Institut für Theoretische Informatik
Fakultät für Informatik
Universität Karlsruhe (TH)

Language Technologies Institute
Carnegie Mellon University, Pittsburgh, USA

Integrating Paraphrasing Features into Statistical Machine Translation

Studienarbeit
von
Matthias Bracht

May 22, 2007

Advisors: Prof. Dr. Alex Waibel
Dr. Stephan Vogel

Acknowledgements

I would like to express my utmost gratitude to all those who helped me write this paper. To begin with, I would like to thank my advisor, Dr. Stephan Vogel, for being a source of constant support and inspiration. Furthermore, I am most grateful to the whole Statistical Machine Translation group at Carnegie Mellon University for their numerous suggestions and ideas. Finally, I wish to thank the interACT center for funding my stay in Pittsburgh.

Erklärung

Hiermit erkläre ich, die vorliegende Arbeit selbstständig erstellt und keine anderen als die angegebenen Quellen verwendet zu haben.

Karlsruhe, den 22. Mai 2007 .... Matthias Bracht ....

Abstract

This report presents various applications of paraphrasing features within a state-of-the-art SMT system. We show how the task of paraphrasing can be viewed as a monolingual translation process. While recent work has shown the usefulness of paraphrasing source sentences under the condition of only scarce bilingual data being available, we examine the benefits of paraphrasing source sentences under the scenario of an abundance of bilingual data. In addition, results of experiments using additional paraphrased reference sentences for training are presented. Although these techniques do not lead to significant increases in BLEU scores under the examined scenarios, we argue, having observed initial promising results from our novel idea of matching longer phrases, that we can expect more substantial improvements in other cases from the integration of paraphrasing features into SMT.

Contents

List of Figures

List of Tables

1 Introduction

Statistical machine translation (SMT) as the currently leading solution for the task of automatic translation of large vocabulary text heavily depends on the amounts of data used in the process. While bilingual corpora and reference sets are available to train corresponding models, their size and structure also determine the consequent performance and automatic evaluation of SMT systems. Therefore, it can be helpful to either simplify or add more variety to data provided for the task of statistical machine translation automatically.

Recent progress in the field of paraphrasing has stimulated research on how to apply paraphrasing techniques in order to improve SMT quality. Improvements were achieved by including paraphrasing techniques at different stages of the translation process, namely by paraphrasing whole corpora before training, by paraphrasing source input before the actual translation task, by paraphrasing system output on the target side, or by generating additional paraphrased reference sentences or even whole new metrics which take paraphrase information into account for evaluation.

After giving an overview on statistical machine translation and the state-of-the-art system used for our experiments in Chapter 2, we summarize recent work in the field of paraphrasing and its applicability for the statistical machine translation task in Chapter 3. We then present our experimental setup for the incorporation of three interesting approaches into our system in Chapter 4: viewing paraphrasing as a monolingual translation task in Section 4.2, using paraphrases of source phrases for lattice input, which opens up the novel idea of matching longer phrases that were never seen in the training data before, in Section 4.3, and generating additional paraphrased references for more reliable training in Section 4.4. The results of our work can be found in Chapter 5, implying that under a scenario of large amounts of parallel data being available and paraphrases being learned from the same data, BLEU scores do not rise significantly; however, even with this unfavorable setup, numerous examples for the successful application of the novel concept of matching longer phrases are observed. We conclude with an assessment of our approaches and an outlook on potential further improvements.

2 Statistical Machine Translation

The ongoing process of globalization along with the exponential growth of computing power has gradually fueled the interest in using computers for the expensive and complicated task of translating text from one language into another. Until the 1990s, all approaches to the task of machine translation (MT) were based on so-called expert systems that tried to capture linguistic knowledge in manually generated sets of rules that were then applied to source language text for translation. However, these expert systems are expensive to build, hard to maintain and difficult to port to new domains and languages. Statistical machine translation, initially proposed by IBM researchers in [BPPM93], attacks these problems by breaking out in a different direction, i.e. training statistical models on large amounts of data, monolingual text in the target language as well as bilingual parallel corpora containing translations of source language text in the target language.

2.1 Overview

In general, when translating a source language text f into a target language text e, the resulting translation is the sentence that maximizes the probability of what f translates into:

$$\hat{e} = \arg\max_{e} p(e|f) \tag{2.1}$$

Using Bayes' Theorem, we get:

$$\hat{e} = \arg\max_{e} \frac{p(f|e)p(e)}{p(f)} = \arg\max_{e} p(f|e)p(e) \tag{2.2}$$

Thus, linguistic knowledge can be divided into probabilities yielding from a translation model $p(f|e)$, built on a correlation analysis of jointly occurring patterns in source and target language, and a language model $p(e)$, mapping the probability of the sequence e appearing in the target language. These models are first trained separately, then their combination parameters are optimized. After that, the actual translation takes place[1] and is finally evaluated.

While initial SMT systems were word-based, the basic unit of translation was recently extended to phrases, leading to distinct improvements in translation quality: Phrases allow for $n \times m$ alignments, make local reordering priceless and facilitate the learning of local context. Also, segmentation errors can be corrected.

2.2 The CMU SMT System

In order to examine the use of paraphrasing in the area of statistical machine translation , we use the phrase-based state-of-the-art CMU SMT system, relying on the log-linear approach (published in [ON02]), which has recently replaced the maximum likelihood optimization used before. This approach combines several models through feature functions, each of which assigns a raw score to a candidate hypothesis for a certain model. The raw scores h_i are weighted by scaling factors λ_i and then added up for an overall score of a hypothesis. The decoding task becomes the question of finding a target sentence that maximizes the following formula:

[1] Translation is also called *decoding*, stemming from the application of the noisy channel model to the translation process: By removing noise, one can decode the channel output (i.e. the sentence to be translated) to come up with the channel input (i.e. the translation).

$$\hat{e} = \arg\max_{e} \sum_{i=1}^{n} \lambda_i h_i(e, f) \tag{2.3}$$

Eight of these feature functions are used in the basic system with which we evaluated our approaches, integrating scores from a language model, a distortion model, word count and phrase count penalties, and four translation probabilities (lexical and phrase translation probabilities for both translation directions). The scaling factors λ_i were optimized by minimum error rate (MER) training, described in [Och03]. Consequently, we were able to assess the helpfulness of new features by integrating them through additional feature functions. For further details about the CMU SMT system, consult [VZH$^+$03].

3 Related Work

The task of finding paraphrases forms its own research field - however, only some of the approaches taken in order to find paraphrases promise to be useful with the goal of their application in statistical machine translation in mind.

3.1 Strategies for Finding Paraphrases

Summarizing past work on acquiring paraphrases, we can distinguish between using dictionaries on the one hand and data-driven approaches on the other hand. Since dictionaries like WordNet mostly work on word level, there is a natural constraint for their usefulness in the area of phrase-based SMT. The data-driven approaches can be further subdivided by the use of either monolingual or bilingual corpora. Multiple translations of a novel in the same language serve as the data source for finding paraphrases in [BM01], where contextual and paraphrase classifiers are trained based on similarity in local context. In [PKM03], multiple reference translations for the evaluation of translation quality are used to build finite state automata based on the results of syntactic parsing, so that alternative paths in such automata come to signify paraphrases of each other. Especially in fields such as Question Answering and Information Extraction, Named Entities (NE) can also be used to find paraphrases, such as in [Sek05], where the assumption is made that in different news articles about the same events, words used around the same Named Entities are likely to be paraphrases of each other. Similarly, as presented in [BL03], phrase patterns can be learned that tend to be paraphrases if they take the same arguments in different descriptions of the same events. In [DB05] and [BD05], heuristic extraction techniques and SVM-based classifiers are used on monolingual news articles.

Most of these techniques prefer paraphrase quality over quantity, reporting relatively low numbers of found paraphrases. Bilingual parallel corpora are taken in [BCB05] to find a much larger number of paraphrases - this idea is presented in more detail in Section 4.1, because it serves as the basis for our approach.

3.2 Paraphrases in SMT

Recent work has described the use of paraphrasing features at different stages of the decoding process, be it on the whole training corpus, on the source side or the target side of test sets, or for evaluation. In [WSS02] a reduction in Word Error Rate is reported along with an improvement of subjective evaluation when normalizing whole corpora by means of replacing phrases with their most frequent paraphrases, thereby reducing vocabulary size and facilitating parameter estimation in the training phase. The paraphrases found by the technique presented in [BCB05] are successfully used to paraphrase source input in [CBKO06]. While also paraphrasing the actual translation output on the target side later on, [Yam02] propose a controller instance between a paraphraser and the actual language transfer so that only those parts of a source input sentence are paraphrased that actually help the decoder with the translation task.

