains 415 entailed synsets. These entailment relations are consequently stated also at the lexical level. The pair (S_t, S_h) naturally implies that v_t entails v_h for each possible $v_t \in S_t$ and $v_h \in S_h$. It is then possible to derive from the 415 entailment synset a test set of 2,250 verb pairs. As the proposed model is applicable only when hypotheses can be personified, the number of the pairs relevant for the experiment is thus reduced to 856. This set is hereafter called the *True Set* (TS).

As the *True Set* is our starting point for the evaluation, it is not possible to produce a natural distribution in the verb pair space between entailed and not-entailed elements. Then, precision, recall, and f-measure are not applicable. The only solution is to use a ROC (Green and Swets, 1996) curve mixing *sensitity* and *specificity*. What we then need is a *Control Set* (CS) of verb pairs that in principle are not in entailment relation. The *Control Set* has been randomly built on the basis of the *True Set*: given the set of all the hypothesis verbs H and the set of all the text verbs T of the *True Set*, control pairs are obtained randomly extracting one element from H and one element from T. A pair is considered a control pair if it is not in the *True Set*. For comparative purposes the *Control Set* has the same cardinality of the *True Set*. However, even if the intersection between the *True Set* and the *Control Set* is empty, we are not completely sure that the *Control Set* does not contains any pair where the entailment relation holds. What we may assume is that this last set at least contains a smaller number of positive pairs.

Sensitivity, i.e. the probability of having positive answers for positive pairs, and *specificity*, i.e. the probability of having negative answers for negative pairs, are then defined as:

$$Sensitivity(t) = p((v_h, v_t) \in TS | \mathcal{S}(v_h, v_t) > t)$$
$$Specificity(t) = p((v_h, v_t) \in CS | \mathcal{S}(v_h, v_t) < t)$$

where $p((v_h, v_t) \in TS | \mathcal{S}(v_h, v_t) > t)$ is the probability of a candidate pair (v_h, v_t) to belong to *TS* if the test is positive, i.e. the value $\mathcal{S}(v_h, v_t)$ of the entailment detection measure is greater than t, while $p((v_h, v_t) \in CS | \mathcal{S}(v_h, v_t) < t)$ is the probability of belonging to *CS* if the test is negative. The ROC curve ($Sensitivity$ vs. $1 - Specificity$) naturally follows (see Fig. 1).

Results are encouraging as textual entailment patterns show a positive correlation with the entailment relation. Both ROC curves, the one related to the frequency indicator $\mathcal{S}_f$ (f in figure) and the one related to the mutual information $\mathcal{S}_{MI}$ (MI in figure), are above the $Baseline$ curve. Moreover, both curves are above the second baseline ($Baseline2$) applicable when it is really possible to use the indicators. In fact, textual entailment patterns have a non-zero frequency only for 61.4% of the elements in the *True Set*. This is true also for 48.1% of the elements in the *Control Set*. The presence-absence in the corpus is then already an indicator for the entailment relation of verb pairs, but the application of the two indicators can help in deciding among elements that have a non-zero frequency in the corpus. Finally, in this case, mutual information appears to be a better indicator for the entailment relation with respect to the frequency.

4 Conclusions

We have defined a method to recognise and extract entailment relations between verb pairs based on what we call *textual entailment pattern*. In this work we defined a first kernel of *textual entailment patterns* based on subject-verb relations. Potentials of the method are still high as different kinds of textual

entailment patterns may be defined or discovered investigating relations between sentences and subsentences as done in (Lapata and Lascarides, 2004) for temporal relations or between near sentences as done in (Basili et al., 2003) for cause-effect relations between domain events. Some interesting and simple inter-sentential patters are defined in (Chklovski and Pantel, 2004). Moreover, with respect to anchor-based approaches, the method we presented here offers a different point of view on the problem of acquiring textual entailment relation prototypes, as textual entailment patterns do not depend on the repetition of "similar" facts. This practically independent view may open the possibility to experiment co-training algorithms (Blum and Mitchell, 1998) also in this area. Finally, the approach proposed can be useful to define better probability estimations in probabilistic entailment detection methods such as the one described in (Glickman et al., 2005).

References

Roberto Basili, Maria Teresa Pazienza, and Fabio Massimo Zanzotto. 2003. Inducing hyperlinking rules from text collections. In *Proceedings of the International Conference Recent Advances of Natural Language Processing (RANLP-2003)*, Borovets, Bulgaria.

Avrim Blum and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *COLT: Proceedings of the Conference on Computational Learning Theory*. Morgan Kaufmann.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1991. Word-sense disambiguation using statistical methods. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics (ACL)*, Berkely, CA.

Timoty Chklovski and Patrick Pantel. 2004. VerbOCEAN: Mining the web for fine-grained semantic verb relations. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, Barcellona, Spain.

K.W. Church and P. Hanks. 1989. Word association norms, mutual information and lexicography. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics (ACL)*, Vancouver, Canada.

Beatrice Daille. 1994. *Approche mixte pour l'extraction de terminologie: statistque lexicale et filtres linguistiques*. Ph.D. thesis, C2V, TALANA, Universitè Paris VII.

F.J. Damerau. 1993. Evaluating domain-oriented multiword terms from text. *Information Processing and Management*, 29(4):433–447.

R.M. Fano. 1961. *Transmission of Information: a statistical theory of communications*. MIT Press, Cambridge,MA.

Oren Glickman and Ido Dagan. 2004. Probabilistic textual entailment: Generic applied modeling of language variability. In *Proceedings of the Workshop on Learning Methods for Text Understanding and Mining*, Grenoble, France.

Oren Glickman, Ido Dagan, and Moshe Koppel. 2005. Web based probabilistic textual entailment. In *Proceedings of the 1st Pascal Challenge Workshop*, Southampton, UK.

D.M. Green and J.A. Swets. 1996. *Signal Detection Theory and Psychophysics*. John Wiley and Sons, New York, USA.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 15th International Conference on Computational Linguistics (CoLing-92)*, Nantes, France.

Frank Keller and Mirella Lapata. 2003. Using the web to obtain frequencies for unseen bigrams. *Computational Linguistics*, 29(3), September.

Mirella Lapata and Alex Lascarides. 2004. Inferring sentence-internal temporal relations. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, Boston, MA.

George A. Miller. 1995. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41, November.

G. Minnen, J. Carroll, and D. Pearce. 2001. Applied morphological processing of english. *Natural Language Engineering*, 7(3):207–223.

Emmanuel Morin. 1999. *Extraction de liens sémantiques entre termes à partir de corpus de textes techniques*. Ph.D. thesis, Univesité de Nantes, Faculté des Sciences et de Techniques.

Deepak Ravichandran and Eduard Hovy. 2002. Learning surface text patterns for a question answering system. In *Proceedings of the 40th ACL Meeting*, Philadelphia, Pennsilvania.

Idan Szpektor, Hristo Tanev, Ido Dagan, and Bonaventura Coppola. 2004. Scaling web-based acquisition of entailment relations. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, Barcellona, Spain.

A Probabilistic Setting and Lexical Cooccurrence Model for Textual Entailment

Oren Glickman and Ido Dagan
Department of Computer Science
Bar Ilan University
{glikmao,Dagan}@cs.biu.ac.il

Abstract

This paper proposes a general probabilistic setting that formalizes a probabilistic notion of textual entailment. We further describe a particular preliminary model for lexical-level entailment, based on document cooccurrence probabilities, which follows the general setting. The model was evaluated on two application independent datasets, suggesting the relevance of such probabilistic approaches for entailment modeling.

1 Introduction

Many Natural Language Processing (NLP) applications need to recognize when the meaning of one text can be expressed by, or inferred from, another text. Information Retrieval (IR), Question Answering (QA), Information Extraction (IE), text summarization and Machine Translation (MT) evaluation are examples of applications that need to assess this semantic relationship between text segments. The Textual Entailment Recognition task (Dagan et al., 2005) has recently been proposed as an application independent framework for modeling such inferences.

Within the textual entailment framework, a text *t* is said to entail a textual hypothesis *h* if the truth of *h* can be inferred from *t*. Textual entailment captures generically a broad range of inferences that are relevant for multiple applications. For example, a QA system has to identify texts that entail a hypothesized answer. Given the question "Does John Speak French?", a text that includes the sentence "*John is a fluent French speaker*" entails the suggested answer "John speaks French." In many cases, though, entailment inference is uncertain and has a probabilistic nature. For example, a text that includes the sentence "*John was born in France.*" does not strictly entail the above answer. Yet, it is clear that it does increase substantially the likelihood that the hypothesized answer is true.

The uncertain nature of textual entailment calls for its explicit modeling in probabilistic terms. We therefore propose a general generative probabilistic setting for textual entailment, which allows a clear formulation of concrete probabilistic models for this task. We suggest that the proposed setting may provide a unifying framework for modeling uncertain semantic inferences from texts.

