FBK: Machine Translation Evaluation and Word Similarity metrics for Semantic Textual Similarity

José Guilherme C. de Souza
Fondazione Bruno Kessler
University of Trento
Povo, Trento, Italy
desouza@fbk.eu

Matteo Negri
Fondazione Bruno Kessler
Povo, Trento
Italy
negri@fbk.eu

Yashar Mehdad
Fondazione Bruno Kessler
Povo, Trento
Italy
mehdad@fbk.eu

Abstract

This paper describes the participation of FBK in the Semantic Textual Similarity (STS) task organized within Semeval 2012. Our approach explores lexical, syntactic and semantic machine translation evaluation metrics combined with distributional and knowledge-based word similarity metrics. Our best model achieves 60.77% correlation with human judgements (*Mean* score) and ranked 20 out of 88 submitted runs in the *Mean* ranking, where the average correlation across all the sub-portions of the test set is considered.

1 Introduction

The Semantic Textual Similarity (STS) task proposed at SemEval 2012 consists of examining the degree of semantic equivalence between two sentences and assigning a score to quantify such similarity ranging from 0 (the two texts are about different topics) to 5 (the two texts are semantically equivalent). The complete description of the task, the datasets and the evaluation methodology adopted can be found in (Agirre et al., 2012).

Typical approaches to measure semantic textual similarity exploit information at the lexical level. The proposed solutions range from calculating the overlap of common words between the two text segments (Salton et al., 1997) to the application of knowledge-based and corpus-based word similarity metrics to cope with the low recall achieved by on simple lexical matching (Mihalcea et al., 2006).

Our participation in the STS task is inspired by previous work on paraphrase recognition, in which machine translation (MT) evaluation metrics are used to identify whether a pair of sentences are semantically equivalent or not (Finch and Hwang, 2005; Wan et al., 2006). Our approach to semantic textual similarity makes use of not only lexical information but also syntactic and semantic information. To this aim, our metrics are based on different natural language processing tools that provide syntactic and semantic annotation. These include shallow parsing, constituency parsing, dependency parsing, semantic roles labeling, discourse representation analyzer, and named entities recognition. In addition, we employed distributional and knowledge-based word similarity metrics in an attempt to improve the results given by the MT metrics. The computed scores are used as features to train a regression model in a supervised learning framework.

Our best run model achieves 60.77% correlation with human judgements when evaluating the semantic similarity of texts from the entire test set and was ranked in the 20th position (out of 88 submitted runs) in the *Mean* ranking.

2 System Description

The system has been designed following a machine learning based approach in which a regression model is induced using different shallow and deep linguistic features extracted from the datasets. The STS training corpora are first preprocessed using different tools that annotate the texts at different levels. Using the preprocessed data, the features are extracted for each pair and used to train a model that will be applied to unseen test pairs. The training set is composed by three datasets (*MSRpar*, *MSRvid* and *SMTeuroparl*) which combined contain a total of 2234 instances. The test data is composed by a different sample of the same three datasets plus instances derived from two additional corpora (*OnWN*

*First Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 624–630,
Montréal, Canada, June 7-8, 2012. ©2012 Association for Computational Linguistics

and *SMTnews*). The datasets construction and annotation are described in (Agirre et al., 2012).

Our system exploits two sets of features which respectively build on MT evaluation metrics (2.1) and word similarity metrics (2.2). The whole feature set is summarized in figure 1.

2.1 Machine Translation Evaluation Metrics

MT evaluation metrics are designed to assess whether the output of a MT system is semantically equivalent to a set of reference translations. The MT evaluation metrics described in this section, implemented in the Asiya Open Toolkit for Automatic Machine Translation (Meta-) Evaluation[1] (Giménez and Màrquez, 2010) are used to extract features at different linguistic levels: lexical, syntactic and semantic. For the syntactic and semantic levels, Asiya calculates similarity measures based on the linguistic elements provided by each kind of annotation. Linguistic elements are defined as "the linguistic units, structures, or relationships" (Giménez, 2008) (e.g. dependency relations, discourse relations, named entities, part-of-speech tags, among others). (Giménez, 2008) defines two simple measures using the linguistic elements of a given linguistic level: overlapping and matching. `Overlapping` is a measure of the proportion of items inside the linguistic elements of a certain type shared by both texts. `Matching` is defined in the same way with the difference that the order between the items inside a linguistic element is taken into consideration. That is, the items of a linguistic element are concatenated in a single unit from left to right.