As far as evaluation is concerned, it is difficult to compare translation quality employing current metrics like BLEU ([PRWZ02]), because BLEU only considers n-grams matching any of the reference translations as correct. Therefore, semantically correct translations might be judged as bad on sentence level if their n-grams are not contained in the corresponding reference sentences, a fact also stated in [CBOK06]. This harms the translation process when optimizing toward a higher BLEU score in the training phase as well as in the stage of final evaluation of translation output. Thus, concerning the training phase, a performance improvement is reported in [MARD07] when training the hierarchical SMT framework HIERO, which learns synchronous context-free grammar rules, with additional paraphrased references, while still evaluating only with given references. With this technique, better scaling factors are found in the training phase. Addressing the evaluation phase, the inclusion of paraphrase features into evaluation measures in [RLLR05] leads to a significantly higher correlation with human judgment compared to existing substring matching techniques. Paraphrasing reference sentences to make them more similar to a given system output sentence has been examined by [KB06]. [OGGW06] argue that the automatic creation of additional paraphrased reference sentences for evaluation beforehand by using different parts of source reference sentences aligned with the same part of a target reference sentence as paraphrases correlates better with human judgments as well.

The lack of paraphrasing support in BLEU has led to the development of a variety of new metrics such as METEOR, described in [BL05], where, in addition to an "exact" module that links identical words, and a "porter-stem" module that links words that do not differ in their stems, a "WordNet synonymy" module is employed so that synonymous translations of words do not get penalized. Another recently proposed metric called ParaEval, described in [ZLH06], also allows for paraphrase matches in addition to lexical matches.

4 Our Paraphrasing Approaches

Our method of finding paraphrases relies on the findings presented in [BCB05], because this method finds the largest number of paraphrases, using techniques also found in the statistical machine translation field. We then apply these paraphrases to whole sentences by handling paraphrasing as a monolingual translation task. Furthermore, we examine and refine ideas stated in [CBKO06] to paraphrase source input sentences before decoding and concepts out of [MARD07] to generate additional references for better optimization in the training phase.

4.1 Finding Paraphrases

Having large bilingual corpora available, we can assume that both a phrase e_1 and its paraphrase e_2 are likely to be translated as the same phrase f in the target language. This means that the SMT system will probably have learned translational alignments for both e_1 and its paraphrase e_2 with f. Therefore, we can use the target language as pivot to determine a paraphrase probability by multiplying $p(f|e_1)$, the probability that a source phrase e_1 translates into a target phrase f, with $p(e_2|f)$, the probability that this target phrase f translates back into a source phrase e_2, and then summing over all target phrases f aligned with e_1 and e_2. These translation probabilities can be taken from a phrase table generated with the Pharaoh package described in [Koe04]. Thus, the probability that a phrase e_2 is a paraphrase of e_1 can be computed with the following formula:

$$p(e_2|e_1) = \sum_f p(f|e_1)p(e_2|f) \qquad (4.1)$$

See Table 4.1 for an example of how a phrase in another language is used as pivot.

```
$PHARAO # answer # respuesta # 0.312589 ...  0.641541 ...
...
$PHARAO # response # respuesta # 0.350133 ...  0.75412 ...
...
(Format: $PHARAO # e # s # p(e|s) ...  p(s|e) ...)
```

Table 4.1: Sample entries from a Pharaoh phrase table. The lexical probabilities are not shown, because they are not considered for the purpose of paraphrase generation. The product of the probabilities shown in color contributes $p(respuesta|answer) * p(response|respuesta) = 0.641541 * 0.350133 = 0.2246$ to $p(response|answer) = 0.2335$.

The paraphrases and probabilities found this way can then be stored in a paraphrase table for our following experiments. However, the initial paraphrase table is much larger than usual phrase tables used for translation. Therefore, we have to concern ourselves with pruning techniques. First of all, we choose to ignore phrase pairs e_1 and e_2 with $e_1 == e_2$, because we do not gain any new information from phrases being paraphrased into themselves. We also perform *absolute value pruning* and discard all entries with $p(e_2|e_1) < 0.001$, because phrase pairs with such low paraphrase probabilities hardly ever constitute sensible paraphrases. Obviously, this drastically reduces the size of the paraphrase table, as does disregarding rare paraphrases e_2 only seen once (*singleton pruning*, i.e. $c(e_2) == 1$).

In addition, we also leave out paraphrases that are too frequent, referring to the concept of stop words in information retrieval. Due to their high frequency, words like "*the*" and "*of*" will be aligned with large numbers of phrases, although they rarely constitute sensible paraphrases. Ignoring such potential paraphrases that occur more often in the training corpus than a pre-determined threshold count is a concept we call *stop phrase pruning*.

In order to further reduce the number of translation hypotheses considered in the decoding process, we do not need to consider every single possible paraphrase for a given source phrase, but only the best ones. Therefore, we can choose to only consider paraphrases within a beam around the best paraphrase probability[1], which we call *beam pruning* in the following sections.

Exploiting these pruning techniques by discarding paraphrases that fulfill any of the pruning criteria mentioned above, we can shrink the paraphrase table down to a manageable size without harming translation quality later on.

4.2 Paraphrasing Sentences

A first assessment of the usefulness of the acquired paraphrases can be made by viewing paraphrasing as a process of monolingual translation: We can use the paraphrase table generated above to paraphrase sentences by considering it as one part of the phrase table[2] - the other part would enable the decoder to leave a source word as it is with a paraphrase probability of 1[3].

The paraphrased sentences are then compared to reference sentences, because in this case we can optimize the translation parameters toward generating better paraphrased sentences using MER training and BLEU ([PRWZ02]) as the evaluation metric. Reference sets containing multiple translations come in handy for this: We refine the idea of creating additional paraphrased references stated in [MARD07][4] by translating a

[1] A paraphrase is discarded if its probability is smaller than the one of the best paraphrase for the corresponding source phrase, multiplied by a pre-determined beam size factor.

[2] Consequently, we optimize one phrase table scaling factor instead of the four mentioned in 2.2 with this setup, resulting in a total of five feature weights.

[3] As we show in Section 5.5, while dropping this part leads to much lower BLEU scores, under certain conditions this technique actually renders paraphrases that differ a lot from the input, which is exactly what we want when generating more references.

[4] They randomly chose the sentences which are to serve as source sentences.

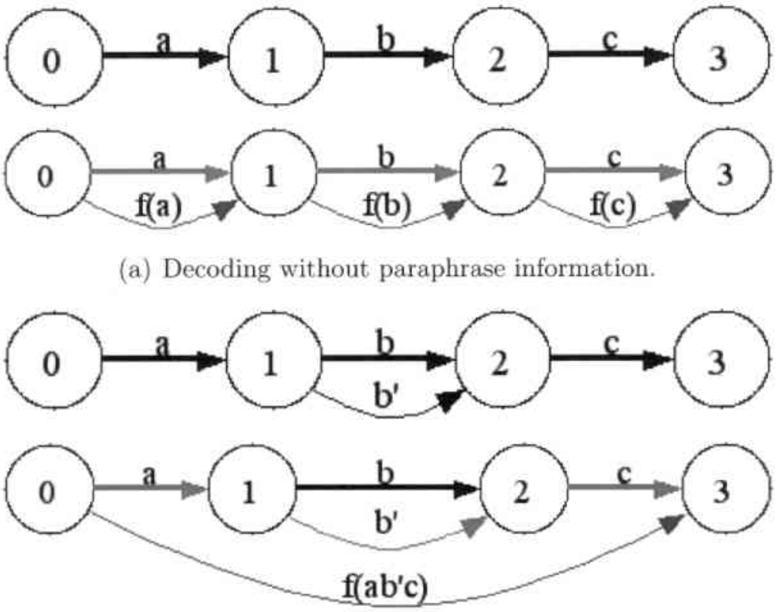
set of those references that are evaluated worse by a language model[5], with the "better" references as the reference set. We evaluated the parameters that we optimized on development references on unseen test references, once again paraphrasing the "worse" references and evaluating against the "better" ones. The paraphrased references can then be used for training, as described in Section 4.4.

4.3 Embedding Paraphrased Input into the CMU SMT System

While our approach to paraphrasing for translation is similar to the one illustrated in [CBKO06], we do not mainly focus on translating previously unknown words, since this is only a problem under the scenario of scarcity of bilingual data. Instead, our approach differs in the sense that we examine to what extent paraphrasing can still be of help when using large amounts of data. This means that we cannot expect to gain much by reducing the number of unknown source words (since the system knows them already from the training phase). However, by offering the system alternative formulations of the same source input sentence, we can hope to match longer phrases as in the following example: If a sentence contains the sequence *abc* and the system has not seen this sequence before, it would decode this sequence in three parts, although it might know a translation of the sequence *ab'c*, with *b'* being a paraphrase of *b*. By replacing *b* with *b'*, we could thus match longer phrases in the translation process and thereby come up with more reliable translations. See Figure 4.1 for an illustration of this idea.

Our method of using paraphrases introduces a certain level of variability on the source side and possibly even some grammatically incorrect input alternatives - however, since the decoder assigns language model scores to the consequent translation hypotheses, we work under the assumption that for our purpose, slight errors in paraphrasing exactitude caused by our relatively loose definition of what makes up a paraphrase will be corrected by the language model scores.

[5] We used the SRI language model described in [Sto02].



(a) Decoding without paraphrase information.

(b) Decoding with paraphrase information.

Figure 4.1: 4.1(a) shows how the decoder would translate the three phrases *a*, *b* and *c* separately into *f(a) f(b) f(c)* without paraphrase information. If the information is available that *b'* is a paraphrase of *b* and the phrase *ab'c* is known instead, a longer phrase can be matched and *ab'c* can be translated into a single target phrase *f(ab'c)* in 4.1(b).

