Triplet Lexicon Models for Statistical Machine Translation

Saša Hasan, Juri Ganitkevitch, Hermann Ney, Jesús Andrés-Ferrer†*
Human Language Technology and Pattern Recognition, RWTH Aachen University, Germany
†Universidad Politécnica de Valencia, Dept. Sist. Informáticos y Computación
{hasan,ganitkevitch,ney}@cs.rwth-aachen.de jandres@dsic.upv.es

Abstract

This paper describes a lexical trigger model for statistical machine translation. We present various methods using triplets incorporating long-distance dependencies that can go beyond the local context of phrases or n-gram based language models. We evaluate the presented methods on two translation tasks in a reranking framework and compare it to the related IBM model 1. We show slightly improved translation quality in terms of BLEU and TER and address various constraints to speed up the training based on Expectation-Maximization and to lower the overall number of triplets without loss in translation performance.

1 Introduction

Data-driven methods have been applied very successfully within the machine translation domain since the early 90s. Starting from single-word-based translation approaches, significant improvements have been made through advances in modeling, availability of larger corpora and more powerful computers. Thus, substantial progress made in the past enables today's MT systems to achieve acceptable results in terms of translation quality for specific language pairs such as Arabic-English. If sufficient amounts of parallel data are available, statistical MT systems can be trained on millions of sentence pairs and use an extended level of context based on bilingual groups of words which denote the building blocks of state-of-the-art phrase-based SMT systems.

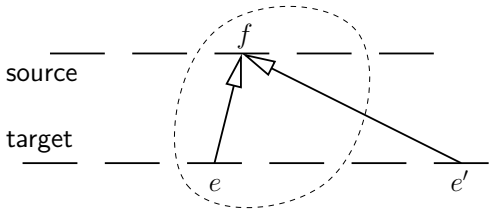

Figure 1: Triplet example: a source word f is triggered by two target words e and e', where one of the words is within and the other outside the considered phrase pair (indicated by the dashed line).

Due to data sparseness, statistical models are often trained on local context only. Language models are derived from n-grams with $n \leq 5$ and bilingual phrase pairs are extracted with lengths up to 10 words on the target side. This captures the local dependencies of the data in detail and is responsible for the success of data-driven phrase-based approaches.

In this work, we will introduce a new statistical model based on lexicalized triplets (f, e, e') which we will also refer to as cross-lingual triggers of the form $(e, e' \rightarrow f)$. This can be understood as two words in one language triggering one word in another language. These triplets, modeled by $p(f|e, e')$, are closely related to lexical translation probabilities based on the IBM model 1, i.e. $p(f|e)$. Several constraints and setups will be described later on in more detail, but as an introduction one can

*The work was carried out while the author was at the Human Language Technology and Pattern Recognition group at RWTH Aachen University and partly supported by the Valencian *Conselleria d'Empresa, Universitat i Ciència* under grants CTBPRA/2005/ and BEFPI/2007/014.

Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pages 372–381,
Honolulu, October 2008.

think of the following interpretation which is depicted in Figure 1: Using a phrase-based MT approach, a source word f is triggered by its translation e which is part of the phrase being considered, whereas another target word e' outside this phrase serves as an additional trigger in order to allow for more fine-grained distinction of a specific word sense. Thus, this cross-lingual trigger model can be seen as a combination of a lexicon model (i.e. f and e) and a model similar to monolingual long-range (i.e. distant bigram) trigger models (i.e. e and e', although these dependencies are reflected indirectly via $e' \rightarrow f$) which uses both local (in-phrase) and global (in-sentence) information for the scoring. The motivation behind this approach is to get non-local information outside the current context (i.e. the currently considered bilingual phrase pair) into the translation process. The triplets are trained via the EM algorithm, as will be shown later in more detail.

2 Related Work

In the past, a significant number of methods has been presented that try to capture long-distance dependencies, i.e. use dependencies in the data that reach beyond the local context of n-grams or phrase pairs. In language modeling, monolingual trigger approaches have been presented (Rosenfeld, 1996; Tillmann and Ney, 1997) as well as syntactical methods that parse the input and model long-range dependencies on the syntactic level by conditioning on the predecessing words and their corresponding parent nodes (Chelba and Jelinek, 2000; Roark, 2001). The latter approach was shown to reduce perplexities and improve the WER in speech recognition systems. One drawback is that the parsing process might slow down the system significantly and the approach is complicated to be integrated directly in the search process. Thus, the effect is often shown offline in reranking experiments using n-best lists.

