Comparison of Extended Lexicon Models in Search and Rescoring for SMT

Saša Hasan and Hermann Ney
Human Language Technology and Pattern Recognition Group
Chair of Computer Science 6, RWTH Aachen University, Germany
{hasan,ney}@cs.rwth-aachen.de

Abstract

We show how the integration of an extended lexicon model into the decoder can improve translation performance. The model is based on lexical triggers that capture long-distance dependencies on the sentence level. The results are compared to variants of the model that are applied in reranking of n-best lists. We present how a combined application of these models in search and rescoring gives promising results. Experiments are reported on the GALE Chinese-English task with improvements of up to +0.9% BLEU and -1.5% TER absolute on a competitive baseline.

1 Introduction

Phrase-based statistical machine translation has improved significantly over the last decade. The availability of large amounts of parallel data and access to open-source software allow for easy setup of translation systems with acceptable performance. Public evaluations such as the NIST MT Eval or the WMT Shared Task help to measure overall progress within the community. Most of the groups use a phrase-based decoder (e.g. Pharaoh or the more recent Moses) based on a log-linear fusion of models that enable the avid researcher to quickly incorporate additional features and investigate the effect of additional knowledge sources to guide the search for better translation hypotheses.

In this paper, we deal with an extended lexicon model and its incorporation into a state-of-the-art decoder. We compare the results of the integration to a similar setup used within a rescoring framework and show the benefits of integrating additional models directly into the search process. As will be shown, although a rescoring framework is suitable for obtaining quick trends of incorporating additional models into a system, an alternative that includes the model in search should be preferred. The integration does not only yield better performance, we will also show the benefit of combining both approaches in order to boost translation quality even more. The extended lexicon model which we apply is motivated by a trigger-based approach (Hasan et al., 2008). A standard lexicon modeling dependencies of target and source words, i.e. $p(e|f)$, is extended with a second trigger f' on the source side, resulting in $p(e|f, f')$. This model allows for a more fine-grained lexical choice of the target word depending on the additional source word f'. Since the second trigger can move over the whole sentence, we capture global (sentence-level) context that is not modeled in local n-grams of the language model or in bilingual phrase pairs that cover only a limited amount of consecutive words.

Related work A similar approach has been tried in the word-sense disambiguation (WSD) domain where local but also across-sentence unigram collocations of words are used to refine phrase pair selection dynamically by incorporating scores from the WSD classifier (Chan et al., 2007). A maximum-entropy based approach with different features of surrounding words that are locally bound to a context of three positions to the left and right is reported in (García-Varea et al., 2001). A logistic regression-based word translation model is investigated by Vickrey et al. (2005) but has not been evaluated on a machine translation task. Another WSD approach incorporating context-dependent phrasal translation lexicons is presented by Carpuat and Wu (2007) and has been evaluated on several translation

Proceedings of NAACL HLT 2009: Short Papers, pages 17–20,
Boulder, Colorado, June 2009. ©2009 Association for Computational Linguistics

tasks. The triplet lexicon model presented in this work can also be interpreted as an extension of the standard IBM model 1 (Brown et al., 1993) with an additional trigger.

2 Setup

The main focus of this work investigates an extended lexicon model in search and rescoring. The model that we consider here and its integration in the decoder and setup for rescoring are presented in the following sections.

2.1 Extended lexicon model

The triplets of the extended lexicon model $p(e|f, f')$ are composed of two words in the source language triggering one target word. In order to limit the overall number of triplets, we apply a training constraint that reuses the word alignment information obtained in the GIZA++ step. For source words f, we only consider the ones that are aligned to a target word e given the GIZA++ word alignment. The second trigger f' is allowed to move over the whole source sentence, thus capturing long-distance effects that can be observed in the training data:

$$p(e_1^I | f_1^J, \{a_{ij}\}) = \prod_{i=1}^{I} p(e_i | f_1^J, \{a_{ij}\}) = \prod_{i=1}^{I} \frac{1}{Z_i} \sum_{j \in \{a_i\}} \sum_{j'=1}^{J} p(e_i | f_j, f_{j'}) \qquad (1)$$

where $\{a_{ij}\}$ denotes the alignment matrix of the sentence pair f_1^J and e_1^I and the first sum goes over all f_j that are aligned to the current e_i (expressed as $j \in \{a_i\}$). The factor $Z_i = J \cdot |\{a_i\}|$ normalizes the double summation accordingly. Eq. 1 is used in the iterative EM training on all sentence pairs of the training data. Empty words are allowed on the triggering part and low probability triplets are trimmed.