Thus, instead of only using source sentences as input for the SMT system, we allow for paraphrases of source phrases to be used in place of the original source phrases as well. While in [CBKO06] this is done by augmenting the phrase table with all the previously unknown translations of all the translations of paraphrases[6], we choose to utilize lattice input instead of sentence input. Lattice input is usually employed when translating speech: In the Automatic Speech Recognition (ASR) task,

[6] With that approach, it is not possible to match longer phrases, because the phrase table is only augmented with new translations that stem solely from paraphrases. A translation of a phrase that is made up partly of original phrases and partly of paraphrases as suggested in our example is therefore not contained in the phrase table.

one tries to determine before the actual translation phase "what has actually been said", creating a lattice from audio input. An ASR lattice is a directed graph, the nodes being spaces between utterances and the edges labeled with words and their corresponding acoustic scores that try to capture how likely it is that a word has really been said[7]. In our case, however, we do not use acoustic scores, but paraphrase probabilities. An example of such an input lattice can be found in Figure 5.5. This enables the system to match longer phrases as mentioned before. In the example in Figure 4.1, [CBKO06] would offer additional translations for *b*, but none for *abc* by paraphrasing *b* through *b'*.

Just as we would proceed with audio input, we use an additional ninth feature function (cf. Section 2.2) for paraphrased input so that the system can learn the importance of paraphrased input during MER training. A lattice edge belonging to a word w is assigned a score depending on whether w originates from the original input sentence or from a paraphrase as follows, using the paraphrase probabilities computed with Equation 4.1:

$$lattice\_score(w) = \begin{cases} 1 & \text{if } w \text{ is from the original sentence,} \\ \sqrt[n]{p(e_2|e_1)} & \text{if } w \text{ is part of a paraphrase } e_2 \\ & \text{of length } n \text{ of a source phrase } e_1. \end{cases} \tag{4.2}$$

This way a paraphrase probability for a phrase of n words is split evenly over the corresponding n edges in the lattice. Using this setup, we can use minimum error rate training described in [Och03] in order to optimize the scaling factor for the importance of paraphrases without making any additional assumptions beforehand about how likely the use of paraphrases is compared to the use of source phrases.

[7] Our decoder also works with lattices internally, generating new edges for all possible translations found in the phrase table and then finding the path with the lowest costs as the translation result.

4.4 Generating More References for Training

While the previous section concerned itself with paraphrasing on the source side, we also examine a promising application of paraphrasing in the target language, i.e. the generation of more references for the training phase, an idea suggested in [MARD07]. We use the techniques mentioned in Section 4.2 and take the first n-best list entries that differ from the human reference[8] as additional references.

Since the initial approach allows for words to stay unchanged with a probability of 1 and since it therefore rather rarely favors paraphrases, we also examine a different and rather drastic approach to reference generation. In order to favor paraphrases of higher variety, we train the system again, this time leaving out the part of the phrase table that allows the decoder to leave words unchanged with a paraphrase probability of 1[9].

Following good engineering practices, we set up a baseline by first training the system without any additional references and then evaluating with the scaling factors thus derived. To test our approaches, we supply different numbers of additional reference sentences generated by paraphrasing to train the system and then evaluate again with only the original evaluation references given.

[8] If there is one. Otherwise the unchanged human reference is added again.

[9] Considering paraphrasing as a monolingual translation process, the problem of unknown words is not as severe as in the bilingual case. Words for which no translations were learned will just remain untranslated - this is an emergency solution for bilingual translation, but an acceptable option when paraphrasing.

5 Results

In order to make it easier for readers to understand the paraphrased sentences and to judge their usefulness, paraphrasing has always been conducted on the English side, i.e. sentences have been paraphrased with a paraphrase table as the phrase table in English language; paraphrasing sentences on the source side has been done for the task of translating English to Spanish, and the generation of more references on the target side for the task of translating Spanish to English.

5.1 Data Sets

The data we used originates from the Spanish EPPS Verbatim task of the 2006 TC-STAR project (Technology and Corpora for Speech to Speech Translation)[1] and mainly contains publicly available transcriptions of European Parliament Plenary Sessions (EPPS). The characteristics of the data used for the specific tasks can be seen in Table 5.1.

5.2 Paraphrases Found

When generating paraphrase tables for all n-grams contained in specific test sets, the initial paraphrases tables were huge. Therefore, we performed (in that order) absolute value pruning with a threshold of 0.001, singleton pruning, stop phrase pruning with a threshold phrase count of $50,000$ in the training corpus, and beam pruning with beam sizes of 0.1, 0.25, 0.5, 0.75 and 1.0. We also generated a paraphrase table

[1] http://www.tc-star.org.

		English		**Spanish**	
		Sent.	Words	Sent.	Words
Train		1247314	35748618	1247314	37458054
		English		**English**	
		Sent.	Words	Sent.	Words
Dev **5.3**		1712	53994	1712	56525
Eval **5.3**		897	30289	897	31863
		English		**Spanish**	
		Sent.	Words	Sent.	Words
Dev **5.4**		1194	30255	2388	63344
Eval **5.4**		1155	30486	2310	62022
		Spanish		**English**	
		Sent.	Words	Sent.	Words
Dev **5.5**	H_1	1712	54263	1712	56185
	H_1Q_1	...	...	3424	112121
	H_1V_1	...	...	3424	110841
	$H_1Q_1V_1$	...	...	5136	166677
	H_1H_2	...	...	3424	110519
	$H_1H_2Q_1Q_2$	...	...	6848	220381
	$H_1H_2V_1V_2$	...	...	6848	218818
	$H_1H_2Q_1Q_2V_1V_2$	...	...	10272	328680
Eval **5.5**		897	30246	1794	62152

Table 5.1: Characteristics of corpora for training (Train), development (Dev), evaluation (Eval). The actual phrase tables used for decoding were a paraphrase table in Section 5.3 and bilingual phrase tables for all n-grams contained in the input for the other tasks (cf. Section 5.5 for the explanation of the different reference sets in the final development phase).

with no beam pruning, but absolute value pruning instead with a threshold of 0.1. See Table 5.2 and Figure 5.1 for the effects on paraphrase table size[2].

Paraphrase lengths after different pruning stages are shown in Figure 5.2, illustrating that most of the paraphrases are two words long. Logically, the number of different phrases for which paraphrases are kept does not change anymore after the stop phrase pruning stage if beam pruning is applied, because this strategy will always keep at least one paraphrase for each phrase, i.e. the best one.

Pruning Strategy	**Entries**	**#Phrases (Avg Length)**	**#Paraphrases (Avg Length)**
None	158324576	39760 (2.80)	2038753 (3.51)
$p > 0.001$	2178730	39759 (2.80)	716975 (3.40)
$c(e_2) > 1$	2021393	38940 (2.78)	580042 (3.24)
$c(e_2) < 50000$	1914515	38938 (2.78)	579949 (3.24)
Beam 0.1	404022	...	187940 (3.12)
Beam 0.25	178476	...	98567 (3.04)
Beam 0.5	89156	...	55417 (2.96)
Beam 0.75	56448	...	37232 (2.89)
Beam 1.0	42011	...	29222 (2.87)
$p > 0.1$	32532	22645 (2.89)	23576 (2.90)

Table 5.2: Characteristics of the paraphrase table created for evaluation in Section 5.4. Each pruning step was performed in addition to the ones given in the lines above, except the final one, which bypassed the beam size pruning stages.

The results justify the fact that we carried out beam pruning only *after* stop phrases were deleted: If we performed beam pruning first, we would sometimes only keep stop phrases, which would then be pruned out in the later stop pruning phrase, leaving us with no paraphrases at all for the phrases concerned. Although stop phrase pruning does not shrink the table significantly, it greatly improves quality: Stop phrases constitute the highest-ranked paraphrase for more than 5% of the phrases

[2] Within our approach, paraphrase tables are generated specifically for a set of sentences, containing paraphrases only for matching n-grams in a test set.

```
$PARA # on joint # in the # 0.154303
$PARA # on joint # in the combination of # 0.0833332
...
$PARA # on legal # on the # 0.267931
$PARA # on legal # legal # 0.127191
```

Table 5.3: These examples show the importance of stop phrase pruning before beam pruning. In the stop phrase pruning stage, the colored entries will be dropped in favor of paraphrases with lower probabilities, but higher semantic equivalence.

in the table and are mostly judged worse by human experts than the paraphrases with lower probabilities. See Table 5.3 for examples.

In general, our approach delivers paraphrases and corresponding probability rankings that intuitively make sense. A sample extracted from a paraphrase table can be seen in Table 5.4. While most of the entries in that extract would be considered valid, note the disturbing example of a best-judged paraphrase that actually means just the opposite in the first line[3].

Since we do not include any syntactic or semantic information in our approach thus far, some paraphrases still contain incorrect or "problematic" paraphrases. For example, potential paraphrases can perform a different syntactic function and should not be considered if we are looking for exact paraphrases, as in Table 5.5. Additionally, the first line shows an example of a paraphrase which is an antagonym[4]: The best paraphrase suggested for the word *"abandoned"* can cause errors, since in the sentence *"He's left"*, *"left"* can both refer to the concept of *"still remaining"* as well as *"having gone"*. Including semantic information would be desirable in this case, because paraphrasing can add ambiguity to previously precise sentences. Also, phrases that are yet to be paraphrased can be ambiguous or serve as both nouns and verbs, as in Table

[3] This particular example was generated by English sentences about "having a bad conscience" and corresponding Spanish translations expressing "not having a clear conscience", and is therefore a result of alignment errors.