One of the simplest models that can be seen in the context of lexical triggers is the IBM model 1 (Brown et al., 1993) which captures lexical dependencies between source and target words. It can be seen as a lexicon containing correspondents of translations of source and target words in a very broad sense since the pairs are trained on the full sentence level. The model presented in this work is very close to the initial IBM model 1 and can be seen as taking another word into the conditioning part, i.e. the triggering items.[1] Furthermore, since the second trigger can come from any part of the sentence, we also have a link to long-range monolingual triggers as presented above.

A long-range trigram model is presented in (Della Pietra et al., 1994) where it is shown how to derive a probabilistic link grammar in order to capture long-range dependencies in English using the EM algorithm. Expectation-Maximization is used in the presented triplet model as well which is described in more detail in Section 3. Instead of deriving a grammar automatically (based on POS tags of the words), we rely on a fully lexicalized approach, i.e. the training is taking place at the word level.

Related work in the context of fine-tuning language models by using cross-lingual lexical triggers is presented in (Kim and Khudanpur, 2003). The authors show how to use cross-lingual triggers on a document level in order to extract translation lexicons and domain-specific language models using a mutual information criterion.

Recently, word-sense disambiguation (WSD) methods have been shown to improve translation quality (Chan et al., 2007; Carpuat and Wu, 2007). Chan et al. (2007) use an SVM based classifier for disambiguating word senses which are directly incorporated in the decoder through additional features that are part of the log-linear combination of models. They use local collocations based on surrounding words left and right of an ambiguous word including the corresponding parts-of-speech. Although no long-range dependencies are modeled, the approach yields an improvement of +0.6% BLEU on the NIST Chinese-English task. In Carpuat and Wu (2007), another state-of-the-art WSD engine (a combination of naive Bayes, maximum entropy, boosting and Kernel PCA models) is used to dynamically determine the score of a phrase pair under consideration and, thus, let the phrase selection adapt to the context of the sentence. Although the baseline is significantly lower than in the work of Chan et al., this setup reaches an improvement of 0.5% BLEU on the NIST CE task and up to 1.1% BLEU on the

[1] Thus, instead of $p(f|e)$ we model $p(f|e, e')$ with different additional constraints as explained later on.

IWSLT'06 test sets.

The work in this paper tries to complement the WSD approaches by using long-range dependencies. If triggers from a local context determine different lexical choice for the word being triggered, the setting is comparable to the mentioned WSD approaches (although local dependencies might already be reflected sufficiently in the phrase models). A distant second trigger, however, might have a beneficial effect for specific languages, e.g. by capturing word splits (as it is the case in German for verbs with separable prefixes) or, as already mentioned, allowing for a more fine-grained lexical choice of the word being triggered, namely based on another word which is not part of the current local, i.e. phrasal, context.

The basic idea of triplets of the form $(e, f' \rightarrow f)$, called multi-word extensions, is also mentioned in (Tillmann, 2001) but neither evaluated nor investigated in further detail.

In the following sections, we will describe the model proposed in this work. In Section 3, a detailed introduction is given, as well as the EM training and variations of the model. The different settings will be evaluated in Section 4, where we show experiments on the IWSLT Chinese-English and TC-STAR EPPS English-Spanish/Spanish-English tracks. A discussion of the results and further examples are given in Section 5. Final remarks and future work are addressed in Section 6.

3 Model

As an extension to commonly used lexical word pair probabilities $p(f|e)$ as introduced in (Brown et al., 1993), we define our model to operate on word triplets. A triplet (f, e, e') is assigned a value $\alpha(f|e, e') \geq 0$ with the constraint such that

$$\forall e, e' : \sum_{f} \alpha(f|e, e') = 1.$$

Throughout this paper, e and e' will be referred to as the *first* and the *second trigger*, respectively. In view of its triggers f will be termed the *effect*.