2.1.1 Lexical Level

At the lexical level we explored different n-gram and edit distance based metrics. The difference among them is in the way each algorithm calculates the lexical similarity, which yields to different results. We used the following n-gram-based metrics: BLEU (Papineni et al., 2002), NIST (Doddington, 2002), ROUGE (Lin and Och, 2004), GTM (Melamed et al., 2003), METEOR (Banerjee and Lavie, 2005). Besides those, we also used metrics based on edit distance. Such metrics calculate the number of edit operations (e.g. insertions, deletions, and substitutions) necessary to transform one text into the other (the lower the number of edit operations, the higher the similarity score). The edit-distance-based metrics used were: WER (Nieß en et al., 2000), PER (Tillmann et al., 1997), TER (Snover et al., 2006) and TER-Plus (Snover et al., 2009). The lexical metrics form a group of metrics that we hereafter call `lex`.

2.1.2 Syntactic Level

The syntactic level was explored by running constituency parsing (`cp`), dependency parsing (`dp`), and shallow parsing (`sp`). Constituency trees were produced by the Max-Ent reranking parser (Charniak, 2005). The **constituency parse** trees were exploited by using three different classes of metrics that were designed to calculate the similarities between the trees of two texts: `overlapping` in function of a given part-of-speech; `matching` in function of a given constituency type; and syntactic tree matching (STM) metric proposed by (Liu and Gildea, 2005).

Dependency trees were obtained using MINIPAR (Lin, 2003). Two types of metrics were used to calculate the similarity between two texts using dependency trees. In the first, different similarity measures were calculated taking into consideration three different perspectives: overlap of words that hang in the same level or in a deeper level of the dependency tree; overlap between words that hang directly from terminal nodes given a specified part-of-speech; and overlap between words that are ruled by non-terminal nodes given a specified grammatical relation (subject, object, relative clause, among others). The second type is an implementation of the head-word chain matching introduced in (Liu and Gildea, 2005).

The **shallow syntax** approach proposed by (Giménez, 2008) uses three different tools to explore the parts-of-speech, word lemmas and base phrases chunks, respectively: SVMTool (Giménez and Màrquez, 2004), Freeling (Carreras et al., 2004) and Phreco (Carreras et al., 2005). In this type of metrics the idea is to measure the similarity between the two texts using parts-of-speech and chunk types. The following metrics were used: `overlapping` according to the part-of-speech; `overlapping` according to the chunk type; the accumulated NIST metric (Doddington, 2002) scores over different