[4] An antagonym is a word that can also mean its exact opposite.

5.6. Therefore, syntactic or semantic information about the phrase to be paraphrased would enable us to filter the candidate paraphrases further.

5.3 Paraphrased Sentences

The BLEU scores achieved when translating the set of evaluation references judged worse by a language mode, evaluating against the better-ranked sentences and with parameters optimized on the identical setup using development references, can be seen in Table 5.7. The more paraphrases were considered, the higher the BLEU scores generally were, improving by 0.4 BLEU points when using low beam sizes or no beam at all, considering all paraphrases left after absolute value pruning with a threshold of 0.001.

While we are aware of the problems that lie in the evaluation of paraphrasing results with the BLEU metric and although the computation of BLEU scores might be considered an academic exercise on this setup, the results still show that to a certain extent, the system has learned to make a sentence more similar to the other reference translation. Our approach obviously depends on the degree in which multiple reference sentences of a single source sentence differ from each other. The more similar the reference sentences are, the less the system will learn to paraphrase. 82 out of 1712 times, the two reference translations given for a source sentence were completely identical, punishing all paraphrasing attempts. This also led to the observation that for our first approach that allowed the decoder to leave words unchanged with a paraphrase probability of 1, 64 of the first-best hypotheses did not differ from the input, and the other best hypotheses mostly varied by one phrase only.

The second approach, which does not allow the decoder to leave words unchanged at no cost[5], led to much lower BLEU scores because of fewer n-gram matches when compared to a rather similar reference sentence. However, most of the paraphrases generated this way were not only

[5] Since the system works with a log-linear model and costs given by negative logarithmic probabilities, a probability of 1 is transformed into a cost of $-\log 1 = 0$.

judged valid by human inspection, but also differed to a much larger degree from the input. Using a beam size of 0.5 again, the system generated new sentences as the first-best hypothesis in all cases, making them considerable candidates for additional references with the goal of adding more variety in mind. See Tables 5.8 and 5.9 for examples to compare the initial approach and the "brute force" paraphrasing technique when they are applied in order to create more references in Section 5.5. A better evaluation of our paraphrases in the training phase would be possible by the integration of metrics that take paraphrases into account.

5.4 Translation with Paraphrased Input

As Figure 5.3 illustrates, the generated paraphrase lattices grow much larger the less pruning is applied. Consequently, the size of the set of translation hypotheses is immense due to combinatorial explosion caused by alternative paths and, for each path, numerous translation candidates[6]. This requires a strong degree of pruning for experiments and therefore a low number of considered paraphrases because of memory restrictions. Example lattices for our absolute value pruning approach as well as different beam sizes can be seen in Figures 5.4 to 5.8. Note that in the paraphrased lattices, some paths would lead to grammatically incorrect sentences, which could be remedied by the integration of a source language model. Also, the number of content words can change[7] or additional articles are inserted[8].

As Table 5.10 shows, the paraphrased input using paraphrases derived from the same corpus that was also used to train the translation system does not lead to significantly higher BLEU scores. This correlates with the results in [CBKO06] stating that paraphrasing becomes less helpful

[6] Although to the decoder, the number of possible paths through the lattice matters more than only the number of edges, we confine ourselves to edge numbers in Figure 5.3. The number of resulting translation hypotheses is immensely high and, for higher beam sizes, surmounts 2^{32}.

[7] *"lie ahead"* is paraphrased into *"ahead"* in Figure 5.5.

[8] *"real challenges"* becomes *"the real challenges"* in Figure 5.5.

the more parallel data the translation system is already trained with. In fact, the decoder learns to ignore paraphrases by setting the paraphrase scaling factor much higher than all the other ones[9]. However, Table 5.10 also demonstrates that in spite of those high scaling factors, paraphrases still formed parts of the lattice paths of first-best translation hypotheses in numerous cases. We expect this to show more strongly if we use different paraphrase sources or fewer parallel data. See Figures 5.9 to 5.11 for examples of varying quality in which longer phrases are matched when paraphrases are considered, proving that our idea of matching longer phrases with lattice input works in general.

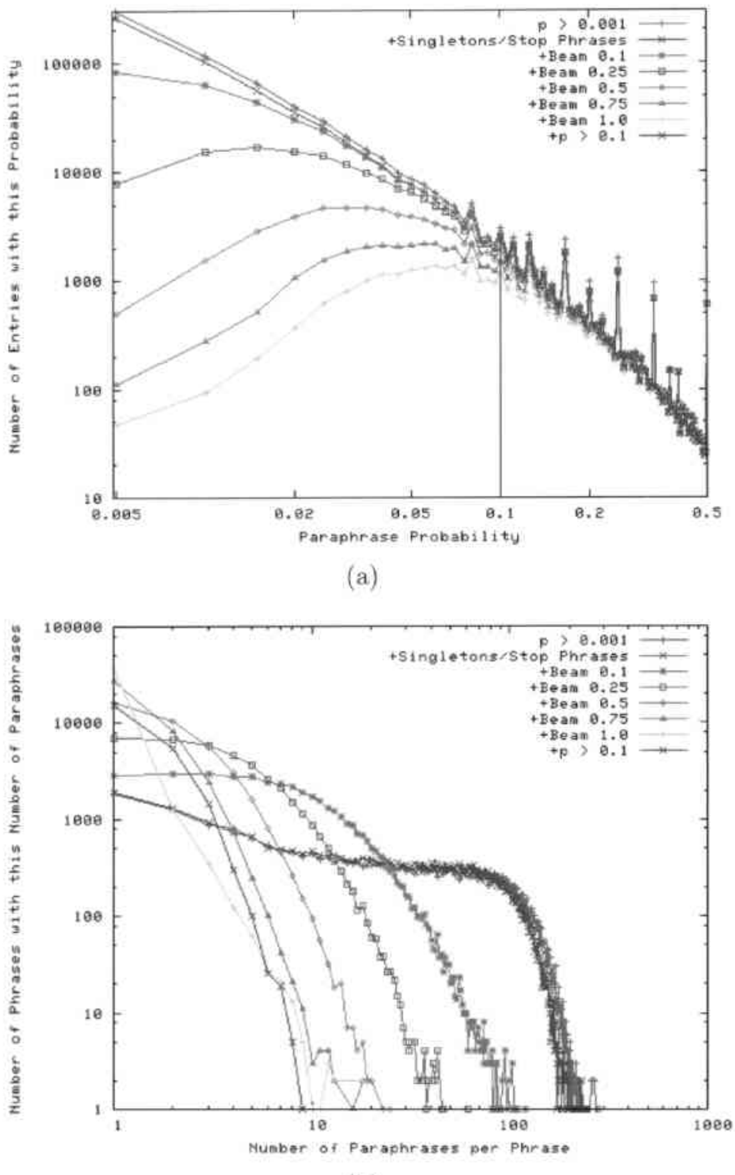
Especially Figure 5.8 offers an idea how paraphrased lattices could be compacted further: For example, all the final edges are labeled with the period at the end of a sentence, differing only in edge scores. These edges could be combined to form only one edge by merging all the second-to-last nodes into one node and backpropagating scores to the incoming edges of those nodes. Similar techniques could be applied at other places in the lattice[10]. Figure 5.3 also shows how the longest sentences set the upper boundary of beam sizes for which decoding is still possible without memory concerns. Therefore, instead of a fixed beam size, a flexible beam size should be used in future experiments depending on the length of each sentence in order to flatten the curves given in the figure. In that case, more paraphrases for shorter sentences and fewer paraphrases for longer ones could be considered so that the available memory would be fully used for each sentence.

[9] Example set of scaling factors for the run with a beam size of 0.75, the last of which being the paraphrase scaling factor, trained with an upper boundary of 5.0: 0.26308_0.66270_ − 0.20954_0.06810_0.06260_0.09565_0.09039_0.07651_4.83901.

[10] e.g. to partially merge the top six paths going out of node 3.

```
$PARA # a bad conscience # a clear conscience # 0.122222
$PARA # a bad conscience # guilty conscience # 0.111111
$PARA # a bad conscience # ugly conscience # 0.104377
$PARA # a bad conscience # bad conscience # 0.0993265
$PARA # a bad conscience # a guilty conscience # 0.0606059
$PARA # a bad conscience # a good conscience # 0.0222222
$PARA # a balanced text # a well-balanced text # 0.0491071
$PARA # a balanced text # a balanced text , # 0.0357143
$PARA # a balanced text # a balanced document # 0.0245536
$PARA # a balanced text # a balanced report # 0.0245536
$PARA # a balanced text # a text which is balanced # 0.0245536
$PARA # a balanced text # well-balanced # 0.0245536
$PARA # a barbaric # a barbarous # 0.25
$PARA # a barbaric # barbaric # 0.229672
$PARA # a barbaric # brutal # 0.183566
$PARA # a basic law # , a fundamental law # 0.269841
$PARA # a basic law # a basic act # 0.214285
$PARA # a basic law # a fundamental law # 0.190476
$PARA # a behaviour # behaviour # 0.512195
$PARA # a better # better # 0.312956
$PARA # a better # best # 0.0614035
$PARA # a better # greater # 0.0420554
$PARA # a better # improved # 0.0388021
$PARA # a better response # a better answer # 0.207143
$PARA # a big # a great # 0.147579
$PARA # a big # a large # 0.0880404
$PARA # a big # a major # 0.065936
$PARA # a big # great # 0.0406377
$PARA # a big # a serious # 0.0402758
$PARA # a big # a huge # 0.0296231
$PARA # a big # an important # 0.0201535
$PARA # a big hit # a great success # 0.451327
$PARA # a big hit # very successful # 0.0575221
$PARA # a big hit # a major success # 0.0530973
```

Table 5.4: Sample of a paraphrase table pruned with a beam size of 0.1. The entries in color would constitute the paraphrase table generated with a beam size of 1.0.



