Extracting Paraphrases from a Parallel Corpus

Regina Barzilay and **Kathleen R. McKeown**
Computer Science Department
Columbia University
10027, New York, NY, USA
{regina,kathy}@cs.columbia.edu

Abstract

While paraphrasing is critical both for interpretation and generation of natural language, current systems use manual or semi-automatic methods to collect paraphrases. We present an unsupervised learning algorithm for identification of paraphrases from a corpus of multiple English translations of the same source text. Our approach yields phrasal and single word lexical paraphrases as well as syntactic paraphrases.

1 Introduction

Paraphrases are alternative ways to convey the same information. A method for the automatic acquisition of paraphrases has both practical and linguistic interest. From a practical point of view, diversity in expression presents a major challenge for many NLP applications. In multidocument summarization, identification of paraphrasing is required to find repetitive information in the input documents. In generation, paraphrasing is employed to create more varied and fluent text. Most current applications use manually collected paraphrases tailored to a specific application, or utilize existing lexical resources such as WordNet (Miller et al., 1990) to identify paraphrases. However, the process of manually collecting paraphrases is time consuming, and moreover, the collection is not reusable in other applications. Existing resources only include lexical paraphrases; they do not include phrasal or syntactically based paraphrases.

From a linguistic point of view, questions concern the operative definition of paraphrases: what types of lexical relations and syntactic mechanisms can produce paraphrases? Many linguists (Halliday, 1985; de Beaugrande and Dressler, 1981) agree that paraphrases retain "approximate conceptual equivalence", and are not limited only to synonymy relations. But the extent of interchangeability between phrases which form paraphrases is an open question (Dras, 1999). A corpus-based approach can provide insights on this question by revealing paraphrases that people use.

This paper presents a corpus-based method for automatic extraction of paraphrases. We use a large collection of multiple parallel English translations of novels[1]. This corpus provides many instances of paraphrasing, because translations preserve the meaning of the original source, but may use different words to convey the meaning. An example of parallel translations is shown in Figure 1. It contains two pairs of paraphrases: ("*burst into tears*", "*cried*") and ("*comfort*", "*console*").

Emma burst into tears and he tried to comfort her, saying things to make her smile.
Emma cried, and he tried to console her, adorning his words with puns.

Figure 1: Two English translations of the French sentence from Flaubert's "Madame Bovary"

Our method for paraphrase extraction builds upon methodology developed in Machine Translation (MT). In MT, pairs of translated sentences from a bilingual corpus are aligned, and occurrence patterns of words in two languages in the text are extracted and matched using correlation measures. However, our parallel corpus is far from the clean parallel corpora used in MT. The

[1] Foreign sources are not used in our experiment.

rendition of a literary text into another language not only includes the translation, but also restructuring of the translation to fit the appropriate literary style. This process introduces differences in the translations which are an intrinsic part of the creative process. This results in greater differences across translations than the differences in typical MT parallel corpora, such as the Canadian Hansards. We will return to this point later in Section 3.

Based on the specifics of our corpus, we developed an unsupervised learning algorithm for paraphrase extraction. During the preprocessing stage, the corresponding sentences are aligned. We base our method for paraphrasing extraction on the assumption that phrases in aligned sentences which appear in similar contexts are paraphrases. To automatically infer which contexts are good predictors of paraphrases, contexts surrounding identical words in aligned sentences are extracted and filtered according to their predictive power. Then, these contexts are used to extract new paraphrases. In addition to learning lexical paraphrases, the method also learns syntactic paraphrases, by generalizing syntactic patterns of the extracted paraphrases. Extracted paraphrases are then applied to the corpus, and used to learn new context rules. This iterative algorithm continues until no new paraphrases are discovered.

A novel feature of our approach is the ability to extract multiple kinds of paraphrases:

Identification of lexical paraphrases. In contrast to earlier work on similarity, our approach allows identification of multi-word paraphrases, in addition to single words, a challenging issue for corpus-based techniques.

Extraction of morpho-syntactic paraphrasing rules. Our approach yields a set of paraphrasing patterns by extrapolating the syntactic and morphological structure of extracted paraphrases. This process relies on morphological information and a part-of-speech tagging. Many of the rules identified by the algorithm match those that have been described as productive paraphrases in the linguistic literature.