[1] `http://nlp.lsi.upc.edu/asiya/`

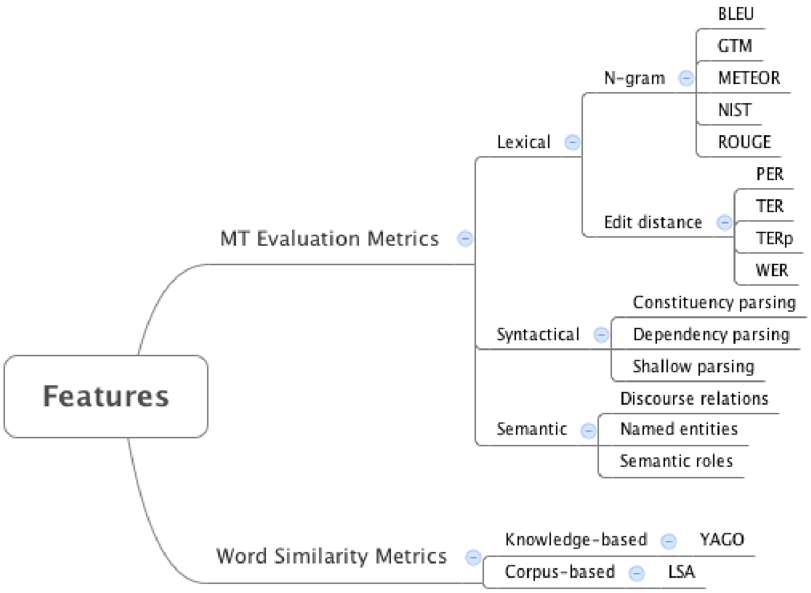

Figure 1: A summary of the class of features explored.

sequences (lemmas, parts-of-speech, base phrase chunks and chunk IOB labels).

2.1.3 Semantic Level

At the semantic level we aplored three different types of information, namely: discourse representations, named entities and semantic roles. Hereafter they are respectively referred to as `dr`, `ne`, and `sr` features. The discourse relations are automatically annotated using the C&C Tools (Clark and Curran, 2004). The following metrics using semantic tree representations were proposed by (Giménez, 2008). A metric similar to the STM in which semantic trees are used instead of constituency trees; the `overlapping` between **discourse representation** structures according to their type; and the morphosyntactic `overlapping` of discourse representation structures that share the same type.

Named entities metrics are calculated by comparing the entities that appear in each text. The named entities were annotated using the BIOS package (Surdeanu et al., 2005). Two types of metrics were used: the `overlapping` between the named entities in each sentence according to their type and the `matching` between the named entities in function of their type.

Semantic roles were automatically annotated using the SwiRL package (Surdeanu and Turmo, 2005). The arguments and adjuncts annotated in each sentence are compared according to three different metrics: `overlapping` between the semantic roles according to their type; the `matching` between the semantic roles according to their type; and the `overlapping` of the roles without taking into consideration their lexical realization.

2.2 Word Similarity Metrics

Besides the MT evaluation metrics, we experimented with lexical semantics by calculating word similarity metrics. For that, we followed a distributional and a knowledge-based word similarity approach.

2.2.1 Distributional Word Similarity

As some previous work on semantic textual textual similarity (Mihalcea et al., 2006) and textual entailment (Kouylekov et al., 2010; Mehdad et al., 2010) have shown, distributional word similarity measures can improve the performance of both tasks by allowing matches between terms that are lexically different. We measure the word similarity computing a set of Latent Semantic Analysis (LSA) metrics over Wikipedia. The 200,000 most visited articles of Wikipedia were extracted and cleaned to build the term-by-document matrix using the jLSI tool[2].

Using this model we designed three different similarity metrics that compute the similarity between all elements in one text with all elements in the other text. For two metrics we calculate the similarities between different parts-of-speech: (i) similarity over nouns and adjectives, and (ii) similarity over verbs. The third metric computes the similarity between all words in the two sentences. The similarity is computed by averaging the pairwise similarity using the LSA model between the elements of each text. These metrics are hereafter called `lsa`.

2.2.2 Knowledge-based Word Similarity

In order to incorporate world knowledge information about entities (persons, organizations, locations, among others) into our model we experimented with knowledge-based (thesaurus-based) word similarity metrics. Usually such approaches have a very limited coverage of concepts due to the reduced size of the available thesauri. In order to increase the coverage we extracted concepts from the YAGO2 semantic knowledge base (Hoffart et al., 2011) derived from Wikipedia, Wordnet (Miller, 1995) and Geonames[3]. YAGO2 contains knowledge about 10 million entities and more than 120 million facts about these entities.

In order to link the entities in the text to the entities in YAGO2 we have used "The Wiki Machine" (TWM) tool[4]. The tool solves the linking problem by disambiguating each entity mention in the text (excluding pronouns) using Wikipedia to provide the sense inventory and the training data (Giuliano et al., 2009). After preprocessing the datasets with TWM the entities are annotated with their respective Wikipedia entries represented by their URLs. Using the entity's URL it is possible to retrieve the Wordnet synsets related to the entity's entry in YAGO2 and explore different knowledge-based metrics to compute word similarity between entities.