(a)

(b)

Figure 5.1: Effects of pruning on the paraphrase table used for evaluation in Section 5.4. Figure 5.1(a) illustrates that the various techniques mostly prune out paraphrases with lower probabilities. Figure 5.1(b) shows the number of different paraphrases for a given phrase. The further left a curve is, the fewer alternative paths have to be considered when decoding. Mind the logarithmic scales and the fact that before pruning with a probability threshold of 0.1, no beam pruning was performed.

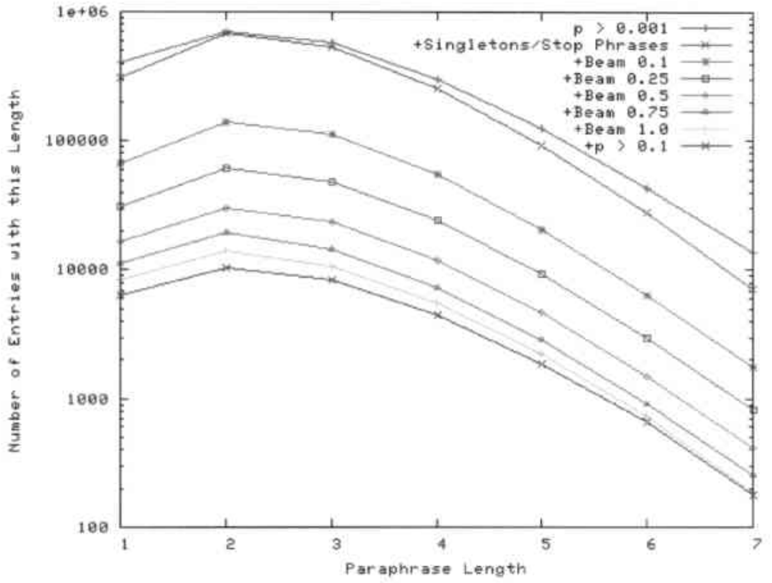


Figure 5.2: Numbers of paraphrase table entries for evaluation in Section 5.4, grouped by paraphrase lengths. Most paraphrases are of length 2. Mind the logarithmic scale and the fact that before pruning with a probability threshold of 0.1, no beam pruning was performed.

```
$PARA # abandoned # left # 0.107052
$PARA # abandoned # leave # 0.0254715
$PARA # abandoned # abandon # 0.0238598
$PARA # abandoned # abandonment # 0.0195465
$PARA # abandoned # given up # 0.0136555
$PARA # abandoned # abandoning # 0.0130148
$PARA # abandoned # neglected # 0.00877858
$PARA # abandoned # leaving # 0.00723795
$PARA # abandoned # dropped # 0.00720887
$PARA # abandoned # been abandoned # 0.00623596
$PARA # abandoned # be abandoned # 0.00584803
```

Table 5.5: Sample of a paraphrase table where syntactic errors come in: While the general meaning is still contained in these paraphrases, tenses and word types differ from the ones of the source phrase. These errors can be avoided by incorporating semantic and part-of-speech information into paraphrase table pruning or by evaluating a paraphrased sentence with a language model.

```
$PARA # answer # response # 0.2335
$PARA # answer # reply # 0.142478
$PARA # answer # respond # 0.0505235
$PARA # answer # solution # 0.0156104
$PARA # answer # respond to # 0.0154792
$PARA # answer # answers # 0.0118446
$PARA # answer # to respond # 0.00983687
$PARA # answer # responding # 0.00911422
$PARA # answer # reply to # 0.00761935
$PARA # answer # answered # 0.00752065
$PARA # answer # reaction # 0.00594168
$PARA # answer # meet # 0.00565372
$PARA # answer # response to # 0.00500908
```

Table 5.6: Sample of a paraphrase table with ambiguity on the source side. Note how depending on whether *answer* is used as a verb or a noun, *respond* or *response* would be a valid paraphrase, but not both.

Paraphrases Considered	BLEU Score
None	0.3910
$p > 0.1$	0.3940
Beam 1.0	0.3931
Beam 0.75	0.3932
Beam 0.5	0.3950
Beam 0.25	0.3937
Beam 0.1	0.3950
No Singletons or Stop Phrases	0.3949
All with $p > 0.001$	0.3939

Table 5.7: BLEU scores achieved in Section 5.3 after different pruning phases.

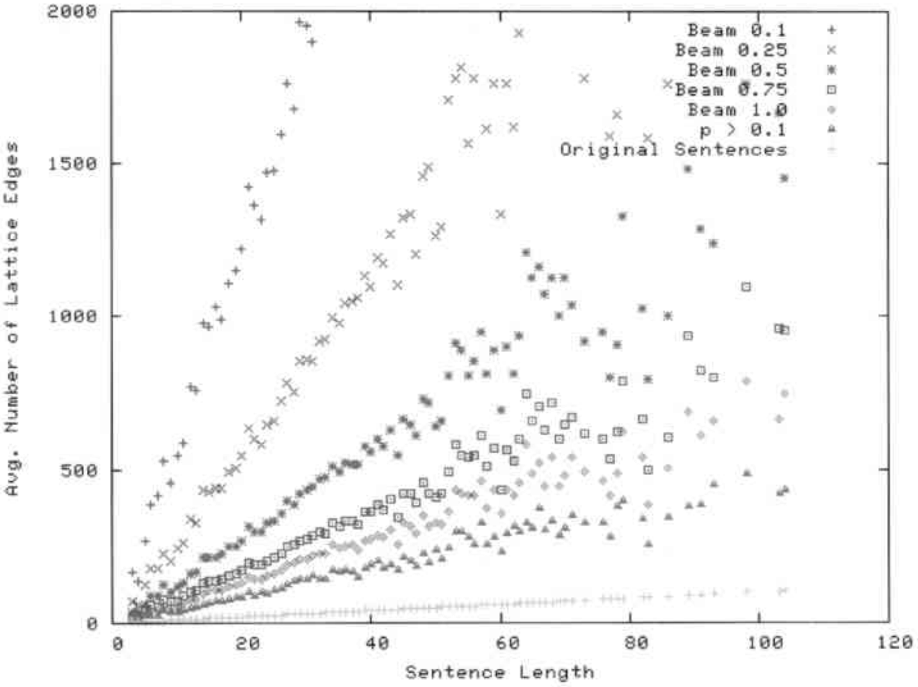


Figure 5.3: Comparison of sentence length and average resulting number of edges for the lattices used for evaluation in Section 5.4.

H_1	I will basically restrict myself to three aspects .
Q_1	I will basically restrict myself to three points .
V_1	I shall essentially confine myself to just three points .
H_2	I shall limit myself , basically , to three aspects .
H_1	we cannot allow this to go on happening and we have to continue supporting the renewal of the fleet .
Q_1	we cannot allow this to go on happening and we have to continue supporting the fleet renewal .
V_1	we must not allow this to continue going on and we must continue to support the fleet renewal .
H_2	we can't let this go on happening , and we have to continue supporting the renovation of the fleet .
H_1	so , I encourage you to keep on investing in the protection of the environment .
Q_1	so , I urge you to keep on investing in the protection of the environment .
V_1	therefore , I urge you to continue to invest in environmental protection .
H_2	this I encourage you to do , to continue investing in environment protection .
H_1	we move on to the next item on the agenda .
Q_1	we move on to the next point on the agenda .
V_1	we shall now proceed to the next point on the agenda .
H_2	we go on with the next point of the agenda .

Table 5.8: Sample first-best paraphrase hypotheses using a beam size of 0.5. H_1 is the human reference used as input, Q_1 the paraphrased output of the system with the first approach emphasizing quality, V_1 the output with the second approach stressing variety, and H_2 the other human reference provided. Notice how in general, V_1 differs a lot from Q_1 and the two human references and, ignoring slight grammatical errors, would be considered a valid paraphrase in these examples.