An important sub task of textual entailment, which we term *lexical entailment*, is recognizing if the lexical concepts in a hypothesis *h* are entailed from a given text *t*, even if the relations which hold between these concepts may not be entailed from *t*. This is typically a necessary, but not sufficient, condition for textual entailment. For example, in order to infer from a text the hypothesis "Chrysler stock rose," it is a necessary that the concepts of *Chrysler*, *stock* and *rise* must be inferred from the text. However, for proper entailment it is further needed that the right relations hold between these concepts. In this paper we demonstrate the relevance of the general probabilistic setting for modeling lexical entailment, by devising a preliminary model that is based on document co-occurrence probabilities in a bag of words representation.

Although our proposed lexical system is relatively simple, as it doesn't rely on syntactic or other deeper analysis, it nevertheless was among the top ranking systems in the first Recognising Textual Entailment (RTE) Challenge (Glickman et al., 2005a). The model was evaluated also on an additional dataset, where it compares favorably with a state-of-the-art heuristic score. These results suggest that the proposed probabilistic framework is a promising basis for devising improved models that incorporate richer information.

Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, pages 43–48,
Ann Arbor, June 2005.

example	text	hypothesis
1	*John is a French Speaker*	John speaks French
2	*John was born in France*	
3	*Harry's birthplace is Iowa*	*Harry was born in Iowa*
4	*Harry is returning to his Iowa hometown*	

Table 1: example sentence pairs

2 Probabilistic Textual Entailment

2.1 Motivation

A common definition of entailment in formal semantics (Chierchia. and McConnell-Ginet, 1990) specifies that a text t entails another text h (hypothesis, in our terminology) if h is true in every circumstance (possible world) in which t is true. For example, in examples 1 and 3 from Table 1 we'd assume humans to agree that the hypothesis is necessarily true in any circumstance for which the text is true. In such intuitive cases, textual entailment may be perceived as being certain, or, taking a probabilistic perspective, as having a probability of 1.

In many other cases, though, entailment inference is uncertain and has a probabilistic nature. In example 2, the text doesn't contain enough information to infer the hypothesis' truth. And in example 4, the meaning of the word *hometown* is ambiguous and therefore one cannot infer for certain that the hypothesis is true. In both of these cases there are conceivable circumstances for which the text is true and the hypothesis false. Yet, it is clear that in both examples, the text does increase substantially the likelihood of the correctness of the hypothesis, which naturally extends the classical notion of certain entailment. Given the text, we expect the probability that the hypothesis is indeed true to be relatively high, and significantly higher than its probability of being true without reading the text. Aiming to model application needs, we suggest that the probability of the hypothesis being true given the text reflects an appropriate confidence score for the correctness of a particular textual inference. In the next subsections we propose a concrete probabilistic setting that formalizes the notion of truth probabilities in such cases.

2.2 A Probabilistic Setting

Let T denote a space of possible texts, and $t \in T$ a specific text. Let H denote the set of all possible *hypotheses*. A hypothesis $h \in H$ is a propositional statement which can be assigned a truth value. For now it is assumed that h is represented as a textual statement, but in principle it could also be expressed as a formula in some propositional language.

A semantic state of affairs is captured by a mapping from H to {0=false, 1=true}, denoted by $w: H \rightarrow \{0, 1\}$ (called here *possible world*, following common terminology). A possible world w represents a concrete set of truth value assignments for all possible propositions. Accordingly, W denotes the set of all possible worlds.

2.2.1 A Generative Model

We assume a probabilistic generative model for texts and possible worlds. In particular, we assume that texts are generated along with a concrete state of affairs, represented by a possible world. Thus, whenever the source generates a text t, it generates also corresponding hidden truth assignments that constitute a possible world w.

The probability distribution of the source, over all possible texts and truth assignments $T \times W$, is assumed to reflect inferences that are based on the generated texts. That is, we assume that the distribution of truth assignments is not bound to reflect the state of affairs in a particular "real" world, but only the inferences about propositions' truth which are related to the text. In particular, the probability for generating a true hypothesis h that is not related at all to the corresponding text is determined by some prior probability $P(h)$. For example, h="Paris is the capital of France" might have a prior smaller than 1 and might well be false when the generated text is not related at all to Paris or France. In fact, we may as well assume that the notion of textual entailment is relevant only for hypotheses for which $P(h) < 1$, as otherwise (i.e. for tautologies) there is no need to consider texts that would support h's truth. On the other hand, we assume that the probability of h being true (generated within w) would be higher than the prior when the corresponding t does contribute information that supports h's truth.

We define two types of events over the probability space for $T \times W$:
I) For a hypothesis h, we denote as Tr_h the random variable whose value is the truth value assigned to h in a given world. Correspondingly, Tr_h=1 is the event of h being assigned a truth value of 1 (true).
II) For a text t, we use t itself to denote also the event that the generated text is t (as usual, it is clear from the context whether t denotes the text or the corresponding event).

2.3 Probabilistic textual entailment definition

We say that a text t probabilistically entails a hypothesis h (denoted as $t \Rightarrow h$) if t increases the likelihood of h being true, that is, if $P(Tr_h = 1| t) > P(Tr_h = 1)$ or equivalently if the pointwise mutual information, I(Trh=1,t), is greater then 0. Once *knowing* that $t \Rightarrow h$, $P(Tr_h=1| t)$ serves as a probabilistic confidence value for h being true given t.

Application settings would typically require that $P(Tr_h = 1| t)$ obtains a high value; otherwise, the text would not be considered sufficiently relevant to support h's truth (e.g. a supporting text in QA or IE should entail the extracted information with high confidence). Finally, we ignore here the case in which t contributes negative information about h, leaving this relevant case for further investigation.

2.4 Model Properties

It is interesting to notice the following properties and implications of our model:
A) Textual entailment is defined as a relationship between texts and propositions whose representation is typically based on text as well, unlike logical entailment which is a relationship between propositions only. Accordingly, textual entailment confidence is conditioned on the actual generation of a text, rather than its truth. For illustration, we would expect that the text "His father was born in Italy" would logically entail the hypothesis "He was born in Italy" with high probability – since most people who's father was born in Italy were also born there. However we expect that the text would actually not probabilistically textually entail the hypothesis since most people for whom it is specifically reported that their father was born in Italy were not born in Italy.[1]
B) We assign probabilities to propositions (hypotheses) in a similar manner to certain probabilistic reasoning approaches (e.g. Bacchus, 1990; Halpern, 1990). However, we also assume a generative model of text, similar to probabilistic language and machine translation models, which supplies the needed conditional probability distribution. Furthermore, since our conditioning is on texts rather than propositions we do not assume any specific logic representation language for text meaning, and only assume that textual hypotheses can be assigned truth values.
C) Our framework does not distinguish between textual entailment inferences that are based on knowledge of language semantics (such as *murdering* $\Rightarrow$ *killing*) and inferences based on domain or world knowledge (such as *live in Paris* $\Rightarrow$ *live in France*). Both are needed in applications and it is not clear at this stage where and how to put such a borderline.
D) An important feature of the proposed framework is that for a given text many hypotheses are likely to be true. Consequently, for a given text t and hypothesis h, $\sum_h P(Tr_h=1|t)$ does not sum to 1. This differs from typical generative settings for IR and MT (Ponte and croft, 1998; Brown et al., 1993), where all conditioned events are disjoint by construction. In the proposed model, it is rather the case that $P(Tr_h=1|t) + P(Tr_h=0|t) = 1$, as we are interested in the probability that a single particular hypothesis is true (or false).
E) An implemented model that corresponds to our probabilistic setting is expected to produce an estimate for $P(Tr_h = 1| t)$. This estimate is expected to reflect all probabilistic aspects involved in the modeling, including inherent uncertainty of the entailment inference itself (as in example 2 of Table 1), possible uncertainty regarding the correct disambiguation of the text (example 4), as well as probabilistic estimates that stem from the particular model structure.

3 A Lexical Entailment Model

We suggest that the proposed setting above provides the necessary grounding for probabilistic

[1] This seems to be the case, when analyzing the results of entering the above text in a web search engine.

modeling of textual entailment. Since modeling the full extent of the textual entailment problem is clearly a long term research goal, in this paper we rather focus on the above mentioned sub-task of *lexical entailment* - identifying when the lexical elements of a textual hypothesis *h* are inferred from a given text *t*.

To model lexical entailment we first assume that the meanings of the individual content words in a hypothesis can be assigned truth values. One possible interpretation for such truth values is that lexical concepts are assigned existential meanings. For example, for a given text *t*, $Tr_{book}=1$ if it can be inferred in *t*'s state of affairs that a book exists. Our model does not depend on any such particular interpretation, though, as we only assume that truth values can be assigned for lexical items but do not explicitly annotate or evaluate this sub-task.