In the following sections, we provide an overview of existing work on paraphrasing, then we describe data used in this work, and detail our paraphrase extraction technique. We present results of our evaluation, and conclude with a discussion of our results.

2 Related Work on Paraphrasing

Many NLP applications are required to deal with the unlimited variety of human language in expressing the same information. So far, three major approaches of collecting paraphrases have emerged: manual collection, utilization of existing lexical resources and corpus-based extraction of similar words.

Manual collection of paraphrases is usually used in generation (Iordanskaja et al., 1991; Robin, 1994). Paraphrasing is an inevitable part of any generation task, because a semantic concept can be realized in many different ways. Knowledge of possible concept verbalizations can help to generate a text which best fits existing syntactic and pragmatic constraints. Traditionally, alternative verbalizations are derived from a manual corpus analysis, and are, therefore, application specific.

The second approach — utilization of existing lexical resources, such as WordNet — overcomes the scalability problem associated with an application specific collection of paraphrases. Lexical resources are used in statistical generation, summarization and question-answering. The question here is what type of WordNet relations can be considered as paraphrases. In some applications, only synonyms are considered as paraphrases (Langkilde and Knight, 1998); in others, looser definitions are used (Barzilay and Elhadad, 1997). These definitions are valid in the context of particular applications; however, in general, the correspondence between paraphrasing and types of lexical relations is not clear. The same question arises with automatically constructed thesauri (Pereira et al., 1993; Lin, 1998). While the extracted pairs are indeed similar, they are not paraphrases. For example, while “dog” and “cat” are recognized as the most similar concepts by the method described in (Lin, 1998), it is hard to imagine a context in which these words would be interchangeable.

The first attempt to derive paraphrasing rules from corpora was undertaken by (Jacquemin et al., 1997), who investigated morphological and syntactic variants of technical terms. While these rules achieve high accuracy in identifying term paraphrases, the techniques used have not been extended to other types of paraphrasing yet. Statistical techniques were also successfully used by (Lapata, 2001) to identify paraphrases of adjective-noun phrases. In contrast, our method is not limited to a particular paraphrase type.

3 The Data

The corpus we use for identification of paraphrases is a collection of multiple English translations from a foreign source text. Specifically, we use literary texts written by foreign authors. Many classical texts have been translated more than once, and these translations are available on-line. In our experiments we used 5 books, among them, Flaubert's *Madame Bovary*, Andersen's *Fairy Tales* and Verne's *Twenty Thousand Leagues Under the Sea*. Some of the translations were created during different time periods and in different countries. In total, our corpus contains 11 translations [2].

At first glance, our corpus seems quite similar to parallel corpora used by researchers in MT, such as the Canadian Hansards. The major distinction lies in the degree of proximity between the translations. Analyzing multiple translations of the literary texts, critics (e.g. (Wechsler, 1998)) have observed that translations "are never identical", and each translator creates his own interpretations of the text. Clauses such as *"adorning his words with puns"* and *"saying things to make her smile"* from the sentences in Figure 1 are examples of distinct translations. Therefore, a complete match between words of related sentences is impossible. This characteristic of our corpus is similar to problems with noisy and comparable corpora (Veronis, 2000), and it prevents us from using methods developed in the MT community based on clean parallel corpora, such as (Brown et al., 1993).

Another distinction between our corpus and parallel MT corpora is the irregularity of word matchings: in MT, no words in the source language are kept as is in the target language translation; for example, an English translation of a French source does not contain untranslated French fragments. In contrast, in our corpus the same word is *usually* used in both translations, and only sometimes its paraphrases are used, which means that word–paraphrase pairs will have lower co-occurrence rates than word–translation pairs in MT. For example, consider occurrences of the word "boy" in two translations of "Madame Bovary" — E. Marx-Aveling's translation and Etext's translation. The first text contains 55 occurrences of "boy", which correspond to 38 occurrences of "boy" and 17 occurrences of its paraphrases ("son", "young fellow" and "youngster"). This rules out using word translation methods based only on word co-occurrence counts.