H_1	let me tell you that your attitude with respect to the subject we are dealing with this morning is shameful .
Q_1	let me tell you that your attitude with respect to the subject we are dealing with this morning is a disgrace .
V_1	I would like to say that its attitude with regard to the issue we are discussing this morning is appalling .
H_2	let me tell you that your attitude is shaming considering the topic we are debating here this morning .
H_1	in order to meet these goals , it will be necessary to have a European textile plan that considers help for re-structuring and specific resources within the framework of Union funds .
Q_1	in order to meet these objectives , it will be necessary to have a European textile plan that considers help for re-structuring and specific resources within the framework of Community funds .
V_1	to achieve these objectives , we must have a textile sector plan to consider aid for re-structuring specific resources in the context of Community funds .
H_2	to meet these objectives , a European textile plan , contemplating restructuring aids and specific resources within the Union's funds frame , will be necessary .
H_1	firstly , we believe that demanding reciprocity is essential .
Q_1	first of all , we think that demanding reciprocity is crucial .
V_1	first of all , we think that demand reciprocity necessary .
H_2	in the first place , we think it indispensable to demand reciprocity .
H_1	our generosity does not go beyond that .
Q_1	our generosity does not go further .
V_1	of our generous it goes no further than this .
H_2	that's how far our generosity goes .

Table 5.9: More sample first-best paraphrase hypotheses. Here, the first approach barely leads to any paraphrasing; the brute force approach introduces new semantically correlated words, but also inexact paraphrases.

	Dev (1194 Sentences)		Eval (1155 Sentences)	
Paraphrases	**BLEU**	**PP Used**	**BLEU**	**PP Used**
None	0.4767	-	0.4697	-
$p > 0.1$	0.4765	153(137)	0.4699	148(123)
Beam 1.0	0.4784	159(150)	0.4660	154(134)
Beam 0.75	0.4791	65(53)	0.4679	47(39)

Table 5.10: BLEU scores achieved in Section 5.4 in the training (Dev) and evaluation (Eval) phase for different pruning strategies. Also, the numbers of sentences are shown in which paraphrases formed part of the path of the first-best hypothesis (in parentheses: numbers of sentences for which the application of paraphrases actually led to a different translation compared to the baseline). Paraphrasing did not lead to higher BLEU scores under this scenario where the paraphrases were learned from the same large amount of data that the translation system was trained with.

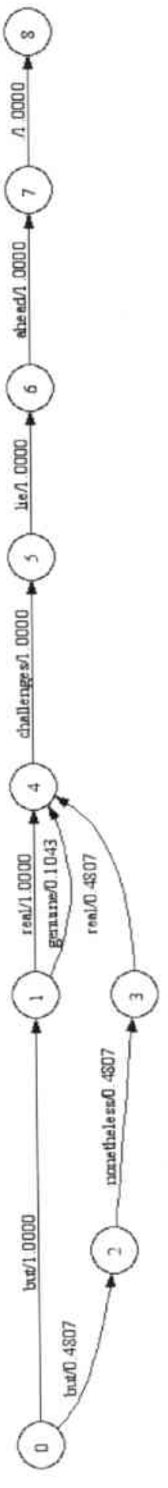


Figure 5.4: Paraphrased input lattice for the sentence *"but real challenges lie ahead ."*, minimum paraphrase probability 0.1. Decoding the sentence *"but genuine challenges lie ahead ."* is now possible.

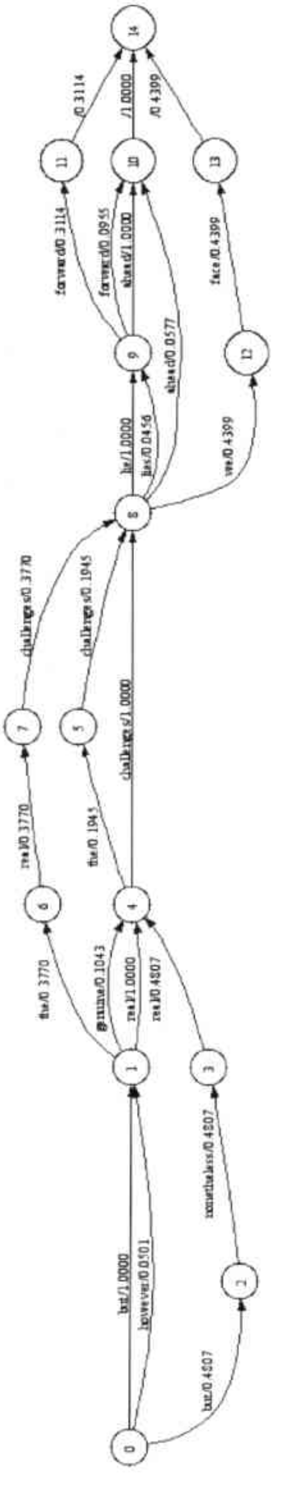


Figure 5.5: Paraphrased input lattice for the sentence *"but real challenges lie ahead ."*, beam size 1.0. Unlike in Figure 5.4, decoding the sentence *"however genuine challenges we face ."* is now possible.

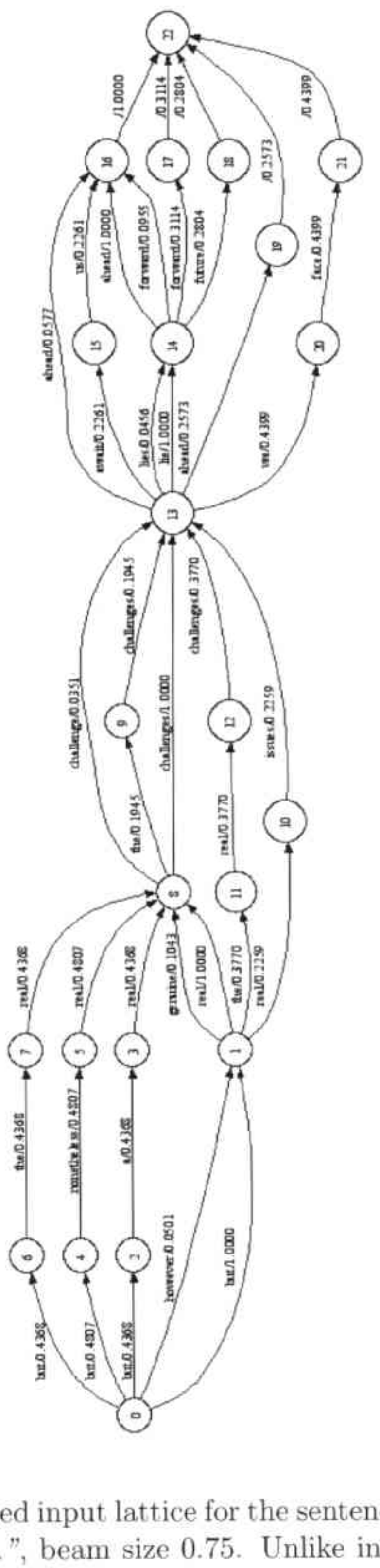


Figure 5.6: Paraphrased input lattice for the sentence *"but real challenges lie ahead ."*, beam size 0.75. Unlike in Figure 5.5, decoding e.g. the sentence *"however , real issues await us ."* is now possible.

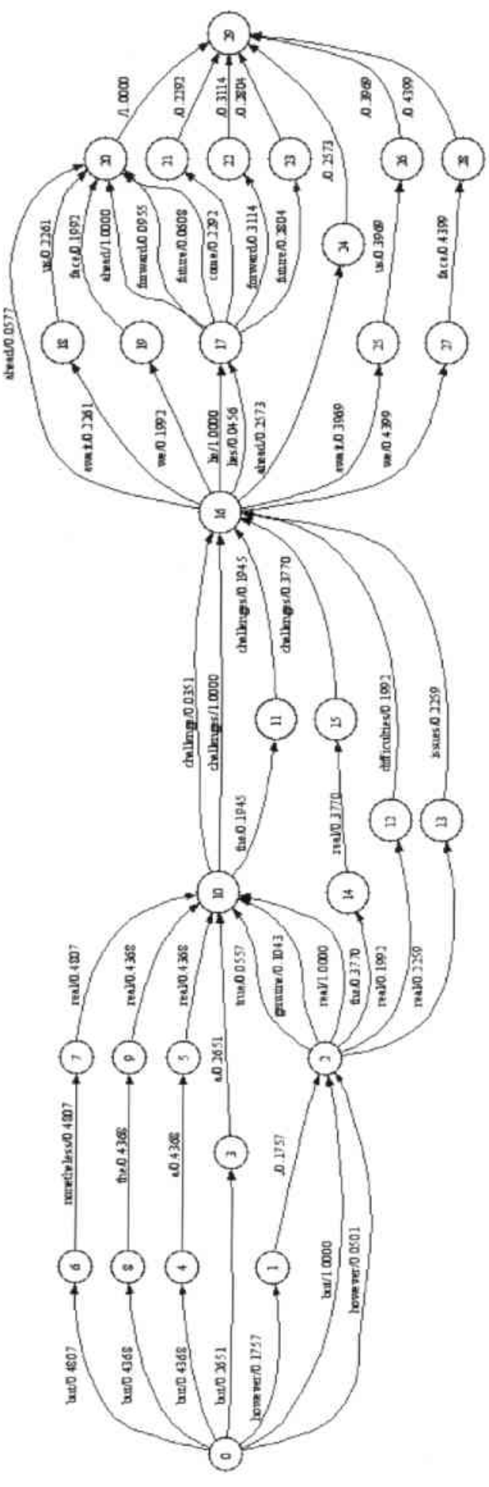


Figure 5.7: Paraphrased input lattice for the sentence *"but real challenges lie ahead ."*, beam size 0.5. Unlike in Figure 5.6, decoding e.g. the sentence *"however , real difficulties ahead come ."* is now possible.