2.2 Decoding

Regarding the search, we can apply this model directly when scoring bilingual phrase pairs. Given a trained model for $p(e|f, f')$, we compute the feature score h_t of a phrase pair $(\tilde{e}, \tilde{f})$ as

$$h_t(\tilde{e}, \tilde{f}, \{\tilde{a}_{ij}\}, f_1^J) = \qquad (2)$$
$$-\sum_i \log \sum_{j \in \{\tilde{a}_i\}} \sum_{j'} p(\tilde{e}_i | \tilde{f}_j, f_{j'}) + \sum_i \log Z_i$$

where i moves over all target words in the phrase $\tilde{e}$, the sum over j selects the aligned source words $\tilde{f}_j$ given $\{\tilde{a}_{ij}\}$, the alignment matrix within the phrase pair, and j' incorporates the whole source sentence f_1^J. Analogous to Eq. 1, $Z_i = J \cdot |\{\tilde{a}_i\}|$ denotes the number of overall source words times the number of aligned source words to each $\tilde{e}_i$. In Eq. 2, we take negative log-probabilities and normalize to obtain the final score (representing costs) for the given phrase pair. Note that in search, we can only use this direction, $p(e|f, f')$, since the whole source sentence is available for triggering effects whereas not all target words have been generated so far, as it would be necessary for the reverse direction, $p(f|e, e')$. Due to data sparseness, we smooth the model by using a floor value of 10^{-7} for unseen events during decoding. Furthermore, an implicit backoff to IBM1 exists if the second trigger is the empty word, i.e. for events of the form $p(e|f, \varepsilon)$.

2.3 Rescoring

In rescoring, we constrain the scoring of our hypotheses to a limited set of n-best translations that are extracted from the word graph, a pruned compact representation of the search space. The advantage of n-best list rescoring is the full availability of both source text and target translation, thus allowing for the application of additional (possibly more complex) models that are hard to implement directly in search, such as e.g. syntactic models based on parsers or huge LMs that would not fit in memory during decoding. Since we are limiting ourselves to a small extract of translation hypotheses, rescoring models cannot outperform the same models if applied directly in search. One advantage though is that we can apply the introduced trigger model also in the other direction, i.e. using $p(f|e, e')$, where two target words trigger one source word. Generally, the combination of two directions of a model yields further improvements, so we investigated how this additional direction helps in rescoring (cf. Section 3.1).

In our experiments, we use 10 000-best lists extracted from the word graphs. An initial setting uses the baseline system, whereas a comparative setup incorporates the $(e|f, f')$ direction of the trigger lexicon model in search and adds the reversed direction in rescoring. Additionally, we use n-gram posteriors, a sentence length model and two large language

	train (ch/en)	test08 (NW/WT)	
Sent. pairs	9.1M	480	490
Run. words	259M/300M	14.8K	12.3K
Vocabulary	357K/627K	3.6K	3.2K

Table 1: GALE Chinese-English corpus statistics.

models, a 5-gram count LM trained on 2.5G running words and the Google Web 1T 5-grams. The feature weights of the log-linear mix are tuned on a separate development set using the Downhill Simplex algorithm.

3 Experiments

The experiments are carried out with a GALE system using the official development and test sets of the GALE 2008 evaluation. The corpus statistics are shown in Table 1. The triplet lexicon model was trained on a subset of the overall data. We used 1.4M sentence pairs with 32.3M running words on the English side. The vocabulary sizes were 76.5K for the source and 241.7K for the target language. The final lexicon contains roughly 62 million triplets.

The baseline system incorporates the standard model setup used in phrase-based SMT which combines phrase translation and word lexicon models in both directions, a 5-gram language model, word and phrase penalties, and two models for reordering (a standard distortion model and a discriminative phrase orientation model). For a fair comparison, we also added the related IBM model 1 $p(e|f)$ to the baseline since it can be computed on the sentence-level for this direction, target given source. This step achieves +0.5% BLEU on the development set for newswire but has no effect on test. As will be presented in the next section, the extension to another trigger results in improvements over this baseline, indicating that the extended triplet model is superior to the standard IBM model 1. The feature weights were optimized on separate development sets for both newswire and web text.

We perform the following pipeline of experiments: A first run generates word graphs using the baseline models. From this word graph, we extract 10k-best lists and compare the performance to a reranked version including the additional models. In a second step, we add one of the trigger lexicon models to the search process, regenerate word graphs, extract updated n-best lists and add the remaining models again in a reranking step.

Chinese-English	newswire		web text	
GALE test08	BLEU	TER	BLEU	TER
baseline	32.5	59.4	25.8	64.0
rescore, no triplets	32.8	59.0	26.6	63.5
resc. triplets fe+ef	33.2	58.6	27.1	63.0
triplets in search ef	33.1	58.8	26.0	63.5
rescore, no triplets	33.2	58.6	26.7	63.5
rescore, triplets fe	33.7	58.1	27.2	62.0

Table 2: Results obtained for the two test sets. For the triplet models, "fe" means $p(f|e, e')$ and "ef" denotes $p(e|f, f')$. BLEU/TER scores are shown in percent.