For a given bilingual sentence pair (f_1^J, e_1^I), the probability of a source word f_j given the whole target sentence e_1^I for the triplet model is defined as:

$$p_{all}(f_j|e_1^I) = \frac{1}{Z} \sum_{i=1}^{I} \sum_{k=i+1}^{I} \alpha(f_j|e_i, e_k), \qquad (1)$$

where Z denotes a normalization factor based on the corresponding target sentence length, i.e.

$$Z = \frac{I(I-1)}{2}. \qquad (2)$$

The introduction of a second trigger (i.e. e_k in Eq. 1) enables the model to combine local (i.e. word or phrase level) and global (i.e. sentence level) information.

In the following, we will describe the training procedure of the model via maximum likelihood estimation for the unconstrained case.

3.1 Training

The goal of the training procedure is to maximize the log-likelihood F_{all} of the triplet model for a given bilingual training corpus $\{(f_1^J, e_1^I)\}_1^N$ consisting of N sentence pairs:

$$F_{all} := \sum_{n=1}^{N} \sum_{j=1}^{J_n} \log p_{all}(f_j|e_1^{I_n}),$$

where J_n and I_n are the lengths of the n^{th} source and target sentences, respectively. As there is no closed form solution for the maximum likelihood estimate, we resort to iterative training via the EM algorithm (Dempster et al., 1977). We define the auxiliary function $Q(\mu; \bar{\mu})$ based on F_{all} where $\bar{\mu}$ is the new estimate within an iteration which is to be derived from the current estimate μ. Here, μ stands for the entire set of model parameters to be estimated, i.e. the set of all $\{\alpha(f|e, e')\}$. Thus, we obtain

$$\begin{aligned} Q\big(\{\alpha(f|e,e')\}; \{\bar{\alpha}(f|e,e')\}\big) = \\ \sum_{n=1}^{N} \sum_{j=1}^{J_n} \sum_{i=1}^{I_n} \sum_{k=i+1}^{I_n} \Bigg[\frac{Z_n^{-1} \alpha(f_j|e_i,e_k)}{p_{all}(f_j|e_1^{I_n})} \cdot \\ \log\big(Z_n^{-1} \bar{\alpha}(f_j|e_i,e_k)\big) \Bigg], \end{aligned} \qquad (3)$$

where Z_n is defined as in Eq. 2. Using the method of Lagrangian multipliers for the normalization constraint, we take the derivative with respect to

$\bar{\alpha}(f|e,e')$ and obtain:

$$\bar{\alpha}(f|e,e') = \frac{A(f,e,e')}{\sum_{f'} A(f',e,e')} \quad (4)$$

where $A(f,e,e')$ is a relative weight accumulator over the parallel corpus:

$$A(f,e,e') = \sum_{n=1}^{N}\sum_{j=1}^{J_n} \delta(f,f_j)\frac{Z_n^{-1}\alpha(f|e,e')}{p_{all}(f_j|e_1^{I_n})}C_n(e,e') \quad (5)$$

and

$$C_n(e,e') = \sum_{i=1}^{I_n}\sum_{k=i+1}^{I_n} \delta(e,e_i)\delta(e',e_k).$$

The function $\delta(\cdot,\cdot)$ denotes the Kronecker delta. The resulting training procedure is analogous to the one presented in (Brown et al., 1993) and (Tillmann and Ney, 1997).

The next section presents variants of the basic unconstrained model by putting restrictions on the valid regions of triggers (in-phrase vs. out-of-phrase) and using alignments obtained from either GIZA++ training or forced alignments in order to reduce the model size and to incorporate knowledge already obtained in previous training steps.

3.2 Model variations

Based on the unconstrained triplet model presented in Section 3, we introduce additional constraints, namely the *phrase-bounded* and the *path-aligned* triplet model in the following. The former reduces the number of possible triplets by posing constraints on the position of where valid triggers may originate from. In order to obtain phrase boundaries on the training data, we use forced alignments, i.e. translate the whole training data by constraining the translation hypotheses to the target sentences of the training corpus.

Path-aligned triplets use an alignment constraint from the word alignments that are trained with GIZA++. Here, we restrict the first trigger pair (f,e) to the alignment path as based on the alignment matrix produced by IBM model 4.