On the other hand, the big advantage of our corpus comes from the fact that parallel translations share many words, which helps the matching process. We describe below a method of paraphrase extraction, exploiting these features of our corpus.

4 Preprocessing

During the preprocessing stage, we perform sentence alignment. Sentences which are translations of the same source sentence contain a number of identical words, which serve as a strong clue to the matching process. Alignment is performed using dynamic programming (Gale and Church, 1991) with a weight function based on the number of common words in a sentence pair. This simple method achieves good results for our corpus, because 42% of the words in corresponding sentences are identical words on average. Alignment produces 44,562 pairs of sentences with 1,798,526 words. To evaluate the accuracy of the alignment process, we analyzed 127 sentence pairs from the algorithm's output. 120(94.5%) alignments were identified as correct alignments.

We then use a part-of-speech tagger and chunker (Mikheev, 1997) to identify noun and verb phrases in the sentences. These phrases become the atomic units of the algorithm. We also record for each token its derivational root, using the CELEX(Baayen et al., 1993) database.

5 Method for Paraphrase Extraction

Given the aforementioned differences between translations, our method builds on similarity in

[2] Free of copyright restrictions part of our corpus(9 translations) is available at `http://www.cs.columbia.edu/˜regina /par`.

the local context, rather than on global alignment. Consider the two sentences in Figure 2.

And finally, dazzlingly white, it shone high above them in the empty $\boxed{?}$.
It appeared white and dazzling in the empty $\boxed{?}$.

Figure 2: Fragments of aligned sentences

Analyzing the contexts surrounding "$\boxed{?}$"-marked blanks in both sentences, one expects that they should have the same meaning, because they have the same premodifier "empty" and relate to the same preposition "in" (in fact, the first "$\boxed{?}$" stands for "sky", and the second for "heavens"). Generalizing from this example, we hypothesize that if the contexts surrounding two phrases look similar enough, then these two phrases are likely to be paraphrases. The definition of the context depends on how similar the translations are. Once we know which contexts are good paraphrase predictors, we can extract paraphrase patterns from our corpus.

Examples of such contexts are verb-object relations and noun-modifier relations, which were traditionally used in word similarity tasks from non-parallel corpora (Pereira et al., 1993; Hatzivassiloglou and McKeown, 1993). However, in our case, more indirect relations can also be clues for paraphrasing, because we know *a priori* that input sentences convey the same information. For example, in sentences from Figure 3, the verbs "ringing" and "sounding" do not share identical subject nouns, but the modifier of both subjects "Evening" is identical. Can we conclude that identical modifiers of the subject imply verb similarity? To address this question, we need a way to identify contexts that are good predictors for paraphrasing in a corpus.

People said "The Evening Noise is sounding, the sun is setting."
"The evening bell is ringing," people used to say.

Figure 3: Fragments of aligned sentences

To find "good" contexts, we can analyze all contexts surrounding identical words in the pairs of aligned sentences, and use these contexts to learn new paraphrases. This provides a basis for a bootstrapping mechanism. Starting with identical words in aligned sentences as a seed, we can incrementally learn the "good" contexts, and in turn use them to learn new paraphrases. Identical words play two roles in this process: first, they are used to learn context rules; second, identical words are used in application of these rules, because the rules contain information about the equality of words in context.

This method of co-training has been previously applied to a variety of natural language tasks, such as word sense disambiguation (Yarowsky, 1995), lexicon construction for information extraction (Riloff and Jones, 1999), and named entity classification (Collins and Singer, 1999). In our case, the co-training process creates a binary classifier, which predicts whether a given pair of phrases makes a paraphrase or not.

Our model is based on the DLCoTrain algorithm proposed by (Collins and Singer, 1999), which applies a co-training procedure to decision list classifiers for two independent sets of features. In our case, one set of features describes the paraphrase pair itself, and another set of features corresponds to contexts in which paraphrases occur. These features and their computation are described below.