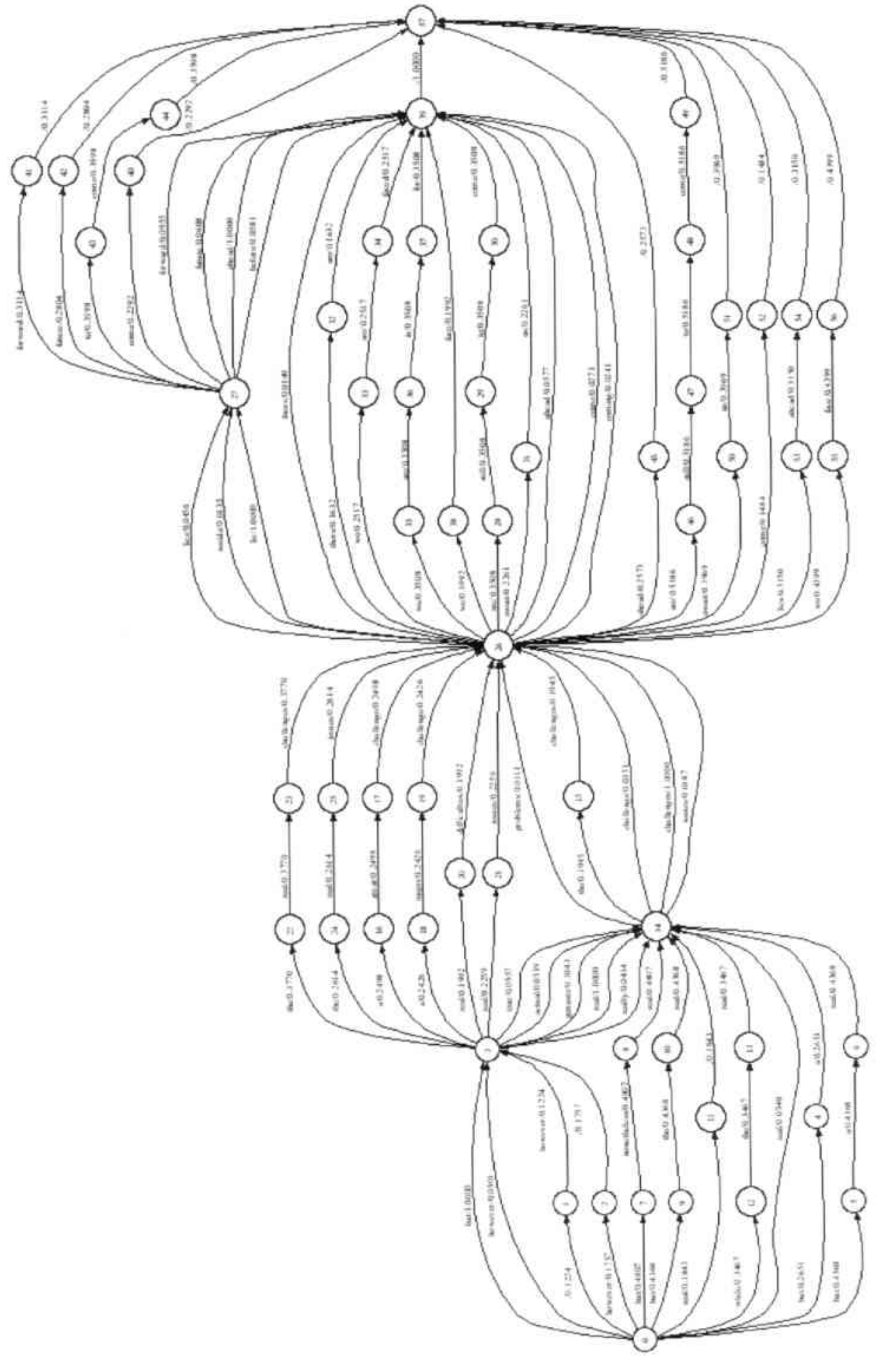
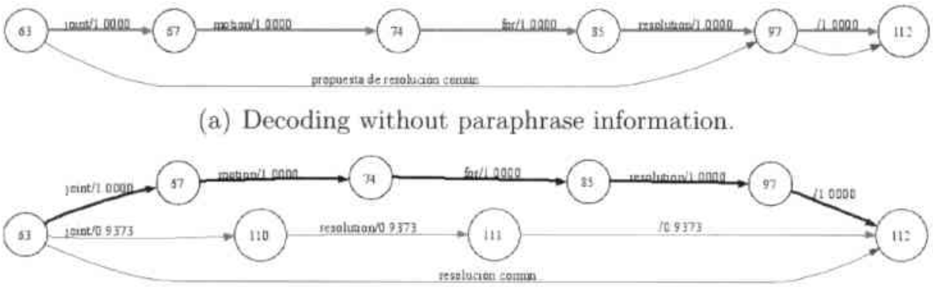


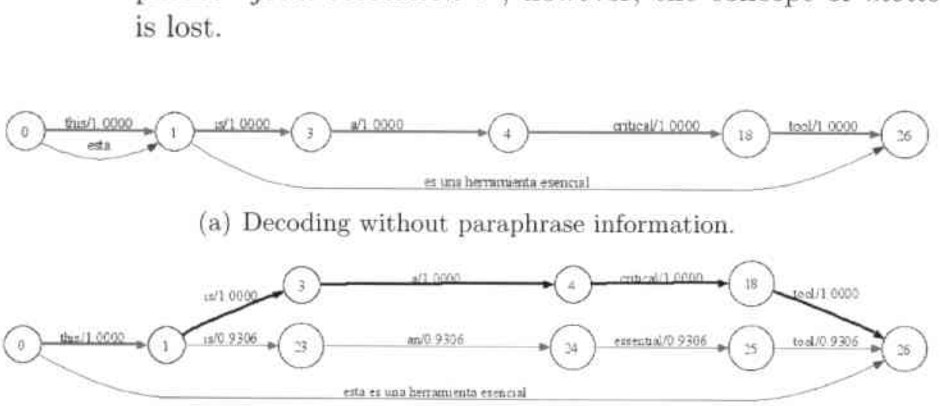
Figure 5.8: Paraphrased input lattice for the sentence *"but real challenges lie ahead ."*, beam size 0.25. Unlike in Figure 5.7, decoding e.g. the sentence *"while the real problems are still to come ."* is now possible. However, the lattice has become too large for efficient decoding.



(a) Decoding without paraphrase information.

(b) Decoding with paraphrase information.

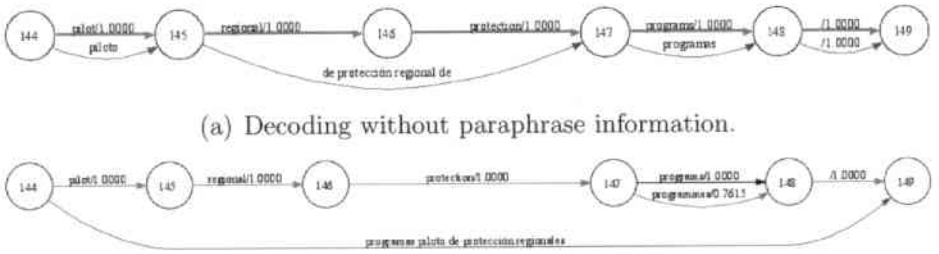
Figure 5.9: The phrase *"joint motion for resolution ."* is decoded in two parts into *"propuesta de resolución común ."* without paraphrase information in 5.9(a). In 5.9(b), the whole phrase is translated as one into *"resolución común ."* using the paraphrase *"joint resolution ."*; however, the concept of *motion* is lost.



(a) Decoding without paraphrase information.

(b) Decoding with paraphrase information.

Figure 5.10: The phrase *"this is a critical tool"* is decoded in two parts into *"esta es una herramienta esencial"* without paraphrase information in 5.10(a). In 5.10(b), only one whole phrase leads to the same translation when substituting *"is a critical tool"* by its paraphrase *"is an essential tool"*, making this an example in which the path length is reduced without any effect on the translation output.



(a) Decoding without paraphrase information.

(b) Decoding with paraphrase information.

Figure 5.11: The phrase *"[to develop] pilot regional protection programs ."* is decoded in four parts into the incorrect Spanish translation *"piloto de protección regional de programas ."* without paraphrase information in 5.11(a). In 5.11(b), inserting the paraphrase *"programmes"* for *"programs"* corrects this error into *"programas piloto de protección regionales ."* and reduces the path length from four to one. While the same effect could have been achieved by normalizing the training data or test sets into either British or American English, this is an example where the incorporation of paraphrases drastically reduces the path length and results in better translation quality as well.