Given this setting, a hypothesis is assumed to be true if and only if all its lexical components are true as well. This captures our target perspective of lexical entailment, while not modeling here other entailment aspects. When estimating the entailment probability we assume that the truth probability of a term *u* in a hypothesis *h* is independent of the truth of the other terms in *h*, obtaining:

$$P(Tr_h = 1 | t) = \Pi_{u \in h} P(Tr_u = 1 | t)$$
$$P(Tr_h = 1) = \Pi_{u \in h} P(Tr_u = 1) \quad (1)$$

In order to estimate $P(Tr_u=1|v_1, \ldots, v_n)$ for a given word *u* and text $t=\{v_1, \ldots, v_n\}$, we further assume that the majority of the probability mass comes from a specific entailing word in *t*:

$$P(Tr_u = 1 | t) = \max_{v \in t} P(Tr_u = 1 | T_v) \quad (2)$$

where *Tv* denotes the event that a generated text contains the word *v*. This corresponds to expecting that each word in *h* will be entailed from a specific word in *t* (rather than from the accumulative context of *t* as a whole[2]). Alternatively, one can view (2) as inducing an alignment between terms in the *h* to the terms in the *t,* somewhat similar to alignment models in statistical MT (Brown et al., 1993).

Thus we propose estimating the entailment probability based on lexical entailment probabilities from (1) and (2) as follows:

$$P(Tr_h = 1 | t) = \prod_{u \in h} \max_{v \in t} P(Tr_u = 1 | T_v) \quad (3)$$

[2] Such a model is proposed in (Glickman et al., 2005b)

3.1 Estimating Lexical Entailment Probabilities

We perform unsupervised empirical estimation of the lexical entailment probabilities, $P(Tr_u=1|T_v)$, based on word co-occurrence frequencies in a corpus. Following our proposed probabilistic model (cf. Section 2.2.1), we assume that the domain corpus is a sample generated by a language source. Each document represents a generated text and a (hidden) possible world. Given that the possible world of the text is not observed we do not know the truth assignments of hypotheses for the observed texts. We therefore further make the simplest assumption that all hypotheses stated verbatim in a document are true and all others are false and hence $P(Tr_u=1|T_v) = P(T_u|T_v)$. This simple co-occurrence probability, which we denote as lexical entailment probability – $lep(u,v)$, is easily estimated from the corpus based on maximum likelihood counts:

$$lep(u, v) = P(Tr_u = 1 | T_v) \approx \frac{n_{u,v}}{n_v} \quad (4)$$

where n_v is the number of documents containing word *v* and $n_{u,v}$ is the number of documents containing both *u* and *v*.

Given our definition of the textual entailment relationship (cf. Section 2.3) for a given word *v* we only consider for entailment words *u* for which $P(Tr_u=1|T_v) > P(Tr_u=1)$ or based on our estimations, for which $n_{u,v}/n_u > n_v/N$ (N is total number of documents in the corpus).

We denote as *tep* the textual entailment probability estimation as derived from (3) and (4) above:

$$tep(t, h) = \prod_{u \in h} \max_{v \in t} lep(u, v) \quad (5)$$

3.2 Baseline model

As a baseline model for comparison, we use a score developed within the context of text summarization. (Monz and de Rijke, 2001) propose modeling the directional entailment between two texts t_1, t_2 via the following score:

$$entscore(t_1, t_2) = \frac{\sum_{w \in (t_1 \cap t_2)} idf(w)}{\sum_{w \in t_2} idf(w)} \quad (6)$$

where $idf(w) = \log(N/n_w)$, N is total number of documents in corpus and n_w is number of docu-

ments containing word *w*. A practically equivalent measure was independently proposed in the context of QA by (Saggion et al., 2004)[3]. This baseline measure captures word overlap, considering only words that appear in both texts and weighs them based on their inverse document frequency.

4 The RTE challenge dataset

The RTE dataset (Dagan et al., 2005) consists of sentence pairs annotated for entailment. Fo this dataset we used word cooccurrence frequencies obtained from a web search engine. The details of this experiment are described in Glickman et al., 2005a. The resulting accuracy on the test set was 59% and the resulting confidence weighted score was 0.57. Both are statistically significantly better than chance at the 0.01 level. The baseline model (6) from Section 3.2, which takes into account only terms appearing in both the text and hypothesis, achieved an accuracy of only 56%. Although our proposed lexical system is relatively simple, as it doesn't rely on syntactic or other deeper analysis, it nevertheless was among the top ranking systems in the RTE Challenge.

5 RCV1 dataset

In addition to the RTE dataset we were interested in evaluating the model on a more representative set of texts and hypotheses that better corresponds to applicative settings. We focused on the information seeking setting, common in applications such as QA and IR, in which a hypothesis is given and it is necessary to identify texts that entail it.

An annotator was asked to choose 60 hypotheses based on sentences from the first few documents in the *Reuters Corpus Volume 1* (Rose et al., 2002). The annotator was instructed to choose sentential hypotheses such that their truth could easily be evaluated. We further required that the hypotheses convey a reasonable information need in such a way that they might correspond to potential questions, semantic queries or IE relations. Table 2 shows a few of the hypotheses.

In order to create a set of candidate entailing texts for the given set of test hypotheses, we followed the common practice of WordNet based expansion (Nie and Brisebois, 1996; Yang and Chua, 2002). Using WordNet, we expanded the hypotheses' terms with morphological alternations and semantically related words[4].

For each hypothesis stop words were removed and all content words were expanded as described above. Boolean Search included a conjunction of the disjunction of the term's expansions and was performed at the paragraph level over the full Reuters corpus, as common in IR for QA. Since we wanted to focus our research on semantic variability we excluded from the result set paragraphs that contain all original words of the hypothesis or their morphological derivations. The resulting dataset consists of 50 hypotheses and over a million retrieved paragraphs (10 hypotheses had only exact matches). The number of paragraphs retrieved per hypothesis range from 1 to 400,000.[5]

5.1 Evaluation

The model's entailment probability, *tep,* was compared to the following two baseline models. The first, denoted as *base*, is the naïve baseline in which all retrieved texts are presumed to entail the hypothesis with equal confidence. This baseline corresponds to systems which perform blind expansion with no weighting. The second baseline, *entscore*, is the entailment score (6) from 3.2.

The top 20 best results for all methods were given to judges to be annotated for entailment. Judges were asked to annotate an example as true if given the text they can infer with high confidence that the hypothesis is true (similar to the guidelines published for the RTE Challenge dataset). Accordingly, they were instructed to annotate the example as false if either they believe the hypothesis is false given the text or if the text is unrelated to the hypothesis. In total there were 1683 text-hypothesis pairs, which were randomly divided between two judges. In order to measure agreement, we had 200 of the pairs annotated by both judges, yielding a moderate agreement (a *Kappa* of 0.6).

[3] (Saggion et al., 2004) actually proposed the above score with no normalizing denominator. However for a given hypothesis it results with the same ranking of candidate entailing texts.

[4] The following WordNet relations were used: *Synonyms*, *see also*, *similar to, hypernyms/hyponyms*, *meronyms/holonyms*, *pertainyms*, *attribute*, *entailment*, *cause* and *domain*

[5] The dataset is available at: http://ir-srv.cs.biu.ac.il:64080/emsee05_dataset.zip

5.2 Results

	base	entscore	tep
precision	0.464	0.568	0.647
cws	0.396	0.509	0.575

Table 2: Results

Table 2 includes the results of macro averaging the precision at top-20 and the average *confidence weighted score* (cws) achieved for the 50 hypotheses. Applying Wilcoxon Signed-Rank Test, our model performs significantly better (at the 0.01 level) than entscore and base for both precision and cws. Analyzing the results showed that many of the mistakes were not due to wrong expansion but rather to a lack of a deeper analysis of the text and hypothesis (e.g. example 3 in Table 2). Indeed this is a common problem with lexical models. Incorporating additional linguistic levels into the probabilistic entailment model, such as syntactic matching, co-reference resolution and word sense disambiguation, becomes a challenging target for future research.

6 Conclusions

This paper proposes a generative probabilistic setting that formalizes the notion of probabilistic textual entailment, which is based on the conditional probability that a hypothesis is true given the text. This probabilistic setting provided the necessary grounding for a concrete probabilistic model of lexical entailment that is based on document co-occurrence statistics in a bag of words representation. Although the illustrated lexical system is relatively simple, as it doesn't rely on syntactic or other deeper analysis, it nevertheless achieved encouraging results. The results suggest that such a probabilistic framework is a promising basis for improved implementations incorporating deeper types of knowledge and a common test-bed for more sophisticated models.

Acknowledgments

This work was supported in part by the IST Programme of the European Community, under the *PASCAL Network of Excellence*, IST-2002-506778. This publication only reflects the authors' views. We would also like to thank Ruthie Mandel and Tal Itzhak Ron for their annotation work.

References

Fahiem Bacchus. 1990. *Representing and Reasoning with Probabilistic Knowledge*, M.I.T. Press.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. *The Mathematics of Statistical Machine Translation: Parameter Estimation*. Computational Linguistics, 19(2):263–311.