5.1 Feature Extraction

Our paraphrase features include lexical and syntactic descriptions of the paraphrase pair. The lexical feature set consists of the sequence of tokens for each phrase in the paraphrase pair; the syntactic feature set consists of a sequence of part-of-speech tags where equal words and words with the same root are marked. For example, the value of the syntactic feature for the pair ("the vast chimney", "the chimney") is ("DT$_1$ JJ NN$_2$", "DT$_1$ NN$_2$"), where indices indicate word equalities. We believe that this feature can be useful for two reasons: first, we expect that some syntactic categories can not be paraphrased in another syntactic category. For example, a determiner is unlikely to be a paraphrase of a verb. Second, this description is able to capture regularities in phrase level paraphrasing. In fact, a similar representation was used by (Jacquemin et al., 1997) to describe term variations.

The contextual feature is a combination of the left and right syntactic contexts surrounding actual known paraphrases. There are a number of context representations that can be considered as possible candidates: lexical n-grams, POS-ngrams and parse tree fragments. The natural choice is a parse tree; however, existing parsers perform poorly in our domain[3]. Part-of-speech tags provide the required level of abstraction, and can be accurately computed for our data. The left (right) context is a sequence of part-of-speech tags of n words, occurring on the left (right) of the paraphrase. As in the case of syntactic paraphrase features, tags of identical words are marked. For example, when $n = 2$, the contextual feature for the paraphrase pair ("*comfort*", "*console*") from Figure 1 sentences is *left*$_1$="VB$_1$ TO$_2$", ("*tried to*"), *left*$_2$="VB$_1$ TO$_2$", ("*tried to*"), *right*$_1$="PRP\$$_3$,$_4$", ("*her,*") *right_context*\$$_2$="PRP\$$_3$,$_4$", ("*her,*"). In the next section, we describe how the classifiers for contextual and paraphrasing features are co-trained.

5.2 The co-training algorithm

Our co-training algorithm has three stages: initialization, training of the contextual classifier and training of the paraphrasing classifiers.

Initialization Words which appear in both sentences of an aligned pair are used to create the initial "seed" rules. Using identical words, we create a set of positive paraphrasing examples, such as *word*$_1$=*tried, word*$_2$=*tried*. However, training of the classifier demands negative examples as well; in our case it requires pairs of words in aligned sentences which are not paraphrases of each other. To find negative examples, we match identical words in the alignment against all different words in the aligned sentence, assuming that identical words can match only each other, and not any other word in the aligned sentences. For example, "*tried*" from the first sentence in Figure 1 does not correspond to any other word in the second sentence but "*tried*". Based on this observation, we can derive negative examples such as *word*$_1$=*tried, word*$_2$=*Emma* and *word*$_1$=*tried, word*$_2$=*console*. Given a pair of identical words from two sentences of length n and m, the algorithm produces one positive example and $(n-1)+(m-1)$ negative examples.

Training of the contextual classifier Using this initial seed, we record contexts around positive and negative paraphrasing examples. From all the extracted contexts we must identify the ones which are strong predictors of their category. Following (Collins and Singer, 1999), filtering is based on the strength of the context and its frequency. The strength of positive context x is defined as $\mathrm{count}(x+)/\mathrm{count}(x)$, where $\mathrm{count}(x+)$ is the number of times context x surrounds positive examples (paraphrase pairs) and $\mathrm{count}(x)$ is the frequency of the context x. Strength of the negative context is defined in a symmetrical manner. For the positive and the negative categories we select k rules ($k = 10$ in our experiments) with the highest frequency and strength higher than the predefined threshold of 95%. Examples of selected context rules are shown in Figure 4.

The parameter of the contextual classifier is a context length. In our experiments we found that a maximal context length of three produces best results. We also observed that for some rules a shorter context works better. Therefore, when recording contexts around positive and negative examples, we record all the contexts with length smaller or equal to the maximal length.

