Combining Multi-Engine Translations with Moses

Yu Chen[1], **Michael Jellinghaus**[1], **Andreas Eisele**[1,2],**Yi Zhang**[1,2],
Sabine Hunsicker[1], **Silke Theison**[1], **Christian Federmann**[2], **Hans Uszkoreit**[1,2]
1: Universität des Saarlandes, Saarbrücken, Germany
2: Deutsches Forschungszentrum für Künstliche Intelligenz GmbH, Saarbrücken, Germany
{yuchen,micha,yzhang,sabineh,sith}@coli.uni-saarland.de
{eisele,cfedermann,uszkoreit}@dfki.de

Abstract

We present a simple method for generating translations with the Moses toolkit (Koehn et al., 2007) from existing hypotheses produced by other translation engines. As the structures underlying these translation engines are not known, an evaluation-based strategy is applied to select systems for combination. The experiments show promising improvements in terms of BLEU.

1 Introduction

With the wealth of machine translation systems available nowadays (many of them online and for free), it makes increasing sense to investigate clever ways of combining them. Obviously, the main objective lies in finding out how to integrate the respective advantages of different approaches: Statistical machine translation (SMT) and rule-based machine translation (RBMT) systems often have complementary characteristics. Previous work on building hybrid systems includes, among others, approaches using reranking, regeneration with an SMT decoder (Eisele et al., 2008; Chen et al., 2007), and confusion networks (Matusov et al., 2006; Rosti et al., 2007; He et al., 2008).

The approach by (Eisele et al., 2008) aimed specifically at filling lexical gaps in an SMT system with information from a number of RBMT systems. The output of the RBMT engines was word-aligned with the input, yielding a total of seven phrase tables which where simply concatenated to expand the phrase table constructed from the training corpus. This approach differs from the confusion network approaches mainly in that the final hypotheses do not necessarily follow any of the input translations as the skeleton. On the other hand, it emphasizes that the additional translations should be produced by RBMT systems with lexicons that cannot be learned from the data.

The present work continues on the same track as the paper mentioned above but implements a number of important changes, most prominently a relaxation of the restrictions on the number and type of input systems. These differences are described in more detail in Section 2. Section 3 explains the implementation of our system and Section 4 its application in a number of experiments. Finally, Section 5 concludes this paper with a summary and some thoughts on future work.

2 Integrating Multiple Systems of Unknown Type and Quality

When comparing (Eisele et al., 2008) to the present work, our proposal is more general in a way that the requirement for knowledge about the systems is minimum. The types and the identities of the participated systems are assumed unknown. Accordingly, we are not able to restrict ourselves to a certain class of systems as (Eisele et al., 2008) did. We rely on a standard phrase-based SMT framework to extract the valuable pieces from the system outputs. These extracted segments are also used to improve an existing SMT system that we have access to.

While (Eisele et al., 2008) included translations from all of a fixed number of RBMT systems and added one feature to the translation model for each system, integrating all given system outputs in this way in our case could expand the search space tremendously. Meanwhile, we cannot rely on the assumption that all candidate systems actually have the potential to improve our baseline. This implies the need for a first step of system selection where the best candidate systems are identified and a limited number of them is chosen to be included in the combination. Our approach would not work without a small set of tuning data being available so that we can evaluate the systems for later selection and adjust the weights of our systems. Such tuning data is included in this year's

Proceedings of the Fourth Workshop on Statistical Machine Translation , pages 42–46,
Athens, Greece, 30 March – 31 March 2009.

task.

In this paper, we use the Moses decoder to construct translations from the given system outputs. We mainly propose two slightly different ways: One is to construct translation models solely from the given translations and the other is to extend an existing translation model with these additional translations.

3 Implementation

Despite the fact that the output of current MT systems is usually not comparable in quality to human translations, the machine-generated translations are nevertheless "parallel" to the input so that it is straightforward to construct a translation model from data of this kind. This is the spirit behind our method for combining multiple translations.

3.1 Direct combination

Clearly, for the same source sentence, we expect to have different translations from different translation systems, just like we would expect from human translators. Also, every system may have its own advantages. We break these translations into smaller units and hope to be able to select the best ones and form them into a better translation.

One single translation of a few thousand sentences is normally inadequate for building a reliable general-purpose SMT system (data sparseness problem). However, in the system combination task, this is no longer an issue as the system only needs to translate sentences within the data set.

When more translation engines are available, the size of this set becomes larger. Hence, we collect translations from all available systems and pair them with the corresponding input text, thus forming a medium-sized "hypothesis" corpus. Our system starts processing this corpus with a standard phrase-based SMT setup, using the Moses toolkit (Koehn et al., 2007).