Chierchia, Gennaro, and Sally McConnell-Ginet. 2001. *Meaning and grammar: An introduction to semantics*, 2nd. edition. Cambridge, MA: MIT Press.

Ido Dagan, Oren Glickman and Bernardo Magnini. 2005. *The PASCAL Recognising Textual Entailment Challenge*. In Proceedings of the PASCAL Challenges Workshop for Recognizing Textual Entailment. Southampton, U.K.

Oren Glickman, Ido Dagan and Moshe Koppel. 2005a. *Web Based Probabilistic Textual Entailment*, PASCAL Challenges Workshop for Recognizing Textual Entailment.

Oren Glickman, Ido Dagan and Moshe Koppel. 2005b. *A Probabilistic Classification Approach for Lexical Textual Entailment*, Twentieth National Conference on Artificial Intelligence (AAAI-05).

Joseph Y. Halpern. 1990. *An analysis of first-order logics of probability. Artificial Intelligence* 46:311-350.

Christof Monz, Maarten de Rijke. 2001. *Light-Weight Entailment Checking for Computational Semantics*. In Proc. of the third workshop on inference in computational semantics (ICoS-3).

Jian-Yun Nie and Martin Brisebois. 1996. *An Inferential Approach to Information Retrieval and Its Implementation Using a Manual Thesaurus*. Artificial Intelligence Revue 10(5-6): 409-439.

Jay M. Ponte, W. Bruce Croft, 1998. *A Language Modeling Approach to Information Retrieval*. SIGIR conference on Research and Development in Information Retrieval.

Tony G. Rose, Mary Stevenson, and Miles Whitehead. 2002. *The Reuters Corpus volume 1 - from yesterday's news to tomorrow's language resources*. Third International Conference on Language Resources and Evaluation (LREC).

Hui Yang and Tat-Seng Chua. 2002. *The integration of lexical knowledge and external resources for question answering*. The eleventh Text REtrieval Conference (TREC-11).

Generating an Entailment Corpus from News Headlines

John Burger, Lisa Ferro

The MITRE Corporation
202 Burlington Rd.
Bedford, MA 01730, USA
{john,lferro} @ mitre.org

Abstract

We describe our efforts to generate a large (100,000 instance) corpus of textual entailment pairs from the lead paragraph and headline of news articles. We manually inspected a small set of news stories in order to locate the most productive source of entailments, then built an annotation interface for rapid manual evaluation of further exemplars. With this training data we built an SVM-based document classifier, which we used for corpus refinement purposes—we believe that roughly three-quarters of the resulting corpus are genuine entailment pairs. We also discuss the difficulties inherent in manual entailment judgment, and suggest ways to ameliorate some of these.

1 Introduction

MITRE has a long-standing interest in robust text understanding, and, like many, we believe that adequate progress in such an endeavor requires a well-designed evaluation methodology. We have explored in great depth the use of human reading comprehension exams for this purpose (Hirschman et al., 1999, Wellner et al., 2005) as well as TREC-style question answering (Burger, 2004).

In this context, the recent Pascal RTE evaluation (Recognizing Textual Entailment, Dagan et al., 2005) captured our interest. The goal of RTE is to assess systems' abilities at judging semantic entailment with respect to a pair of sentences, e.g.:

- *Fred spilled wine on the carpet.*
- *The rug was wet.*

In RTE parlance, the antecedent sentence is known as the *text*, while the consequent sentence is known as the *hypothesis*. Simply put, the challenge for an RTE system is to judge whether the text entails the hypothesis. Judgments are Boolean, and the primary evaluation metric is simple accuracy, although there were other, secondary metrics used in the evaluation.

The RTE organizers provided 567 exemplar sentence pairs. This is adequate for system development, but not for the application of large-scale statistical models. In particular, we wished to cast the problem as one of statistical alignment as used in machine translation. MT systems typically use millions of sentence pairs, and so we decided to find or generate a much larger corpus. This paper describes our efforts along these lines, as well as some observations about the problems of annotating entailment data. In Section 2 we describe our initial search for an entailment corpus. Section 3 briefly describes an annotation interface we devised, as well as our efforts to refine our corpus. Section 4 explains many of the issues and problems inherent in manual annotation of entailment data.

2 Finding Entailment Data

In our study of the Pascal RTE development corpus, we found that a considerable majority of the TRUE pairs exhibit a stronger relationship than entailment; namely, the hypothesis is a paraphrase of a subset of the text. For instance, given the text

Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, pages 49–54,
Ann Arbor, June 2005.

Source	No. articles examined	No. articles in 1.5 mos.
miami-herald (US)	19	94,278
washington-post (US)	18	13,813
cs-monitor (US)	11	7,102
all-africa	18	68,521
dawn (Pakistan)	17	46,839
gulf-daily-news	10	26,837
national-post (Canada)	18	14,124

Figure 1: MiTAP News Sources Examined

John murdered Bill yesterday, the hypothesis *Bill is dead* is an entailment, while the hypothesis *Bill was killed by John* exhibits the stronger partial paraphrase relationship to the text. We found that 94% (131/140) of the TRUE pairs in the Pascal RTE dev2 corpus were these sorts of paraphrases.

In our search for an entailment corpus, we observed that the headline of a news article is often a partial paraphrase of the lead paragraph, much like the RTE data, or is sometimes a genuine entailment. We thus deduced that headlines and their corresponding lead paragraphs might provide a readily available source of training data. As an initial test of this hypothesis, we manually inspected over 200 news stories from 11 different sources. We found a great deal of variety in headline formats, and ultimately found the Xinhua News Agency English Service articles from the Gigaword corpus (Graff, 2003) to be the richest source, though somewhat limited in subject domain. We describe here our data collection and analysis process.

Because our goal was to automatically generate an extremely large corpus of exemplars, we focused on large data sources. We first examined 111 news stories culled from MiTAP (Damianos et al., 2003), which collects over one million articles per month from approximately 75 different sources. By first counting the number of articles typically collected for each source, we selected a mixture of sources that each had more than 10,000 articles for our sample period of one and half months. As discussed further below, part way through our investigation it became clear that we needed to include more native English sources, so the *Christian Science Monitor* articles were added, though they fell below our arbitrary 10K mark. Figure 1 summarizes the MiTAP news sources examined.

	Yes	**No**	**Maybe**	**Total**
All Pairs	54 (49%)	39 (35%)	18 (16%)	111
Filtered	54 (53%)	33 (33%)	14 (14%)	101

Figure 2: MiTAP Corpus Results

For each lead paragraph/headline pair, a human rendered a judgment of *yes*, *no,* or *maybe* as to whether the lead paragraph entailed the headline, where *maybe* meant that the headline was very close to being an entailment or paraphrase. This is likely equivalent to the notion of "more or less semantically equivalent" used in the Microsoft Research Paraphrase Corpus (Dolan et al., 2005). The purpose of *maybe* in this case was that we thought that many of the near-miss pairs would make adequate training data for statistical algorithms, in spite of being less than perfect.

There were many types of news articles in the MiTAP data that did not yield good headline/lead paragraph pairs for our purposes. Many would be difficult to filter out using automated heuristics. Two frequent examples of this were opinion-ed itorial pieces and daily Wall Street summaries. Others would be more amenable to automatic elimination, including obituaries and collections of news snippets like the *Washington Post*'s "World in Brief". Articles consisting of personal narratives never yielded good headlines, but these could easily be eliminated by recognizing first person pronouns in the lead paragraph. Figure 2 shows the judgments for all the MiTAP articles examined, where the Filtered row excludes these easily eliminated article types.

As Figure 2 shows, the MiTAP data did not yield a high percentage of good pairs. In addition, whether due to poor machine translation or English dialectal differences, our evaluator found it difficult to understand some of the text from sources that were not English-primary. A certain amount of ill-formed text was acceptable, since the Pascal RTE challenge included training and test data drawn from MT scenarios, but we did not wish our data to be too dominated by such sources. Thus, we selected additional native-English articles to add to our sample set.

Despite the overall poor yield from this data, it

Source	**Yes**	**No**	**Maybe**	**Total**
APW	8 (31%)	12 (46%)	6 (23%)	26
AFE	14 (56%)	4 (16%)	7 (28%)	25
NYT	8 (31%)	17 (65%)	1 (4%)	26
XIE	22 (85%)	4 (15%)	0 (0%)	26
Total	52 (50%)	37 (36%)	14 (14%)	103

Figure 3: Gigaword Corpus Results

was apparent that some news sources tended to be more fruitful than others. For example, 13 out of 18 of the *Washington Post* articles yielded good pairs, as opposed to only 1 of the 11 *Christian Science Monitor* articles.