Because our corpus consists of translations of several books, created by different translators, we expect that the similarity between translations varies from one book to another. This implies that contextual rules should be specific to a particular pair of translations. Therefore, we train the contextual classifier for each pair of translations separately.

left$_1$ = (VB$_0$ TO$_1$) *right*$_1$ = (PRP\$$_2$,) *left*$_2$ = (VB$_0$ TO$_1$) *right*$_2$ = (PRP\$$_2$,)
left$_1$ = (WRB$_0$ NN$_1$) *right*$_1$ = (NN$_2$ IN) *left*$_2$ = (WRB$_0$ NN$_1$) *right*$_2$ = (NN$_2$ IN)
left$_1$ = (VB$_0$) *right*$_1$ = (JJ$_1$) *left*$_2$ = (VB$_0$) *right*$_2$ = (JJ$_1$)
left$_1$ = (IN NN$_0$) *right*$_1$ = (NN$_2$ IN$_3$) *left*$_2$ = (NN$_0$,) *right*$_2$ = (NN$_2$ IN$_3$)

Figure 4: Example of context rules extracted by the algorithm.

Training of the paraphrasing classifier Context rules extracted in the previous stage are then applied to the corpus to derive a new set of pairs

[3] To the best of our knowledge all existing statistical parsers are trained on WSJ or similar type of corpora. In the experiments we conducted, their performance significantly degraded on our corpus — literary texts.

of positive and negative paraphrasing examples. Applications of the rule performed by searching sentence pairs for subsequences which match the left and right parts of the contextual rule, and are less than N tokens apart. For example, applying the first rule from Figure 4 to sentences from Figure 1 yields the paraphrasing pair ("*comfort*", "*console*"). Note that in the original seed set, the left and right contexts were separated by one token. This stretch in rule application allows us to extract multi-word paraphrases.

For each extracted example, paraphrasing rules are recorded and filtered in a similar manner as contextual rules. Examples of lexical and syntactic paraphrasing rules are shown in Figure 5 and in Figure 6. After extracted lexical and syntactic paraphrases are applied to the corpus, the contextual classifier is retrained. New paraphrases not only add more positive and negative instances to the contextual classifier, but also revise contextual rules for known instances based on new paraphrase information.

(NN$_0$ POS NN$_1$)↔	(NN$_1$ IN DT NN$_0$)
King's son	son of the king
(IN NN^0)↔	(VB^0)
in bottles	bottled
(VB$_0$ to VB^1)↔	(VB$_0$ VB^1)
start to talk	start talking
(VB$_0$ RB$_1$)↔	(RB$_1$ VB$_0$)
suddenly came	came suddenly
(VB NN^0)↔	(VB^0)
make appearance	appear

Figure 5: Morpho-Syntactic patterns extracted by the algorithm. Lower indices denote token equivalence, upper indices denote root equivalence.

(countless, lots of)	(repulsion, aversion)
(undertone, low voice)	(shrubs, bushes)
(refuse, say no)	(dull tone, gloom)
(sudden appearance, apparition)	

Figure 6: Lexical paraphrases extracted by the algorithm.

The iterative process is terminated when no new paraphrases are discovered or the number of iterations exceeds a predefined threshold.

6 The results

Our algorithm produced 9483 pairs of lexical paraphrases and 25 morpho-syntactic rules. To evaluate the quality of produced paraphrases, we picked at random 500 paraphrasing pairs from the lexical paraphrases produced by our algorithm. These pairs were used as test data and also to evaluate whether humans agree on paraphrasing judgments. The judges were given a page of guidelines, defining paraphrase as "approximate conceptual equivalence". The main dilemma in designing the evaluation is whether to include the context: should the human judge see only a paraphrase pair or should a pair of sentences containing these paraphrases also be given? In a similar MT task — evaluation of word-to-word translation — context is usually included (Melamed, 2001). Although paraphrasing is considered to be context dependent, there is no agreement on the extent. To evaluate the influence of context on paraphrasing judgments, we performed two experiments — with and without context. First, the human judge is given a paraphrase pair without context, and after the judge entered his answer, he is given the same pair with its surrounding context. Each context was evaluated by two judges (other than the authors). The agreement was measured using the Kappa coefficient (Siegel and Castellan, 1988). Complete agreement between judges would correspond to K equals 1; if there is no agreement among judges, then K equals 0.

The judges agreement on the paraphrasing judgment without context was $K = 0.68$ which is substantial agreement (Landis and Koch, 1977). The first judge found 439(87.8%) pairs as correct paraphrases, and the second judge — 426(85.2%). Judgments with context have even higher agreement ($K = 0.97$), and judges identified 459(91.8%) and 457(91.4%) pairs as correct paraphrases.