These variants require information in addition to the bilingual sentence pair (f_1^J, e_1^I), namely a corresponding phrase segmentation $\Pi = \{\pi_{ij}\}$ with

$$\pi_{ij} = \begin{cases} 1 & \exists \text{ a phrase pair that covers } e_i \text{ and } f_j \\ 0 & \text{otherwise} \end{cases}$$

for the phrase-bounded method and, similarly, a word alignment $A = \{a_{ij}\}$ where

$$a_{ij} = \begin{cases} 1 & \text{if } e_i \text{ is aligned to } f_j \\ 0 & \text{otherwise} \end{cases} .$$

3.2.1 Phrase-bounded triplets

The phrase-bounded triplet model (referred to as p_{phr} in the following), restricts the first trigger e to the same phrase as f, whereas the second trigger e' is set outside the phrase, resulting in

$$p_{phr}(f_j|e_1^I,\Pi) = \frac{1}{Z_j}\sum_{i=1}^{I}\sum_{k=1}^{I}\pi_{ij}(1-\pi_{kj})\alpha(f_j|e_i,e_k). \quad (6)$$

3.2.2 Path-aligned triplet

The path-aligned triplet model (denoted by p_{align} in the following), restricts the scope of e to words aligned to f by A, yielding:

$$p_{align}(f_j|e_1^I,A) = \frac{1}{Z_j}\sum_{i=1}^{I}\sum_{k=1}^{I} a_{ij}\alpha(f_j|e_i,e_k) \quad (7)$$

where the Z_j are, again, the appropriate normalization terms.

Also, to account for non-aligned words (analogously to the IBM models), the empty word e_0 is considered in all three model variations. We show the effect of the empty word in the experiments (Section 4). Furthermore, we can train the presented models in the inverse direction, i.e. $p(e|f,f')$, and combine the two directions in the rescoring framework. The next section presents a set of experiments that evaluate the performance of the presented triplet model and its variations.

4 Experiments

In this section, we describe the system setup used in this work, including the translation tasks and the corresponding training corpora. The experiments are based on an n-best list reranking framework.

4.1 System

The experiments were carried out using a state-of-the-art phrase-based SMT system. The dynamic programming beam search decoder uses several models during decoding by combining them log-linearly. We incorporate phrase translation and word lexicon models in both directions, a language model, as well as phrase and word penalties including a distortion model for the reordering. While generating the hypotheses, a word graph is created which compactly represents the most likely translation hypotheses. Out of this word graph, we generate n-best lists and use them to test the different setups as described in Section 3.

In the experiments, we use 10,000-best lists containing unique translation hypotheses, i.e. duplicates generated due to different phrase segmentations are reduced to one single entry. The advantage of this reranking approach is that we can directly test the obtained models since we already have fully generated translations. Thus, we can apply the triplet lexicon model based on $p(f|e, e')$ and its inverse counterpart $p(e|f, f')$ directly. During decoding, since e' could be from anywhere outside the current phrase, i.e. even from a part which lies beyond the current context which has not yet been generated, we would have to apply additional constraints during training (i.e. make further restrictions such as $i' < i$ for a trigger pair $(e_i, e_{i'})$).

Optimization of the model scaling factors is carried out using minimum error rate training (MERT) on the development sets. The optimization criterion is 100-BLEU since we want to maximize the BLEU score.

4.2 Tasks

4.2.1 IWSLT

For the first part of the experiments, we use the corpora that were released for the IWSLT'07 evaluation campaign. The training corpus consists of approximately 43K Chinese-English sentence pairs, mainly coming from the BTEC corpus (Basic Travel Expression Corpus). This is a multilingual speech corpus which contains tourism-related material, such as transcribed conversations about making reservations, asking for directions or conversations as taking place in restaurants. For the experiments, we use the clean data track, i.e. transcriptions of read speech. As the development set which is used for tuning the parameters of the baseline system and the reranking framework, we use the IWSLT'04 evaluation set (500 sentence pairs). The two blind test sets which are used to evaluate the final performance of the models are the official evaluation sets from IWSLT'05 (506 sentences) and IWSLT'07 (489 sentences).

The average sentence length of the training corpus is 10 words. Thus, the task is somewhat limited and very domain-specific. One of the advantages of this setting is that preliminary experiments can be carried out quickly in order to analyze the effects of the different models in detail. This and the small vocabulary size (12K entries) makes the corpus ideal for first "rapid application development"-style setups without having to care about possible constraints due to memory requirements or CPU time restrictions.