In our experiments we selected three different algorithms to calculate word similarity using YAGO2: Wu-Palmer (Zhibiao and Palmer, 1994), the Leacock-Chodorow (Leacock et al., 1998) and the path distance (score based on the shortest path that connects the senses in the Wordnet hypernym/hyponym taxonomy). Two classes of metrics were designed: (i) the average of the similarity between all the entities in each sentence and (ii) the similarity of the pair of elements which have the shortest path in the Wordnet taxonomy among all possible pairs. There are six different metrics using the three algorithms in total. An extra metric was designed using only TWM. The metric is calculated by taking the number of common entities in the two sentences divided by the total number of entities annotated in the two sentences. The metrics described in this section are part of the `yago` group.

3 Experiments and Discussion

In this section we present our experiments settings, the configuration of the runs submitted and discuss the results obtained. All our experiments were made using half of the training set for training and half for testing (development). Ten different randomizations were run over the training data in order to obtain ten different pairs of train/development sets and reduce overfitting. We tried several different regression algorithms and the best performance was achieved with the implementation of Support Vector Machines (SVM) of the SVMLight package (Joachims, 1998). We used the radial basis function kernel with default parameters without any special tuning for the different datasets.

3.1 Submitted Runs and Results

Based on the results achieved with different feature sets over training data we have selected the best combinations for our submission. The feature sets for each run are:

> **Run 1:** `lex`, `lsa`, `yago`, and a selection of features in the `cp`, `dp`, `sp`, `dr`, `ne` and `sr` groups, forming a total of 286 features.
>
> **Run 2:** `lex`, `lsa`, and `yago`, in a total of 50 features.
>
> **Run 3:** `lex` and `lsa`, forming a total of 43 features.

The results obtained by our three submitted runs are summarized in table 1. The table reports the

[2] `http://hlt.fbk.eu/en/technology/jlsi`

[3] `http://www.geonames.org/`

[4] `http://thewikimachine.fbk.eu/html/index.html`

		Runs submitted				
		Run 1	Run 2	Run 3	Base	PE
Development		0.885	0.863	0.859	-	-
Test	MSp	0.249	0.512	**0.516**	0.433	0.577
	MSv	0.611	**0.780**	0.777	0.299	0.818
	SMTe	0.149	0.379	**0.441**	0.454	0.450
	Wn	0.421	0.622	**0.629**	0.586	0.629
	SMTn	0.243	0.547	**0.608**	0.390	0.608
	All	0.563	0.643	**0.651**	0.310	0.789
	Allnrm	0.712	0.808	**0.810**	0.673	0.633
	Mean	0.362	0.588	**0.607**	0.435	0.829

Table 1: Results of each run for each dataset (MSRpar, MSRvid, SMTeuroparl, OnWn, SMTnews) calculated with the Pearson correlation between the system's outputs and the gold standard annotation. Official scores obtained using the three evaluation scores *All*, *Allnrm* and *Mean*. Development row presents the average results for each run in the whole training dataset. Base is the official baseline system. Post Evaluation is the experiment ran after the evaluation period with models trained for the specific datasets.

Pearson correlation between the system output and the gold standard annotation provided by the task organizers. The table also presents the official scores used to rank the systems and described in (Agirre et al., 2012). Our best model, Run 3, was ranked 20th according to the *Mean* score, 25th according to the *RankNrm* score and 32th according to the *All* score among 88 submitted runs.

The "Development" row reports the results of our three best models in the development phase. The results obtained for the three training datasets are higher than the results obtained for the testing. One hypothesis that might explain this behavior is overfitting during the training phase due to the way we divided the training set and carried out the experiments. A different experiment setting to carry out the development should be tried to evaluate this hypothesis.