This generalization was likewise true in the second corpus we examined, the Gigaword newswire corpus (Graff, 2003). Gigaword contains over 4 million documents from four news sources:

- Agence France Press English Service (AFE)
- Associated Press Worldstream English Service (APW)
- The New York Times Newswire Service (NYT)
- The Xinhua News Agency English Service (XIE)

For each source, Gigaword articles are classified into several types, including newswire advisories, etc. We restricted our investigations to actual news stories. As Figure 3 shows, overall results were much the same as the MiTAP articles, but 85% of the XIE articles yielded adequate pairs.

Based on these preliminary results we decided to focus further manual investigations on the XIE articles from Gigaword. We also decided to expend some effort on an annotation tool that would allow us to proceed more quickly than the early annotation experiments described above.

3 Refining the Data

MITRE has developed a series of annotation tools for a variety of linguistic phenomena (Day et al, 1997; Day et al, 2004), but these are primarily designed for fine-grained tasks such as named entity and syntactic annotation. For our headline corpus, we wanted the ability to rapidly annotate at a document level from a small set of categories. Further, we wanted the interface to easily support distributed annotation efforts.

The resulting annotation interface is shown in Figure 4. It is web-based, and annotations and other document information are stored in an SQL database. The document to be evaluated is displayed in the user's chosen browser, with the XML

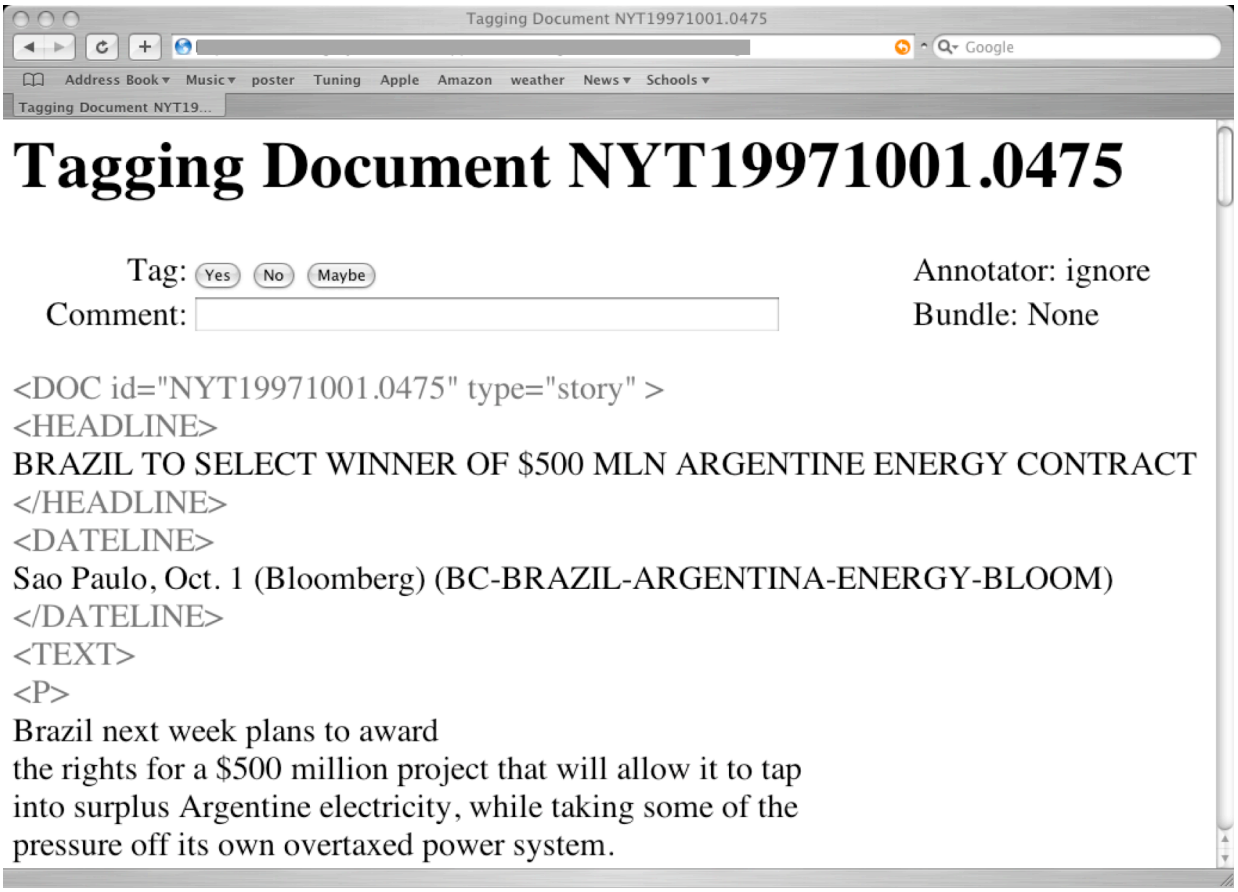

Figure 4: Entailment Tagging Interface

document zoning tags visible so that the user can easily identify the headline and lead paragraph. At the top of the document are three buttons from which to select a *yes/no/maybe* judgment. The user can also add a comment before moving to the next document. Typically several documents can be judged per minute. The client-server architecture supports multiple annotations of the same document by different annotators—accordingly, it has a mode enabling reconciliation of inter-annotator disagreements. All further annotation efforts discussed below were carried out with this tool.

Using the tool, we tagged approximately 900 randomly chosen Gigaword documents, including 520 XIE documents. From this, we estimate that 70% of the XIE headlines in Gigaword are entailed by the corresponding lead paragraph. (This is lower than the rough estimate described in Section 2, but that was based on a very small sample.) We decided to explore ways to refine the data in order to arrive at a smaller, but less noisy subcorpus. We observed that different subgenres within the newspaper corpus evinced the lead-entails-headline quality to different degrees. For example, articles about sports or entertainment often had whimsical (non-entailed) headlines, while articles about politics or business more frequently had the headline quality we sought.

Accordingly, we decided to treat the data refinement process as a text classification problem, one of finding the mix of genres or topics that would most likely possess the lead-entails-headline quality. We used SVM-light (Joachims, 2002) as a document classifier, training it on the initial set of annotated articles. (Note that these text classification experiments made use of the entire article, not just the lead and headline.) We experimented with a variety of feature representations and SVM parameters, but found the best performance with a Boolean bag-of-words representation, and a simple linear kernel. Leave-one-out estimates indicate that SVM-light could identify documents with the requisite entailment quality with 77% accuracy.

We performed one round of active learning (Tong & Koller, 2000), in which we used SVM-light to classify a large subset of the unannotated corpus, and then selected a 100-document subset about which the classifier was least certain. The rationale is that annotating these uncertain documents will be more informative to further learning runs than a randomly selected subset. In the case of large-margin classifiers like SVMs, the natural choice is to select the instances closest to the margin. These were then annotated, and added back to the training data for the next learning run. However, leave-one-out estimates indicated that the classifier benefited little from these new instances.

As described above, we estimate that the base rate of the headline entailment property in the XIE portion of Gigaword is 70%. Our hypothesis in training the SVM was that we could identify a smaller but less noisy subset. In order to evaluate this, we ran the trained SVM on all 679,000 of the unannotated XIE documents, and selected the 100,000 "best" instances—that is, the documents most likely (according to the SVM) to evince the headline quality. We selected a random subset of these best documents, and annotated them to evaluate our hypothesis. 74% of these possessed the lead-entails-headline property, a difference of 4% absolute over the XIE base rate. We used the lead-headline pairs from this 100,000-best subset to train our MT-alignment-based system for the RTE evaluation (Bayer et al., 2005). This system was one of the best performers in the evaluation, which we ascribe to our large training corpus

Later examination showed that the 4% "improvement" in purity is not statistically significant. We intend to perform further experiments in data refinement, but this may prove unnecessary. Perhaps the base rate of the entailment phenomenon in the XIE documents is sufficient to train an effective alignment-based entailment system. In this case, *all* of the XIE documents could be used, perhaps resulting in a more robust, and even better performing system.

4 Judging Headline Entailments

In the process of generating the training data, we doubly-judged an additional 300 XIE documents to measure inter-judge reliability. As in the pilot phase described above, each pair was labeled as *yes*, *no*, or *maybe*. In addition, the judges were given a comment field to record their reasoning and misgivings. The judging was performed in two steps, first on a set of 100 documents and then on a set of 200. One of the judges was already well versed in the RTE task, and had performed the earlier pilot investigations. Prior to judging the first set, the second judge was given a brief verbal

Condition	Set 1 (100 docs)	Set 2 (200 docs)
strict match	75.00%	77.50%
maybe = yes	79.00%	90.00%
maybe = no	84.00%	81.00%
maybe = *	88.00%	94.00%

Figure 5: Agreement for Two XIE Data Sets

overview of the task. After the first 100 documents had been doubly-judged, the more experienced judge then reviewed the differences and drafted a set of guidelines. The guidelines provided a synopsis of the official RTE guidelines, plus a few rules unique to headlines. For example, one rule specified what to do when partial entailment only held if the lead were combined with location or date information from the dateline. The two evaluators then judged the second set. The results for both sets are shown in Figure 5.