4.2.2 EPPS

Furthermore, additional experiments are based on the EPPS corpus (European Parliament Plenary Sessions) as used within the FTE (Final Text Edition) track of the TC-STAR evaluations. The corpus contains speeches held by politicians at plenary sessions of the European Parliament that have been transcribed, "corrected" to make up valid written texts and translated into several target languages. The language pairs considered in the experiments here are Spanish-English and English-Spanish.

The training corpus consists of roughly 1.3M sentence pairs with 35.5M running words on the English side. The vocabulary sizes are considerably larger than for the IWSLT task, namely around 170K on the target side. As development set, we use the development data issued for the 2006 evaluation (1122 sentences), whereas the two blind test sets are the official evaluation data from 2006 (TC-Star'06, 1117 sentences) and 2007 (TC-Star'07, 1130 sentences).

4.3 Results

4.3.1 IWSLT experiments

One of the first questions that arises is how many EM iterations should be carried out during training of the triplet model. Since the IWSLT task is small,

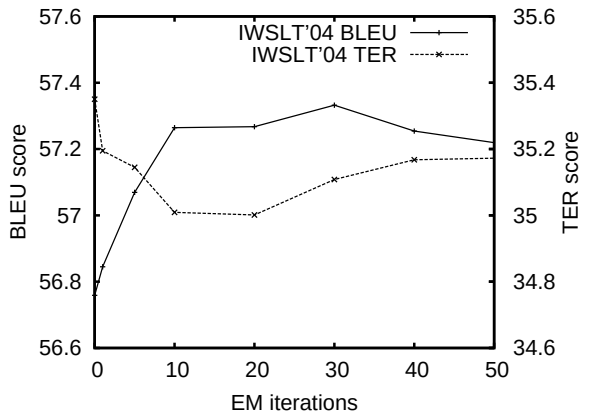

Figure 2: Effect of EM iterations on IWSLT'04, left axis shows BLEU (higher numbers better), right axis (dashed graph) shows TER score (lower numbers better).

	IWSLT'04		IWSLT'05	
	BLEU	TER	BLEU	TER
baseline	56.7	35.49	61.1	30.59
$p_{all}(e\|f, f')$	57.1	35.03	61.3	30.55
w/ singletons	57.3	35.04	61.3	30.61
w/ empties	57.3	35.00	61.2	30.65
+ $p_{all}(f\|e, e')$	**57.5**	**34.69**	**61.7**	**30.24**

Table 1: Different setups showing the effect of singletons and empty words for IWSLT CE IWSLT'04 (dev) and IWSLT'05 (test) sets, p_{all} triplets, 20 EM iterations.

we can quickly run the experiments on a full unconstrained triplet model without any cutoff or further constraints. Figure 2 shows the rescoring performance for different numbers of EM iterations. The first 10 iterations significantly improve the triplet model performance for the IWSLT task. After that, there are no big changes. The performance even degrades a little bit after 30 iterations. For the IWSLT task, we therefore set a fixed number of 20 EM iterations for the following experiments since it shows a good performance in terms of both BLEU and TER score. The oracle TER scores of the 10k-best lists are 14.18% for IWSLT'04, 11.36% for IWSLT'05 and 18.85% for IWSLT'07, respectively.

The next chain of experiments on the IWSLT task investigates the impact of changes to the setup of training an unconstrained triplet model, such as the addition of the empty word and the inclusion of singletons (i.e. triplets that were only seen once in the training data). This might show the importance of rare events in order to derive strategies when moving to larger tasks where it is not feasible to train all possible triplets, such as e.g. on the EPPS task (as shown later) or the Chinese-English NIST task. The results for the unconstrained model are shown in Table 1, beginning with a full triplet model in reverse direction, $p_{all}(e|f, f')$, that contains no singletons and no empty words for the triggering side. In this setting, singletons seem to help on dev but there is no clear improvement on one of the test sets, whereas empty words do not make a significant difference but can be used since they do not harm either. The baseline can be improved by +0.6% BLEU and around -0.5% in TER on the IWSLT'04 set. For the various setups, there are no big differences in the TER score which might be an effect of optimization on BLEU. Therefore, for further experiments using the constraints from Section 3.2, we use both singletons and empty words as the default.