To our surprise, in the test datasets the results of Run 1 and Run 3 swapped positions: in the training setting Run 1 was the best model and Run 3 the third best. The performance of Run 3 was relatively stable across the five datasets ranging from about the 30th to the 48th position the exception being the *SMTnews* dataset. In this dataset Run 3 was the best performing run of the evaluation exercise (and Run 2 the second). One possible explanation for this behavior is the fact that Run 3 is based on lexical features that do not take into consideration the syntactic structure of the two texts and therefore is not penalized by the noise introduced by the texts generated by MT systems. This hypothesis, however, does not explain why Run 3 score for the *SMTeuroparl* dataset was below the baseline score. Error analysis of the effects of different group of features in the test datasets is required to better understand such behaviors.

3.2 Post-evaluation Experiments

After the evaluation period, as a first step towards the required error analysis and a better comprehension of the potential of our approach, we performed an experiment to assess the impact of having models trained for specific datasets. In this experiment, each training dataset (*MSRpar*, *MSRvid* and *SMTeuroparl*) was used to train a model. Each dataset's model was tested on its respective test dataset. The model for the surprise datasets (*OnWn* and *SMTnews*) were trained using the whole training dataset. We used the Run 3 feature set (the best run in the official evaluation). The results of the experiment are reported in the column "Exp" of table 1. The impact of having specific models for each dataset is high. The *Mean* score goes from .607 to .829 and improvements are also observed in the *All* score (0.789). These scores would rank our system at the 7th position in the *Mean* rank. However, it is important to notice that in a real-world setting, knowledge about the source of data is not always available. We consider that having a general model that does not rely on this kind of information represents a more realistic way to confront with real-world applications.

4 Final Remarks

In this paper we described FBK's participation in the STS Semeval 2012 task. Our approach is based on a combination of MT evaluation metrics, distributional, and knowledge-based word similarity metrics. Our best run achieved the 20th position among 88 runs in the *Mean* overall ranking. An error analysis of the problematic test pairs is required to understand the potential of our feature sets and improve the overall performance of our approach. Along this direction, a first experiment with our best features and a different strategy already led to significant improvements in the *Mean* and *All* scores (from .651 to

.789 and from .607 to .829, respectively).

Acknowledgments

This work has been partially supported by the EC-funded project CoSyne (FP7-ICT-4-24853).

The authors would like to thank Claudio Giuliano for kindly helping us to preprocess the datasets with the Wiki Machine.

References

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez. 2012. SemEval-2012 Task 6: A Pilot on Semantic Textual Similarity. In *6th International Workshop on Semantic Evaluation (SemEval 2012), in conjunction with the First Joint Conference on Lexical and Computational Semantics (*SEM 2012)*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *ACL 2005 Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization*.

Xavier Carreras, Isaac Chao, Lluís Padro, and Muntsa Padró. 2004. Freeling: An open-source suite of language analyzers. In *4th International Conference on Language Resources and Evaluation (LREC)*, pages 239–242.

Xavier Carreras, Lluís Màrquez, and Jorge Catro. 2005. Filtering-Ranking Perceptron Learning. *Machine Learning*, 60:41–75.

Eugene Charniak. 2005. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In *Proceedings of the 43rd Annual Meeting on*, volume 1, pages 173–180.

Stephen Clark and James R. Curran. 2004. Parsing the WSJ using CCG and log-linear models. In *ACL '04 Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*.

George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the Second International Conference on Human Language Technology Research*, pages 138–145. Morgan Kaufmann Publishers Inc.

Andrew Finch and YS Hwang. 2005. Using machine translation evaluation techniques to determine sentence-level semantic equivalence. In *Third International Workshop on Paraphrasing*, pages 17–24.

Jesús Giménez and Lluís Màrquez. 2004. SVMTool: A general POS tagger generator based on Support Vector Machines. In *4th International Conference on Language Resources and Evaluation (LREC)*, pages 43–46.

Jesús Giménez and Lluís Màrquez. 2010. Asiya: An Open Toolkit for Automatic Machine Translation (Meta-) Evaluation. *The Prague Bulletin of Mathematical Linguistics*, (94):77–86.