5.5 Training with Additional Paraphrased References

Unlike in [MARD07], where four human reference translations were available and random references were picked as the source, we evaluated our approach with only two human references at hand. Therefore, we set up a baseline training our system with only one (the first given) human reference (H_1)[11]. We supplied additional references found by our two paraphrasing approaches, one stressing quality (Q) and one favoring variety (V). For both approaches, we used the scaling factors derived from paraphrasing worse towards better sentences on the devel-

[11] We did not choose a random reference since we can assume that this approach more accurately simulates the scenario that only one human translator who produces references of the same style is used instead of two, dividing the costs to create reference translations in half.

opment reference set to paraphrase the first and second given human references.[12]. Consequently, we ran eight different experiments with the reference sets H_1, H_1Q_1, H_1V_1, $H_1Q_1V_1$, H_1H_2, $H_1H_2Q_1Q_2$, $H_1H_2V_1V_2$ and $H_1H_2Q_1Q_2V_1V_2$[13].

The scores achieved on the evaluation set after training the system with the different reference sets are shown in Table 5.11. Surprisingly, the BLEU score achieved using only one human reference for training is higher than the one achieved using both human references - the second human development reference might differ in style and/or quality from the first one and the two human references used for evaluation; also, length penalty issues might play a role. We can see that while the brute force paraphrasing method proves to be too drastic in its approach, resulting in significantly lower scores whenever brute force paraphrases were added to the reference set, the addition of n references generated with our first method to n human references did not change BLEU scores significantly and seems to be a too weak approach to paraphrasing. However, having the results from the previous section in mind, we plan to test this approach again for setups with more sophisticated paraphrasing techniques as well as other language pairs and paraphrase sources, for which paraphrasing supplies more additional information than under the examined scenario.

[12] Actually, one would have to use a third independent reference set as a development set just to optimize the scaling factors for parameter generation independently.

[13] H_1 and H_2 stand for the first and second human references; Q_n and V_n are the first-best paraphrase hypotheses for H_n that differ from H_n, generated with the quality (Q) or variety (V) approach.

Dev Set	**BLEU Score**
H_1	0.5416
H_1Q_1	0.5415
H_1V_1	0.5393
$H_1Q_1V_1$	0.5336
H_1H_2	0.5402
$H_1H_2Q_1Q_2$	0.5390
$H_1H_2V_1V_2$	0.5281
$H_1H_2Q_1Q_1V_1V_2$	0.5327

Table 5.11: BLEU scores on the evaluation test set after training the system with the different reference sets.

6 Conclusion and Future Work

We showed how paraphrasing can be incorporated into the CMU SMT system at various stages of the translation process. While in the current case, paraphrasing did not increase BLEU scores in sections 5.4 and 5.5 by a significant margin and although the system learned to exercise paraphrasing only to a low extent, we obtained initial encouraging results regarding the issue of matching longer phrases by taking paraphrases into account in 5.4, an idea that we plan to examine further under other, more promising scenarios.

Performance could be improved for all our approaches by considering paraphrases derived from external sources instead of using paraphrases that were found in the same data that the SMT system was trained with. We are especially interested in trying out paraphrases derived from monolingual data, as they promise to add more new information to the system than parallel bilingual data that the system is trained with anyway. As shown in [CBKO06], paraphrasing source input becomes less helpful the larger the training corpus is, because the system has probably learned enough useful translations from the vast amount of training data already, and the numbers of unknown words and previously unmatched long phrases that could be remedied by paraphrasing them naturally becomes smaller. Related to this aspect is the question whether paraphrases will be more helpful when used for the translation of other language pairs for which current state-of-the-art systems do not work as well yet as for English and Spanish. It would also be interesting to compare the paraphrases derived in Section 5.2 using Spanish as the pivot language with those generated through other pivot languages. Some errors might be avoided using multiple languages of different structure as pivot languages, others by integrating syntactic or semantic information both about the phrase and its paraphrase, as mentioned before. With these different setups, we hope to develop the idea of decreasing path lengths further, because longer phrases can be expected to match more frequently.

Our experiments in Section 5.4 were harmed by the fact that some lattices simply became too large to contain all possible paraphrases, because the resulting translation hypotheses were too numerous. While our pruning strategies employed thus far seem to be promising, more research on lattice pruning before the translation task would be helpful, e.g, a lattice should already in advance only contain those paraphrases that would lead to longer phrase matches. What's more, lattices can be compacted by merging partially identical paths as mentioned. In addition, our definition of what constitutes a paraphrase is relatively loose, and the integration of part-of-speech and contextual information might help to filter out paraphrases beforehand that would form ungrammatical or semantically incorrect sentences. A language model on the source side, potentially added as a feature function to the decoder, could punish translation hypotheses that are formed along a lattice path representing a source sentence that is not well-formed.

The generation of paraphrased references in Section 5.5 might be further refined by not only choosing the best references out of an n-best list that differ from the existing human translations at all, but integrating similarity measures such as word-edit distance compared to the other chosen references into the selection process. A lattice structure for references instead of a high number of partly overlapping sentences is desirable in order to consider all possible paraphrased sentences, as presented in [PKM03]. Also, we could consider a self-training approach similar to the one presented in [Uef06] to perform training that concentrates on those paraphrased references that we consider "good", because our generated paraphrases vary in quality.

With all these further refinements, we expect paraphrasing features to become valuable sources of additional information which could previously not be taken into consideration in the decoding process, allowing for further improvements regarding SMT quality in the foreseeable future.

Bibliography

[BCB05] Colin Bannard and Chris Callison-Burch. Paraphrasing with bilingual parallel corpora. In *Proceedings of ACL*, pages 597–604, 2005.

[BD05] Chris Brockett and William B. Dolan. Support vector machines for paraphrase identification and corpus construction. In *Proceedings of IWP*, 2005.

[BL03] Regina Barzilay and Lillian Lee. Learning to paraphrase: An unsupervised approach using multiple-sequence alignment. In *Proceedings of HLT/NAACL*, pages 16–23, 2003.

[BL05] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of ACL*, pages 65–72. Association for Computational Linguistics, 2005.

[BM01] Regina Barzilay and Kathleen R. McKeown. Extracting paraphrases from a parallel corpus. In *Proceedings of ACL*, pages 50–57, 2001.

[BPPM93] Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, 1993.

[CBKO06] Chris Callison-Burch, Philipp Koehn, and Miles Osborne. Improved statistical machine translation using paraphrases. In *Proceedings of HLT/NAACL*, pages 17–24, 2006.

[CBOK06] Chris Callison-Burch, Miles Osborne, and Philipp Koehn. Re-evaluating the role of BLEU in machine translation research. In *Proceedings of EACL*, 2006.

[DB05] William B. Dolan and Chris Brockett. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of IWP*, 2005.

[KB06] David Kauchak and Regina Barzilay. Paraphrasing for automatic evaluation. In *Proceedings of HLT/NAACL*, pages 455–462, 2006.

[Koe04] Philipp Koehn. Pharaoh: A beam search decoder for phrase-based statistical machine translation models. In *Proceedings of AMTA*, pages 115–124, 2004.

[MARD07] Nitin Madnani, Necip Fazil Ayan, Philip Resnik, and Bonnie Dorr. Using paraphrases for parameter tuning in statistical machine translation. In *Submitted to the Conference on Empirical Methods in Natural Language Processing Conference on Computational Natural Language Learning (EMNLPCoNLL '07)*, 2007.

[Och03] Franz Josef Och. Minimum error rate training in statistical machine translation. In *Proceedings of ACL*, pages 160–167, 2003.

[OGGW06] Karolina Owczarzak, Declan Groves, Josef Van Genabith, and Andy Way. Contextual bitext-derived paraphrases in automatic MT evaluation. In *Proceedings on the Workshop on Statistical Machine Translation*, pages 86–93, 2006.

[ON02] Franz Josef Och and Hermann Ney. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of ACL*, pages 295–302, 2002.

[PKM03] Bo Pang, Kevin Knight, and Daniel Marcu. Syntax-based alignment of multiple translations: Extracting paraphrases and generating new sentences. In *Proceedings of HLT/NAACL*, 2003.

[PRWZ02] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of ACL*, pages 311–318, 2002.

[RLLR05] Grazia Russo-Lassner, Jimmy Lin, and Philip Resnik. A paraphrase-based approach to machine translation evaluation. Technical Report LAMP-TR-125,CS-TR-4754,UMIACS-TR-2005-57, University of Maryland, College Park, 2005.

[Sek05] Satoshi Sekine. Automatic paraphrase discovery based on context and keywords between NE pairs. In *Proceedings of IWP*, 2005.

[Sto02] Andreas Stolcke. SRILM – an extensible language modeling toolkit. In *Proceedings of ICSLP*, pages II: 901–904, 2002.

[Uef06] Nicola Ueffing. Using monolingual source-language data to improve MT performance. In *Proceedings of IWSLT*, 2006.

[VZH+03] Stephan Vogel, Ying Zhang, Fei Huang, Alicia Tribble, Ashish Venogupal, Bing Zhao, and Alex Waibel. The CMU statistical machine translation system. In *Proceedings of MT Summit*, 2003.

[WSS02] Taro Watanabe, Mitsuo Shimohata, and Eiichiro Sumita. Statistical machine translation on paraphrased corpora. In *Proceedings of LREC*, pages 1954–1957, 2002.

[Yam02] Kazuhide Yamamoto. Machine translation by interaction between paraphraser and transfer. In *Proceedings of COLING*, pages 1–7, 2002.

[ZLH06] Liang Zhou, Chin-Yew Lin, and Eduard Hovy. Re-evaluating machine translation results with paraphrase support. In *Proceedings of EMNLP*, 2006.