	IWSLT'05		IWSLT'07	
	BLEU	TER	BLEU	TER
baseline	61.1	30.59	38.9	45.60
IBM model 1	61.5	30.29	39.4	45.31
trip fe+ef p_{all}	**61.7**	**30.24**	**39.7**	45.24
trip fe+ef p_{phr}	61.5	30.32	39.1	45.36
trip fe+ef p_{align}	61.2	30.60	**39.7**	**45.02**

Table 2: Comparison of triplet variants on IWSLT CE test sets, 20 EM iterations, with singletons and empty words.

Adding the other direction $p(f|e, e')$ results in another increase, with a total of +0.8% BLEU and -0.8% TER, which shows that the combination of both directions helps overall translation quality. The results on the two test sets are shown in Table 2. As can be seen, we arrive at similar improvements, namely +0.6% BLEU and -0.3% TER on IWSLT'05 and +0.8% BLEU and -0.4% TER on IWSLT'07, respectively. The constrained models, i.e. the phrase-bounded (p_{phr}) and path-aligned (p_{align}) triplets are outperformed by the full unconstrained case, although on IWSLT'07 both unconstrained and path-aligned models are close.

For a fair comparison, we added a classical IBM model 1 in the rescoring framework. It can be seen that the presented triplet models slightly outperform

	TC-Star'06		TC-Star'07	
	BLEU	TER	BLEU	TER
baseline	52.3	34.57	50.4	36.46
trip fe+ef p_{all}	52.9	34.32	50.6	36.34
+ max dist 10	52.9	34.20	50.8	36.22

Table 3: Effect of using maximum distance constraint for p_{all} on EPPS Spanish-English test sets, occ_3, 4 EM iterations due to time constraints.

the simple IBM model 1. Note that IBM model 1 is a special case of the triplet lexicon model if the second trigger is the empty word.

4.3.2 EPPS experiments

Since EPPS is a considerably harder task (larger vocabulary and longer sentences), the training of a full unconstrained triplet model cannot be done due to memory restrictions. One possibility to reduce the number of extracted triplets is to apply a maximum distance constraint in the training procedure, i.e. only trigger pairs are considered where the distance between first and second trigger is below or equal to the specified maximum.

Table 3 shows the effect of a maximum distance constraint for the Spanish-English direction. Due to the large amount of triplets (we extract roughly two billion triplets[2] for the EPPS data), we drop all triplets that occur less than 3 times which results in 640 million triplets. Also, due to time restrictions[3], we only train 4 iterations and compare it to 4 iterations of the same setting with the maximum distance set to 10. The training with the maximum distance constraints ends with a total of 380 million triplets. As can be seen (Table 3), the performance is comparable while cutting down the computation time from 9.2 to 3.1 hours. The experiments were carried out on a 2.2GHz Opteron machine with 16 GB of memory. The overall gain is +0.4–0.6% BLEU and up to -0.4% in TER. We even observe a slight increase in BLEU for the TC-Star'07 set which might be a random effect due to optimization on the development set where the behavior is the same as for TC-Star'06.

[2]Extraction can be easily done in parallel by splitting the corpus and merging identical triplets iteratively in a separate step for two chunks at a time.

[3]One iteration needs more than 12 hours for the unconstrained case.

	TC-Star'06		TC-Star'07	
	BLEU	TER	BLEU	TER
baseline	49.5	37.65	51.0	36.03
trip fe+ef p_{phr}	50.2	37.01	51.5	35.38
+ occ_2	50.2	37.06	51.8	35.32

Table 4: Results on EPPS, English-Spanish, p_{phr} combined, occ_3, 10 EM iterations.

	TC-Star'06		TC-Star'07	
	BLEU	TER	BLEU	TER
baseline	49.5	37.65	51.0	36.03
using FA	50.0	37.18	51.7	35.52
using IBM4	50.0	37.12	51.7	35.43
+ occ_2	50.2	36.84	52.0	35.10
+ max dist 1	50.0	37.10	51.7	35.51

Table 5: Results on EPPS, English-Spanish, maximum approximation, p_{align} combined, occ_3, 10 EM iterations.