The recall of our method is a more problematic issue. The algorithm can identify paraphrasing relations only between words which occurred in our corpus, which of course does not cover all English tokens. Furthermore, direct comparison with an electronic thesaurus like WordNet is impossible, because it is not known *a priori* which lexical relations in WordNet can form paraphrases. Thus, we can not evaluate recall. We hand-evaluated the coverage, by asking a human judges to extract paraphrases from 50 sentences, and then counted how many of these paraphrases where predicted by our algorithm. From 70 paraphrases extracted by human judge, 48(69%) were identified as paraphrases by our algorithm.

In addition to evaluating our system output through precision and recall, we also compared our results with two other methods. The first of these was a machine translation technique for deriving bilingual lexicons (Melamed, 2001) including detection of non-compositional compounds [4]. We did this evaluation on 60% of the full dataset; this is the portion of the data which is publicly available. Our system produced 6,826 word pairs from this data and Melamed provided the top 6,826 word pairs resulting from his system on this data. We randomly extracted 500 pairs each from both sets of output. Of the 500 pairs produced by our system, 354(70.8%) were single word pairs and 146(29.2%) were multi-word paraphrases, while the majority of pairs produced by Melamed's system were single word pairs (90%). We mixed this output and gave the resulting, randomly ordered 1000 pairs to six evaluators, all of whom were native speakers. Each evaluator provided judgments on 500 pairs without context. Precision for our system was 71.6% and for Melamed's was 52.7%. This increased precision is a clear advantage of our approach and shows that machine translation techniques cannot be used without modification for this task, particularly for producing multi-word paraphrases. There are three caveats that should be noted; Melamed's system was run without changes for this new task of paraphrase extraction and his system does not use chunk segmentation, he ran the system for three days of computation and the result may be improved with more running time since it makes incremental improvements on subsequent rounds, and finally, the agreement between human judges was lower than in our previous experiments. We are currently exploring whether the information produced by the two different systems may be combined to improve the performance of either system alone.

Another view on the extracted paraphrases can be derived by comparing them with the WordNet thesaurus. This comparison provides us with quantitative evidence on the types of lexical relations people use to create paraphrases. We selected 112 paraphrasing pairs which occurred at least 20 times in our corpus and such that the words comprising each pair appear in WordNet. The 20 times cutoff was chosen to ensure that the identified pairs are general enough and not idiosyncratic. We use the frequency threshold to select paraphrases which are not tailored to one context. Examples of paraphrases and their WordNet relations are shown in Figure 7. Only 40(35%) paraphrases are synonyms, 36(32%) are hyperonyms, 20(18%) are siblings in the hyperonym tree, 11(10%) are unrelated, and the remaining 5% are covered by other relations. These figures quantitatively validate our intuition that synonymy is not the only source of paraphrasing. One of the practical implications is that using synonymy relations exclusively to recognize paraphrasing limits system performance.

Synonyms: (rise, stand up), (hot, warm)
Hyperonyms: (landlady, hostess), (reply, say)
Siblings: (city, town), (pine, fir)
Unrelated: (sick, tired), (next, then)

Figure 7: Lexical paraphrases extracted by the algorithm.

7 Conclusions and Future work

In this paper, we presented a method for corpus-based identification of paraphrases from multiple English translations of the same source text. We showed that a co-training algorithm based on contextual and lexico-syntactic features of paraphrases achieves high performance on our data. The wide range of paraphrases extracted by our algorithm sheds light on the paraphrasing phenomena, which has not been studied from an empirical perspective.

Future work will extend this approach to extract paraphrases from comparable corpora, such as multiple reports from different news agencies about the same event or different descriptions of a disease from the medical literature. This extension will require using a more selective alignment technique (similar to that of (Hatzivassiloglou et al., 1999)). We will also investigate a more powerful representation of contextual features. Fortunately, statistical parsers produce reliable results

[4] The equivalences that were identical on both sides were removed from the output

on news texts, and therefore can be used to improve context representation. This will allow us to extract macro-syntactic paraphrases in addition to local paraphrases which are currently produced by the algorithm.