As these results show, the guidelines had only a small effect on the strict measure of agreement. Three problem areas existed:

(1) Raw, messy data. The Gigaword corpus was automatically collected and zoned. Thus, the headlines in particular contained a number of irregularities that made it difficult to judge their appropriateness. Such irregularities included truncations, phrases lacking any proposition, prepended alerts like *URGENT:,* and bylines and date lines miszoned into the headline.

(2) Disagreement on what constitutes synonymy. Our judges found they had irreconcilable differences about differences in meaning. For example, in the following pair, the judges disagreed about whether *safe operation* in the lead paragraph meant the same thing as, and thus entailed, *operates smoothly* in the headline:

- *Shanghai's Hongqiao Airport Operates Smoothly*
- *As of Saturday, Shanghai's Hongqiao Airport has performed safe operation for some 2,600 consecutive days, setting a record in the country.*

(3) Disagreement on the amount of world knowledge permitted. Figure 5 shows that if *maybe* is counted as equivalent to *yes*, the agreement level improves significantly. This is likely because there were two important aspects of the RTE definition of entailment that were not imparted to the second judge until the written guidelines: that one can assume "common human understanding of language and some common background knowledge." However, our judges did not always agree on what counts as "common," which accounts for much of the high overlap between *yes* and *maybe*. Nevertheless, our 90% agreement compares favorably to the 83% agreement rate reported by Dolan et al. (2005) for their judgments on "more or less semantically equivalent" pairs. Our 78% strict agreement compares favorably to the 80% agreement achieved by Dagan et al. (2005), given that our data was messier than the pairs crafted for the RTE challenge.

Like Dagan et al. (2005), we did not force resolution on all disagreements. Disagreements over synonymy and common knowledge result in irreconcilable differences, because it is neither possible nor desirable to use guidelines to force a shared understanding of an utterance. Thus, for the first set of data 15 (15%) of the pairs were left unreconciled. In the second set, 42 (21%) were left unreconciled. Eleven (6%) of the irreconcilable pairs in the second set were due to confusion stemming from the telegraphic nature of headlines, which led to misunderstandings about how to judge truncated headlines (*Chinese President Vows to Open New Chapters With*) vs. headlines lacking propositions (subject headings like *Mandela's Speech*) vs. well-formed but terse headlines (*Crackdown on Auto-Mafia in Bulgaria*).

Despite the high number of irreconcilable pairs, one encouraging sign was evident from the comment field. The judges' comments revealed that on pairs where they disagreed on how to label the pair, they often agreed on what the problem was.

Our experience in generating a training corpus, particularly the number of irreconcilable cases we encountered, raises an important issue, namely, the feasibility of semantic equivalence tasks. We suggest that the optimum method for empirically modeling semantic equivalence is to capture the variation in human judgments. Three judges would evaluate each pair, so that there would always be a tie breaker. After reconciling for disagreements arising from human error, each distinct judgment would become part of the data set. We also recommend that where there is genuine disagreement, the questionable portions of each pair be annotated in some way to capture the source of the problem, going one step further than the comment field we found beneficial in our annotation interface. The three judgments would result in a four way classification of pairs:

TTT = TRUE
TTF = Likely TRUE, but possibly FALSE
TFF = Likely FALSE, but possibly TRUE
FFF = FALSE

System developers could choose to train on all the data, or limit themselves to the TTT/FFF cases. For evaluation purposes, the systems' results on the TTF/TFF pairs could be evaluated in light of the human variation, providing a more realistic measure of the complexity of the task.

5 Conclusion

Given the number of natural language processing applications that require the ability to recognize semantic equivalence and entailment, there is an obvious need for both robust evaluation methodologies and adequate development and test data. We've described here our work in generating supplemental training data for the recent Pascal RTE evaluation, with which we produced a competitive system. Some news corpora provide a rich source of exemplars, and an automatic document classifier can be used to reduce the noisiness of the data. There are lingering difficulties in achieving high inter-judge agreement in determining paraphrase and entailment, and we believe the best way to cope with this is to allow the data to reflect the variance that exists in cross-human judgments.

Acknowledgments

This paper reports on work supported by the MITRE Sponsored Research Program. We would also like to extend our thanks to Sam Bayer, John Henderson and Alex Yeh for their invaluable suggestions and comments. Our gratitude also goes to Laurie Damianos, who provided us with statistics on MiTAP's resources and served as one of the evaluators in our inter-judge reliability study.

References

Samuel Bayer, John Burger, Lisa Ferro, John Henderson, and Alexander Yeh, 2005. MITRE's submissions to the EU Pascal RTE Challenge. *PASCAL Proceedings of the First Challenge Workshop, Recognizing Textual Entailment*, 11–13 April, 2005, Southampton, U.K.

John D. Burger, 2004. MITRE's Qanda at TREC-12. *The Twelvth Text REtrieval Conference*. NIST Special Publication SP 500–255.

Ido Dagan, Oren Glickman, and Bernado Magnini, 2005. The PASCAL recognizing textual entailment challenge. *PASCAL Proceedings of the First Challenge Workshop, Recognizing Textual Entailment*, 11–13 April, 2005, Southampton, U.K.

Laurie Damianos, Steve Wohlever, Robyn Kozierok, and Jay Ponte, 2003. mitap for real users, real data, real problems. In *Proceedings of the Conference on Human Factors of Computing Systems* (CHI 2003), Fort Lauderdale, FL April 5–10.

David Day, John Aberdeen, Lynette Hirschman, Robyn Kozierok, Patricia Robinson and Marc Vilain, 1997. Mixed initiative development of language processing systems. *Proceedings of the Fifth Conference on Applied Natural Language Processing*.

David Day, Chad McHenry, Robyn Kozierok, Laurel Riek, 2004. Callisto: A configurable annotation workbench. *International Conference on Language Resources and Evaluation*.

Bill Dolan, Chris Brockett., and Chris Quirk, 2005. *Microsoft Research Paraphrase Corpus*. http://research.microsoft.com/research/nlp/msr_paraphrase.htm

David Graff, 2003. *English Gigaword*. http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2003T05

Dan Gusfield, 1997. *Algorithms on Strings, Trees and Sequences*. Cambridge University Press.

Lynette Hirschman, Marc Light, Eric Breck, and John D. Burger, 1999. Deep Read: A reading comprehension system. *Proceedings of the 37th annual meeting of the Association for Computational Linguistics*.

Thorsten Joachims, 2002. *Learning to Classify Text Using Support Vector Machines*. Kluwer.

Guido Minnen, John Carroll, and Darren Pearce, 2001. Applied morphological processing of English. *Natural Language Engineering,* 7(3).

Franz Josef Och and Hermann Ney, 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1).

Simon Tong and Daphne Koller, 2000. support vector machine active learning with applications to text classification. *Proceedings of ICML-00, 17th International Conference on Machine Learning*.

Ben Wellner, Lisa Ferro, Warren Greiff, and Lynette Hirschman, 2005. Reading comprehension tests for computer-based understanding evaluation. *Natural Language Engineering* (to appear).

Definition and Analysis of Intermediate Entailment Levels

Roy Bar-Haim, Idan Szpektor, Oren Glickman
Computer Science Department
Bar Ilan University
Ramat-Gan 52900, Israel
{barhair,szpekti,glikmao}@cs.biu.ac.il

Abstract

In this paper we define two intermediate models of textual entailment, which correspond to lexical and lexical-syntactic levels of representation. We manually annotated a sample from the RTE dataset according to each model, compared the outcome for the two models, and explored how well they approximate the notion of entailment. We show that the lexical-syntactic model outperforms the lexical model, mainly due to a much lower rate of false-positives, but both models fail to achieve high recall. Our analysis also shows that *paraphrases* stand out as a dominant contributor to the entailment task. We suggest that our models and annotation methods can serve as an evaluation scheme for entailment at these levels.

1 Introduction

Textual entailment has been proposed recently as a generic framework for modeling semantic variability in many Natural Language Processing applications, such as Question Answering, Information Extraction, Information Retrieval and Document Summarization. The textual entailment relationship holds between two text fragments, termed text and hypothesis, if the truth of the hypothesis can be inferred from the text.

Identifying entailment is a complex task that incorporates many levels of linguistic knowledge and inference. The complexity of modeling entailment was demonstrated in the first PASCAL Challenge Workshop on Recognizing Textual Entailment (RTE) (Dagan et al., 2005). Systems that participated in the challenge used various combinations of NLP components in order to perform entailment inferences. These components can largely be classified as operating at the lexical, syntactic and semantic levels (see Table 1 in (Dagan et al., 2005)). However, only little research was done to analyze the contribution of each inference level, and on the contribution of individual inference mechanisms within each level.