Results on EPPS English-Spanish for the phrase-bounded triplet model are presented in Table 4. Since the number of triplets is less than for the unconstrained model, we can lower the cutoff from 3 to 2 (denoted in the table by occ_3 and occ_2, respectively). There is a small additional gain on the TC-Star'07 test set by this step, with a total of +0.7% BLEU for TC-Star'06 and +0.8% BLEU for TC-Star'07.

Table 5 shows results for a variation of the path-aligned triplet model p_{align} that restricts the first trigger to the best aligned word as estimated in the IBM model 1, thus using a maximum-approximation of the given word alignment. The model was trained on two word alignments, firstly the one contained in the forced alignments on the training data, and secondly on an IBM-4 word alignment generated using GIZA++. For this second model we also demonstrate the improvement obtained when increasing the triplet lexicon size by using less trimming.

Another experiment was carried out to investigate the effect of immediate neighboring words used as triggers within the p_{align} setting. This is equivalent to using a "maximum distance of 1" constraint. We obtained worse results, namely a 0.2-0.3% drop in BLEU and a 0.3-0.4% raise in TER (cf. Table 5, last row), although the training is significantly faster with this setup, namely roughly 30 minutes per it-

	TC-Star'06		TC-Star'07	
	BLEU	TER	BLEU	TER
baseline	49.5	37.65	51.0	36.03
IBM model 1	50.0	37.12	51.8	35.51
p_{all}, occ_3	50.0	37.17	51.8	35.43
p_{phr}, occ_2	50.2	37.06	51.8	35.32
p_{align}, occ_2	**50.2**	**36.84**	**52.0**	**35.10**

Table 6: Final results on EPPS English-Spanish, constrained triplet models, 10 EM iterations, compared to standard IBM model 1.

f	e	e'	$\alpha(f\|e,e')$
pagar	taxpayer	bill	0.76
factura	taxpayer	bill	0.11
contribuyente	taxpayer	bill	0.10
f	e	–	$p_{ibm1}(f\|e)$
contribuyente	taxpayer		0.40
contribuyentes	taxpayer		0.18
europeo	taxpayer		0.08
factura	bill		0.19
ley	bill		0.18
proyecto	bill		0.11

Table 7: Example of triplets and related IBM model 1 lexical probabilities. The triggers "taxpayer" and "bill" have a new effect ("pagar"), previously not seen in the top ranks of the lexicon.

eration using less than 2 GB of memory. However, this shows that triggers outside the immediate context help overall translation quality. Additionally, it supports the claim that the presented methods are a complementary alternative to the WSD approaches mentioned in Section 2 which only consider the immediate context of a single word.

Finally, we compare the constrained models to an unconstrained setting and, again, to a standard IBM model 1. Table 6 shows that the p_{align} model constrained on using the IBM-4 word alignments yields +0.7% in BLEU on TC-Star'06 which is +0.2% more than with a standard IBM model 1. TER decreases by -0.3% when compared to model 1. For the TC-Star'07 set, the observations are similar.

The oracle TER scores of the development n-best list are 25.16% for English-Spanish and 27.0% for Spanish-English, respectively.

5 Discussion

From the results of our reranking experiments, we can conclude that the presented triplet lexicon model outperforms the baseline single-best hypotheses of the decoder. When comparing to a standard IBM model 1, the improvements are significantly smaller though measurable. So far, since IBM model 1 is considered one of the stronger rescoring models, these results look promising. An unconstrained triplet model has the best performance if training is feasible since it also needs the most memory and time to be trained, at least for larger tasks.

In order to cut down computational requirements, we can apply phrase-bounded and path-aligned training constraints that restrict the possibilities of selecting triplet candidates (in addition to simple thresholding). Although no clear effect could be observed for adding empty words on the triggering side, it does not harm and, thus, we get a similar functionality to IBM model 1 being "integrated" in the triplet lexicon model. The phrase-bounded training variant uses forced alignments computed on the whole training data (i.e. search constrained to producing the target sentences of the bilingual corpus) but could not outperform the path-aligned model which reuses the alignment path information obtained in regular GIZA++ training.

Additionally, we observe a positive impact from triggers lying outside the immediate context of one predecessor or successor word.