J. Giménez. 2008. *Empirical Machine Translation and its Evaluation*. Ph.D. thesis.

Claudio Giuliano, Alfio Massimiliano Gliozzo, and Carlo Strapparava. 2009. Kernel methods for minimally supervised wsd. *Computational Linguistics*, 35(4):513–528.

Johannes Hoffart, Fabian M. FM Suchanek, Klaus Berberich, Edwin Lewis Kelham, Gerard de Melo, and Gerhard Weikum. 2011. YAGO2: Exploring and Querying World Knowledge in Time, Space, Context, and Many Languages. In *20th International World Wide Web Conference (WWW 2011)*, pages 229–232.

Thorsten Joachims. 1998. Making Large-Scale SVM Learning Practical. In Bernhard Scholkopf, Christopher J. C. Burges, and Alexander J. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, pages 41–56. MIT Press, Cambridge, USA.

Milen Kouylekov, Yashar Mehdad, and Matteo Negri. 2010. Mining Wikipedia for Large-Scale Repositories of Context-Sensitive Entailment Rules. In *Seventh international conference on Language Resources and Evaluation (LREC 2010)*, pages 3550–3553, La Valletta, Malta.

Claudia Leacock, George A. Miller, and Martin Chodorow. 1998. Using corpus statistics and WordNet relations for sense identification. *Computational Linguistics*, 24(1):147–166.

C.Y. Lin and F.J. Och. 2004. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 605. Association for Computational Linguistics.

Dekang Lin. 2003. Dependency-Based Evaluation of Minipar. *Text, Speech and Language Technology*, 20:317–329.

Ding Liu and Daniel Gildea. 2005. Syntactic features for evaluation of machine translation. In *ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, number June, pages 25–32.

Yashar Mehdad, Alessandro Moschitti, and Fabio Massimo Zanzotto. 2010. Syntactic/semantic structures for textual entailment recognition. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL*, number June, pages 1020–1028.

I. Dan Melamed, Ryan Green, and Joseph P. Turian. 2003. Precision and Recall of Machine Translation. In

Proceedings of the Joint Conference on Human Language Technology and the North American Chapter of the Association for Computational Linguistics (HLT-NAACL).

Rada Mihalcea, Courtney Corley, and Carlo Strapparava. 2006. Corpus-based and knowledge-based measures of text semantic similarity. In *Proceedings of the American Association for Artificial Intelligence*, pages 775–780.

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

Sonja Nieß en, Franz Josef Och, Gregor Leusch, and Hermann Ney. 2000. An evaluation tool for machine translation: Fast evaluation for MT research. In *Language Resources and Evaluation*, pages 0–6.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-jing 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)*, number July, pages 311–318.

Gerard Salton, Amit Singhal, and Mandar Mitra. 1997. Automatic text structuring and summarization. *Information Processing &amp;*, 33(2):193–207.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *Association for Machine Translation in the Americas*.

Matthew G. Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz. 2009. TER-Plus: paraphrase, semantic, and alignment enhancements to Translation Edit Rate. *Machine Translation*, 23(2-3):117–127, December.

Mihai Surdeanu and Jordi Turmo. 2005. Semantic role labeling using complete syntactic analysis. In *9th Conference on Computational Natural Language Learning (CoNLL)*, number June, pages 221–224.

Mihai Surdeanu, Jordi Turmo, and Eli Comelles. 2005. Named Entity Recognition from Spontaneous Open-domain Speech. In *9th International Conference on Speech Communication and Technology (Interspeech)*, pages 3433–3436.

C Tillmann, S Vogel, H Ney, A. Zubiaga, and H. Sawaf. 1997. Accelerated DP Based Search for Statistical Translation. In *Fifth European Conference on Speech Communication and Technology*, pages 2667–2670.

Stephen Wan, Mark Dras, Robert Dale, and Cécile Paris. 2006. Using Dependency-Based Features to Take the "Para-farce" out of Paraphrase. In *2006 Australasian Language Technology Workshop (ALTW2006)*, number 2005, pages 131–138.

Wu Zhibiao and Martha Palmer. 1994. Verb Semantics and Lexical Selection. In *ACL '94 Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 133–138.