Acknowledgments

This work was partially supported by a Louis Morin scholarship and by DARPA grant N66001-00-1-8919 under the TIDES program. We are grateful to Dan Melamed for providing us with the output of his program. We thank Noemie Elhadad, Mike Collins, Michael Elhadad and Maria Lapata for useful discussions.

References

R. H. Baayen, R. Piepenbrock, and H. van Rijn, editors. 1993. *The CELEX Lexical Database(CD-ROM)*. Linguistic Data Consortium, University of Pennsylvania.

R. Barzilay and M. Elhadad. 1997. Using lexical chains for text summarization. In *Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization*, pages 10–17, Madrid, Spain, August.

P. Brown, S. Della Pietra, V. Della Pietra, and R. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

M. Collins and Y. Singer. 1999. Unsupervised models for named entity classification. In *proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*.

R. de Beaugrande and W. V. Dressler. 1981. *Introduction to Text Linguistics*. Longman, New York, NY.

M. Dras. 1999. *Tree Adjoining Grammar and the Reluctant Paraphrasing of Text*. Ph.D. thesis, Macquarie University, Australia.

W. Gale and K. W. Church. 1991. A program for aligning sentences in bilingual corpora. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, pages 1–8.

M. Halliday. 1985. *An introduction to functional grammar*. Edward Arnold, UK.

V. Hatzivassiloglou and K.R. McKeown. 1993. Towards the automatic identification of adjectival scales: Clustering adjectives according to their meaning. In *Proceedings of the 31rd Annual Meeting of the Association for Computational Linguistics*, pages 172–182.

V. Hatzivassiloglou, J. Klavans, and E. Eskin. 1999. Detecting text similarity over short passages: Exploring linguistic feature combinations via machine learning. In *proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*.

L. Iordanskaja, R. Kittredge, and A. Polguere, 1991. *Natural language Generation in Artificial Intelligence and Computational Linguistics*, chapter 11. Kluwer Academic Publishers.

C. Jacquemin, J. Klavans, and E. Tzoukermann. 1997. Expansion of multi-word terms for indexing and retrieval using morphology and syntax. In *proceedings of the 35th Annual Meeting of the ACL*, pages 24–31, Madrid, Spain, July. ACL.

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

I. Langkilde and K. Knight. 1998. Generation that exploits corpus-based statistical knowledge. In *proceedings of the COLING-ACL*.

Maria Lapata. 2001. A corpus-based account of regular polysemy: The case of context-sensitive adjectives. In *Proceedings of the 2nd Meeting of the NAACL*, Pittsburgh, PA.

D. Lin. 1998. Automatic retrieval and clustering of similar words. In *proceedings of the COLING-ACL*, pages 768–774.

Melamed. 2001. *Empirical Methods for Exploiting Parallel Texts*. MIT press.

A. Mikheev. 1997. the ltg part of speech tagger. University of Edinburgh.

G.A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K.J. Miller. 1990. Introduction to WordNet: An on-line lexical database. *International Journal of Lexicography (special issue)*, 3(4):235–245.

F. Pereira, N. Tishby, and L. Lee. 1993. Distributional clustering of english words. In *proceedings of the 30th Annual Meeting of the ACL*, pages 183–190. ACL.

E. Riloff and R. Jones. 1999. Learning Dictionaries for Information Extraction by Multi-level Boot-strapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, pages 1044–1049. The AAAI Press/MIT Press.

J. Robin. 1994. *Revision-Based Generation of Natural Language Summaries Providing Historical Background: Corpus-Based Analysis, Design, Implementation, and Evaluation*. Ph.D. thesis, Department of Computer Science, Columbia University, NY.

S. Siegel and N.J. Castellan. 1988. *Non Parametric Statistics for Behavioral Sciences*. McGraw-Hill.

J. Veronis, editor. 2000. *Parallel Text Processing: Alignment and Use of Translation Corpora*. Kluwer Academic Publishers.

R. Wechsler. 1998. *Performing Without a Stage: The Art of Literary Translation*. Catbird Press.

D. Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pages 189– 196.