5.1 Examples

Table 7 shows an excerpt of the top entries for $(e, e') = (taxpayer, bill)$ and compares it to the top entries of a lexicon based on IBM model 1. We observe a triggering effect since the Spanish word *pagar* (*to pay*) is triggered at top position by the two English words *taxpayer* and *bill*. The average distance of *taxpayer* and *bill* is 5.4 words. The models presented in this work try to capture this property and apply it in the scoring of hypotheses in order to allow for better lexical choice in specific contexts.

In Table 8, we show an example translation where rescoring with the triplet model achieves higher n-gram coverage on the reference translation than the variant based on IBM model 1 rescoring. The differing phrases are highlighted.

Source sentence	... respecto de la Posición Común del Consejo con vistas a la adopción del Reglamento del Parlamento Europeo y del Consejo relativo al ...
IBM-1 rescoring	... on the Council common position with a view to the adoption of the Rules of Procedure of the European Parliament and of the Council ...
Triplet rescoring	... on the common position of the Council with a view to the adoption of the regulation of the European Parliament and of the Council ...
Reference translation	... as regards the Common Position of the Council with a view to the adoption of a European Parliament and Council Regulation as regards the ...

Table 8: A translation example on TC-Star'07 Spanish-English comparing the effect of the triplet model to a standard IBM-1 model.

6 Outlook

We have presented a new lexicon model based on triplets extracted on a sentence level and trained iteratively using the EM algorithm. The motivation of this approach is to add an additional second trigger to a translation lexicon component which can come from a more global context (on a sentence level) and allow for a more fine-grained lexical choice given a specific context. Thus, the method is related to word sense disambiguation approaches.

We showed improvements by rescoring n-best lists of the IWSLT Chinese-English and EPPS Spanish-English/English-Spanish task. In total, we achieve up to +1% BLEU for some of the test sets in comparison to the decoder baseline and up to +0.3% BLEU compared to IBM model 1.

Future work will address an integration into the decoder since the performance of the current rescoring framework is limited by the quality of the n-best lists. For the inverse model, $p(e|f, f')$, an integration into the search is directly possible. Further experiments will be conducted, especially on large tasks such as the NIST Chinese-English and Arabic-English task. Training on these huge databases will only be possible with an appropriate selection of promising triplets.

Acknowledgments

This material is partly based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-06-C-0023, and was partly realized as part of the Quaero Programme, funded by OSEO, French State agency for innovation.

The authors would like to thank the anonymous reviewers for their valuable comments.

References

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

Marine Carpuat and Dekai Wu. 2007. Improving statistical machine translation using word sense disambiguation. In *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007)*, Prague, Czech Republic, June.

Yee Seng Chan, Hwee Tou Ng, and David Chiang. 2007. Word sense disambiguation improves statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 33–40, Prague, Czech Republic, June. Association for Computational Linguistics.

Ciprian Chelba and Frederick Jelinek. 2000. Structured language modeling. *Computer Speech and Language*, 14(4):283–332.

Stephen A. Della Pietra, Vincent J. Della Pietra, John R. Gillett, John D. Lafferty, Harry Printz, and Luboš Ureš. 1994. Inference and estimation of a long-range trigram model. In J. Oncina and R. C. Carrasco, editors, *Grammatical Inference and Applications, Second International Colloquium, ICGI-94*, volume 862, pages 78–92, Alicante, Spain. Springer Verlag.

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–22.

Woosung Kim and Sanjeev Khudanpur. 2003. Cross-lingual lexical triggers in statistical language modeling. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 17–24, Morristown, NJ, USA. Association for Computational Linguistics.

Brian Roark. 2001. Probabilistic top-down parsing and language modeling. *Computational Linguistics*, 27(2):249–276.

Ronald Rosenfeld. 1996. A maximum entropy approach to adaptive statistical language modeling. *Computer Speech and Language*, 10(3):187–228.

Christoph Tillmann and Hermann Ney. 1997. Word triggers and the EM algorithm. In *Proc. Special Interest Group Workshop on Computational Natural Language Learning (ACL)*, pages 117–124, Madrid, Spain, July.

Christoph Tillmann. 2001. *Word Re-Ordering and Dynamic Programming based Search Algorithm for Statistical Machine Translation*. Ph.D. thesis, RWTH Aachen University, Aachen, Germany, May.