3.1 Results

Table 2 presents results that were obtained on the test sets. All results are based on lowercase evaluations since the system is trained on lowercased data in order to keep computational resources feasible. For the newswire setting, the baseline is 32.5% BLEU and 59.4% TER. Rescoring with additional models not including triplets gives only slight improvements. By adding the path-aligned triplet model in both directions, we observe an improvement of +0.7% BLEU and -0.8% TER. Using the triplet model in source to target direction (e, f, f') during the search process, we arrive at a similar BLEU improvement of +0.6% without any reranking models. We add the other direction of the triplets (f, e, e') (the one that can not be used directly in search) and obtain 33.7% BLEU on the newswire set. The overall cumulative improvements of triplets in search and reranking are +0.9% BLEU and -0.9% TER when compared to the rescored baseline not incorporating triplet models and +1.2%/-1.3% on the decoder baseline, respectively.

For the web text setting, the baseline is considerably lower at 25.8% BLEU and 64.0% TER (cf. right part of Table 2). We observe an improvement for the baseline reranking models, a large part of which is due to the Google Web LM. Adding triplets to search does not help significantly (+0.2%/-0.5% BLEU/TER). This might be due to training the triplet lexicon mainly on newswire data. Reranking without triplets performs similar to the baseline experiment. Mixing in the (f, e, e') direction helps again: The final score comes out at 27.2% BLEU and 62.0% TER, the latter being significantly better than the reranked baseline (-1.5% in TER).

3.2 Discussion

The results indicate that it is worth moving models from rescoring to the search process. This is not surprising (and probably well known in the community). Interestingly, the triplet model can improve translation quality in addition to its related IBM model 1 which was already part of the baseline. It seems that the extension by a second trigger helps to capture some language specific properties for Chinese-English which go beyond local lexical (word-to-word) dependencies. In Table 3, we show an example of improved translation quality where a triggering effect can be observed. Due to the topic of the sentence, the phrase *local employment* was chosen over *own jobs*. One of the top triplets in this context is $p(\text{employment} \mid$ 就业 , 人才 $)$, where 就业 is “employment” due to the path-aligned constraint and 人才 means “talent”. Note that the distance between these two triggers is five tokens.

4 Conclusion

We presented the integration of an extended lexicon model into the search process and compared it to a variant which was used in reranking n-best lists. In order to keep the overall number of triplets feasible, and thus memory footprints and training times low, we chose a path-constrained triplet model that restricts the first source trigger to the aligned target word, whereas the second trigger can move along the whole source sentence. The motivation was to allow for a more fine-grained lexical choice of target words by looking at sentence-level context. The overall improvements that can be accounted to the triplets are up to +0.9% BLEU and -1.5% TER.

In the future, we plan to investigate more triplet model variants and work on additional language pairs such as French-English or German-English. The reverse direction, $p(f|e, e')$, is hard to implement outside of a reranking framework where the full target hypotheses are already fully generated. It might be worth looking at cross-lingual trigger models such as $p(f|e, f')$ or constrained variants like $p(f|e, e')$ with $e' < e$, i.e. the second trigger coming from the left context within a sentence which has already been generated.

source	德国 为了 保护 本国 人 就业 , 对 引进 国外 人才 设 了 较 高 的 门槛 .
baseline	germany, in order to protect their own jobs, the introduction of foreign talent, a relatively high threshold.
triplets	in order to protect local employment, germany has a relatively high threshold for the introduction of foreign talent.
reference	in order to protect native employment, germany has set a relatively high threshold for bringing in foreign talents.

Table 3: Translation example on the newswire test set.

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 Juri Ganitkevitch for training the triplet model.

References

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

M. Carpuat and D. Wu. 2007. Improving statistical machine translation using word sense disambiguation. In *Proc. EMNLP-CoNLL*, Prague, Czech Republic, June.

Y. S. Chan, H. T. Ng, and D. Chiang. 2007. Word sense disambiguation improves statistical machine translation. In *Proc. ACL*, pages 33–40, Prague, Czech Republic, June.

I. García-Varea, F. J. Och, H. Ney, and F. Casacuberta. 2001. Refined lexicon models for statistical machine translation using a maximum entropy approach. In *Proc. ACL Data-Driven Machine Translation Workshop*, pages 204–211, Toulouse, France, July.

S. Hasan, J. Ganitkevitch, H. Ney, and J. Andrés-Ferrer. 2008. Triplet lexicon models for statistical machine translation. In *Proc. EMNLP*, pages 372–381, Honolulu, Hawaii, October.

D. Vickrey, L. Biewald, M. Teyssier, and D. Koller. 2005. Word-sense disambiguation for machine translation. In *Proc. HLT-EMNLP*, pages 771–778, Morristown, NJ, USA.