This paper suggests that decomposing the complex task of entailment into subtasks, and analyzing the contribution of individual NLP components for these subtasks would make a step towards better understanding of the problem, and for pursuing better entailment engines. We set three goals in this paper. First, we consider two modeling levels that employ only part of the inference mechanisms, but perform perfectly at each level. We explore how well these models approximate the notion of entailment, and analyze the differences between the outcome of the different levels. Second, for each of the presented levels, we evaluate the distribution (and contribution) of each of the inference mechanisms typically associated with that level. Finally, we suggest that the definitions of entailment at different levels of inference, as proposed in this paper, can serve as guidelines for manual annotation of a "gold standard" for evaluating systems that operate at a particular level. Altogether, we set forth a possible methodology for annotation and analysis of entail-

Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, pages 55–60,
Ann Arbor, June 2005.

ment datasets.

We introduce two levels of entailment: *Lexical* and *Lexical-Syntactic*. We propose these levels as intermediate stages towards a complete entailment model. We define an entailment model for each level and manually evaluate its performance over a sample from the RTE test-set. We focus on these two levels as they correspond to well-studied NLP tasks, for which robust tools and resources exist, e.g. parsers, part of speech taggers and lexicons. At each level we included inference types that represent common practice in the field. More advanced processing levels which involve logical/semantic inference are less mature and were left beyond the scope of this paper.

We found that the main difference between the lexical and lexical-syntactic levels is that the lexical-syntactic level corrects many false-positive inferences done at the lexical level, while introducing only a few false-positives of its own. As for identifying positive cases (recall), both systems exhibit similar performance, and were found to be complementary. Neither of the levels was able to identify more than half of the positive cases, which emphasizes the need for deeper levels of analysis. Among the different inference components, *paraphrases* stand out as a dominant contributor to the entailment task, while synonyms and derivational transformations were found to be the most frequent at the lexical level.

Using our definitions of entailment models as guidelines for manual annotation resulted in a high level of agreement between two annotators, suggesting that the proposed models are well-defined.

Our study follows on previous work (Vanderwende et al., 2005), which analyzed the RTE Challenge test-set to find the percentage of cases in which syntactic analysis alone (with optional use of thesaurus for the lexical level) suffices to decide whether or not entailment holds. Our study extends this work by considering a broader range of inference levels and inference mechanisms and providing a more detailed view. A fundamental difference between the two works is that while Vanderwende et al. did not make judgements on cases where additional knowledge was required beyond syntax, our entailment models were evaluated over all of the cases, including those that require higher levels of inference. This allows us to view the entailment model at each level as an idealized *system* approximating full entailment, and to evaluate its overall success.

The rest of the paper is organized as follows: section 2 provides definitions for the two entailment levels; section 3 describes the annotation experiment we performed, its results and analysis; section 4 concludes and presents planned future work.

2 Definition of Entailment Levels

In this section we present definitions for two entailment models that correspond to the *Lexical* and *Lexical-Syntactic* levels. For each level we describe the available inference mechanisms. Table 1 presents several examples from the RTE test-set together with annotation of entailment at the different levels.

2.1 The Lexical entailment level

At the lexical level we assume that the text T and hypothesis H are represented by a bag of (possibly multi-word) terms, ignoring function words. At this level we define that entailment holds between T and H if every term h in H can be matched by a corresponding entailing term t in T. t is considered as entailing h if either h and t share the same lemma and part of speech, or t can be matched with h through a sequence of lexical transformations of the types described below.

Morphological derivations This inference mechanism considers two terms as equivalent if one can be obtained from the other by some morphological derivation. Examples include nominalizations (e.g. 'acquisition $\Leftrightarrow$ acquire'), pertainyms (e.g. 'Afghanistan $\Leftrightarrow$ Afghan'), or nominal derivations like 'terrorist $\Leftrightarrow$ terror'.

Ontological relations This inference mechanism refers to ontological relations between terms. A term is inferred from another term if a chain of valid ontological relations between the two terms exists (Andreevskaia et al., 2005). In our experiment we regarded the following three ontological relations as providing entailment inferences: (1) 'synonyms' (e.g. 'free $\Leftrightarrow$ release' in example 1361, Table 1); (2) 'hypernym' (e.g. 'produce $\Rightarrow$ make') and (3) 'meronym-holonym' (e.g. 'executive $\Rightarrow$ company').

No.	Text	Hypothesis	Task	Ent.	Lex. Ent.	Syn. Ent.
322	Turnout for the historic vote for the first time since the EU took in 10 new members in May has hit a record low of 45.3%.	New members joined the EU.	IR	true	false	true
1361	A Filipino hostage in Iraq was released.	A Filipino hostage was freed in Iraq.	CD	true	true	true
1584	Although a Roscommon man by birth, born in Rooskey in 1932, Albert "The Slasher" Reynolds will forever be a Longford man by association.	Albert Reynolds was born in Co. Roscommon.	QA	true	true	true
1911	The SPD got just 21.5% of the vote in the European Parliament elections, while the conservative opposition parties polled 44.5%.	The SPD is defeated by the opposition parties.	IE	true	false	false
2127	Coyote shot after biting girl in Vanier Park.	Girl shot in park.	IR	false	true	false

Table 1: Examples of text-hypothesis pairs, taken from the PASCAL RTE test-set. Each line includes the example number at the RTE test-set, the text and hypothesis, the task within the test-set, whether entailment holds between the text and hypothesis (Ent.), whether Lexical entailment holds (Lex. Ent.) and whether Lexical-Syntactic entailment holds (Syn. Ent.).

Lexical World knowledge This inference mechanism refers to world knowledge reflected at the lexical level, by which the meaning of one term can be inferred from the other. It includes both knowledge about named entities, such as 'Taliban $\Rightarrow$ organization' and 'Roscommon $\Leftrightarrow$ Co. Roscommon' (example 1584 in Table 1), and other lexical relations between words, such as WordNet's relations 'cause' (e.g. 'kill $\Rightarrow$ die') and 'entail' (e.g. 'snore $\Rightarrow$ sleep').

2.2 The Lexical-syntactic entailment level

At the lexical-syntactic level we assume that the text and the hypothesis are represented by the set of syntactic dependency relations of their dependency parse. At this level we ignore determiners and auxiliary verbs, but do include relations involving other function words. We define that entailment holds between T and H if the relations within H can be "covered" by the relations in T. In the trivial case, lexical-syntactic entailment holds if all the relations composing H appear verbatim in T (while additional relations within T are allowed). Otherwise, such coverage can be obtained by a sequence of transformations applied to the relations in T, which should yield all the relations in H.

One type of such transformations are the lexical transformations, which replace corresponding lexical items, as described in sub-section 2.1. When applying morphological derivations it is assumed that the syntactic structure is appropriately adjusted. For example, "Mexico produces oil" can be mapped to "oil production by Mexico" (the NOMLEX resource (Macleod et al., 1998) provides a good example for systematic specification of such transformations).

Additional types of transformations at this level are specified below.

Syntactic transformations This inference mechanism refers to transformations between syntactic structures that involve the same lexical elements and preserve the meaning of the relationships between them (as analyzed in (Vanderwende et al., 2005)). Typical transformations include passive-active and apposition (e.g. 'An Wang, a native of Shanghai $\Leftrightarrow$ An Wang is a native of Shanghai').

Entailment paraphrases This inference mechanism refers to transformations that modify the syntactic structure of a text fragment as well as some of its lexical elements, while holding an entailment relationship between the original text and the transformed one. Such transformations are typically denoted as 'paraphrases' in the literature, where a wealth of methods for their automatic acquisition were proposed (Lin and Pantel, 2001; Shinyama et al., 2002; Barzilay and Lee, 2003; Szpektor et al., 2004). Following the same spirit, we focus here on transformations that are local in nature, which, according to the literature, may be amenable for large scale acquisition. Examples include: 'X is Y man by birth $\rightarrow$ X was born in Y' (example 1584 in Table 1), 'X take in Y $\Leftrightarrow$ Y join X'[1] and 'X is holy book of Y $\Rightarrow$ Y follow X'[2].

Co-reference Co-references provide equivalence relations between different terms in the text and thus induce transformations that replace one term in a text with any of its co-referenced terms. For example, the sentence "Italy and Germany have each played twice, and they haven't beaten anybody yet."[3] entails "Neither Italy nor Germany have won yet", involving the co-reference transformation 'they $\Rightarrow$ Italy and Germany'.

Example 1584 in Table 1 demonstrates the need to combine different inference mechanisms to achieve lexical-syntactic entailment, requiring world-knowledge, paraphrases and syntactic transformations.

3 Empirical Analysis

In this section we present the experiment that we conducted in order to analyze the two entailment levels, which are presented in section 2, in terms of relative performance and correlation with the notion of textual entailment.

3.1 Data and annotation procedure

The RTE test-set[4] contains 800 Text-Hypothesis pairs (usually single sentences), which are typical to various NLP applications. Each pair is annotated with a boolean value, indicating whether the hypothesis is entailed by the text or not, and the test-set is balanced in terms of positive and negative cases. We shall henceforth refer to this annotation as the *gold standard*. We constructed a sample of 240 pairs from four different tasks in the test-set, which correspond to the main applications that may benefit from entailment: information extraction (IE), information retrieval (IR), question answering (QA), and comparable documents (CD). We randomly picked 60 pairs from each task, and in total 118 of the cases were positive and 122 were negative.

In our experiment, two of the authors annotated, for each of the two levels, whether or not entailment can be established in each of the 240 pairs. The annotators agreed on 89.6% of the cases at the lexical level, and 88.8% of the cases at the lexical-syntactic level, with Kappa statistics of 0.78 and 0.73, respectively, corresponding to 'substantial agreement' (Landis and Koch, 1977). This relatively high level of agreement suggests that the notion of lexical and lexical-syntactic entailment we propose are indeed well-defined.

Finally, in order to establish statistics from the annotations, the annotators discussed all the examples they disagreed on and produced a final joint decision.

3.2 Evaluating the different levels of entailment

	L	LS
True positive (118)	52	59
False positive (122)	36	10
Recall	44%	50%
Precision	59%	86%
F_1	0.5	0.63
Accuracy	58%	71%

Table 2: Results per level of entailment.

Table 2 summarizes the results obtained from our annotated dataset for both lexical (L) and lexical-syntactic (LS) levels. Taking a "system"-oriented perspective, the annotations at each level can be viewed as the classifications made by an idealized system that includes a perfect implementation of the inference mechanisms in that level. The first two

[1] Example no 322 in the PASCAL RTE test-set.

[2] Example no 1575 in the PASCAL RTE test-set.

[3] Example no 298 in the PASCAL RTE test-set.

[4] The complete RTE dataset can be obtained at http://www.pascal-network.org/Challenges/RTE/Datasets/

rows show for each level how the cases, which were recognized as positive by this level (i.e. the entailment holds), are distributed between "true positive" (i.e. positive according to the gold standard) and "false positive" (negative according to the gold standard). The total number of positive and negative pairs in the dataset is reported in parentheses. The rest of the table details recall, precision, F_1 and accuracy.

The distribution of the examples in the RTE test-set cannot be considered representative of a real-world distribution (especially because of the controlled balance between positive and negative examples). Thus, our statistics are not appropriate for accurate prediction of application performance. Instead, we analyze how well these simplified models of entailment succeed in approximating "real" entailment, and how they compare with each other.

The proportion between true and false positive cases at the lexical level indicates that the correlation between lexical match and entailment is quite low, reflected in the low precision achieved by this level (only 59%). This result can be partly attributed to the idiosyncracies of the RTE test-set: as reported in (Dagan et al., 2005), samples with high lexical match were found to be biased towards the negative side. Interestingly, our measured accuracy correlates well with the performance of systems at the PASCAL RTE Workshop, where the highest reported accuracy of a lexical system is 0.586 (Dagan et al., 2005).

As one can expect, adding syntax considerably reduces the number of false positives - from 36 to only 10. Surprisingly, at the same time the number of true positive cases grows from 52 to 59, and correspondingly, precision rise to 86%. Interestingly, neither the lexical nor the lexical-syntactic level are able to cover more than half of the positive cases (e.g. example 1911 in Table 1).

In order to better understand the differences between the two levels, we next analyze the overlap between them, presented in Table 3. Looking at Table 3(a), which contains only the positive cases, we see that many examples were recognized only by one of the levels. This interesting phenomenon can be explained on the one hand by lexical matches that could not be validated in the syntactic level, and on the other hand by the use of paraphrases, which are introduced only in the lexical-syntactic level. (e.g. example 322 in Table 1).

		Lexical-Syntactic	
		H $\Rightarrow$ T	H $\not\Rightarrow$ T
Lexical	H $\Rightarrow$ T	38	14
	H $\not\Rightarrow$ T	21	45

(a) positive examples

		Lexical-Syntactic	
		H $\Rightarrow$ T	H $\not\Rightarrow$ T
Lexical	H $\Rightarrow$ T	7	29
	H $\not\Rightarrow$ T	3	83

(b) negative examples

Table 3: Correlation between the entailment levels. (a) includes only the positive examples from the RTE dataset sample, and (b) includes only the negative examples.

This relatively symmetric situation changes as we move to the negative cases, as shown in Table 3(b). By adding syntactic constraints, the lexical-syntactic level was able to fix 29 false positive errors, misclassified at the lexical level (as demonstrated in example 2127, Table 1), while introducing only 3 new false-positive errors. This exemplifies the importance of syntactic matching for precision.

3.3 The contribution of various inference mechanisms

Inference Mechanism	f	$\triangle$R	%
Synonym	19	14.4%	16.1%
Morphological	16	10.1%	13.5%
Lexical World knowledge	12	8.4%	10.1%
Hypernym	7	4.2%	5.9%
Mernoym	1	0.8%	0.8%
Entailment Paraphrases	37	26.2%	31.3%
Syntactic transformations	22	16.9%	18.6%
Coreference	10	5.0%	8.4%

Table 4: The frequency (f), contribution to recall ($\triangle$R) and percentage (%), within the gold standard positive examples, of the various inference mechanisms at each level, ordered by their significance.

In order to get a sense of the contribution of the various components at each level, statistics on the inference mechanisms that contributed to the coverage of the hypothesis by the text (either full or partial) were recorded by one annotator. Only the positive cases in the gold standard were considered.

For each inference mechanism we measured its frequency, its contribution to the recall of the related level and the percentage of cases in which it is required for establishing entailment. The latter also takes into account cases where only partial coverage could be achieved, and thus indicates the significance of each inference mechanism for any entailment system, regardless of the models presented in this paper. The results are summarized in Table 4.

From Table 4 it stands that paraphrases are the most notable contributors to recall. This result indicates the importance of paraphrases to the entailment task and the need for large-scale paraphrase collections. Syntactic transformations are also shown to contribute considerably, indicating the need for collections of syntactic transformations as well. In that perspective, we propose our annotation framework as means for evaluating collections of paraphrases or syntactic transformations in terms of recall.

Finally, we note that the co-reference moderate contribution can be partly attributed to the idiosyncracies of the RTE test-set: the annotators were guided to replace anaphors with the appropriate reference, as reported in (Dagan et al., 2005).

4 Conclusions

In this paper we presented the definition of two entailment models, Lexical and Lexical-Syntactic, and analyzed their performance manually. Our experiment shows that the lexical-syntactic level outperforms the lexical level in all measured aspects. Furthermore, paraphrases and syntactic transformations emerged as the main contributors to recall. These results suggest that a lexical-syntactic framework is a promising step towards a complete entailment model.

Beyond these empirical findings we suggest that the presented methodology can be used generically to annotate and analyze entailment datasets.

In future work, it would be interesting to analyze higher levels of entailment, such as logical inference and deep semantic understanding of the text.

Acknowledgements

We would like to thank Ido Dagan for helpful discussions and for his scientific supervision. This work was supported in part by the IST Programme of the European Community, under the *PASCAL Network of Excellence*, IST-2002-506778. This publication only reflects the authors' views.

References

Alina Andreevskaia, Zhuoyan Li and Sabine Bergler. 2005. Can Shallow Predicate Argument Structures Determine Entailment?. In *Proceedings of Pascal Challenge Workshop on Recognizing Textual Entailment, 2005*.

Regina Barzilay and Lillian Lee. 2003. Learning to paraphrase: An unsupervised approach using multiple-sequence alignment. In *Proceedings of HLT-NAACL 2003*. pages 16-23, Edmonton, Canada.

Ido Dagan, Bernardo Magnini and Oren Glickman. 2005. The PASCAL Recognising Textual Entailment Challenge. In *Proceedings of Pascal Challenge Workshop on Recognizing Textual Entailment, 2005*.

J.R. Landis and G.G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33:159-174.

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

C. Macleod, R. Grishman, A. Meyers, L. Barrett and R. Reeves. 1998. Nomlex: A lexicon of nominalizations. In *Proceedings of the 8th International Congress of the European Association for Lexicography, 1998*. Liège, Belgium: EURALEX.

Yusuke Shinyama and Satoshi Sekine, Kiyoshi Sudo and Ralph Grishman. 2002. Automatic paraphrase acquisition from news articles. In *Proceedings of Human Language Technology Conference (HLT 2002)*. San Diego, USA.

Idan Szpektor, Hristo Tanev, Ido Dagan and Bonnaventura Coppola. 2004. Scaling Web-based Acquistion of Entailment Relations. In *Proceedings of EMNLP 2004*.

Lucy Vanderwende, Deborah Coughlin and Bill Dolan. 2005. What Syntax Contribute in Entailment Task. In *Proceedings of Pascal Challenge Workshop on Recognizing Textual Entailment, 2005*.

Author Index