The hypothesis corpus is first tokenized and lowercased. Then, we run GIZA++ (Och and Ney, 2003) on the corpus to obtain word alignments in both directions. The phrases are extracted from the intersection of the alignments with the "grow" heuristics. In addition, we also generate a reordering model with the default configuration as included in the Moses toolkit. This "*hypothesis*" translation model can already be used by the Moses decoder together with a language model to perform translations over the corresponding sentence set.

3.2 Integration into existing SMT system

Sometimes, the goal of system combination is not only to produce a translation but also to improve one of the systems. In this paper, we aim at incorporating the additional system outputs to improve an out-of-domain SMT system trained on the Europarl corpus (Koehn, 2005). Our hope is that the additional translation hypotheses could bring in new phrases or, more generally, new information that was not contained in the Europarl model. In order to facilitate comparisons, we use in-domain LMs for all setups.

We investigate two alternative ways of integrating the additional phrases into the existing SMT system: One is to take the hypothesis translation model described in Section 3.1, the other is to construct system-specific models constructed with only translations from one system at a time.

Although the Moses decoder is able to work with two phrase tables at once (Koehn and Schroeder, 2007), it is difficult to use this method when there is more than one additional model. The method requires tuning on at least six more features, which expands the search space for the translation task unnecessarily. We instead integrate the translation models from multiple sources by extending the phrase table. In contrast to the prior approach presented in (Chen et al., 2007) and (Eisele et al., 2008) which concatenates the phrase tables and adds new features as system markers, our extension method avoids duplicate entries in the final combined table.

Given a set of hypothesis translation models (derived from an arbitrary number of system outputs) and an original large translation model to be improved, we first sort the models by quality (see Section 3.3), always assigning the highest priority to the original model. The additional phrase tables are appended to the large model in sorted order such that only phrase pairs that were never seen before are included. Lastly, we add new features (in the form of additional columns in the phrase table) to the translation model to indicate each pair's origin.

3.3 System evaluation

Since both the system translations and the reference translations are available for the tuning

set, we first compare each output to the reference translation using BLEU (Papineni et al., 2001) and METEOR (Banerjee and Lavie, 2005) and a combined scoring scheme provided by the ULC toolkit (Gimenez and Marquez, 2008). In our experiments, we selected a subset of 5 systems for the combination, in most cases, based on BLEU.

On the other hand, some systems may be designed in a way that they deliver interesting unique translation segments. Therefore, we also measure the similarity among system outputs as shown in Table 2 in a given collection by calculating average similarity scores across every pair of outputs.

	de-en	fr-en	es-en	en-de	en-fr	en-es
Num.	20	23	28	15	16	9
Median	19.87	26.55	22.50	13.78	24.76	23.70
Range	16.37	17.06	9.74	4.75	11.05	13.94
Top 5	de-en	fr-en	es-en	en-de	en-fr	en-es
Median	22.26	27.93	26.43	15.21	26.62	26.61
Range	4.31	4.76	5.71	1.71	0.68	5.56

Table 1: Statistics of system outputs' BLEU scores

The range of BLEU scores cannot indicate the similarity of the systems. The direction with the most systems submitted is Spanish-English but their respective performances are very close to each other. As for the selected subset, the English-French systems have the most similar performance in terms of BLEU scores. The French-English translations have the largest range in BLEU but the similarity in this group is **not** the lowest.

	de-en	fr-en	es-en	en-de	en-fr	en-es
All	34.09	46.48	61.83	31.74	44.95	38.11
Selected	36.65	56.16	56.06	33.92	52.78	57.25

Table 2: Similarity of the system outputs

Ideally, we should select systems with highest quality scores and lowest similarity scores. For German-English, we selected the three with the highest METEOR scores and another two with high METEOR scores but low similarity scores to the first three. For the other language directions, we chose five systems from different institutions with the highest scores.

3.4 Language models

We use a standard n-gram language model for each target language using the monolingual training data provided in the translation task. These LMs are thus specific to the same domain as the input texts. Moreover, we also generate "*hypothesis*" LMs solely based on the given system outputs, that is, LMs that model how the candidate systems convey information in the target language. These LMs do not require any additional training data. Therefore, we do not require any training data other than the given system outputs by using the "hypothesis" language model and the "hypothesis" translation model.

3.5 Tuning

After building the models, it is essential to tune the SMT system to optimize the feature weights. We use Minimal Error Rate Training (Och, 2003) to maximize BLEU on the complete development data. Unlike the standard tuning procedure, we do not tune the final system directly. Instead, we obtain the weights using models built from the tuning portion of the system outputs.

For each combination variant, we first train models on the provided outputs corresponding to the tuning set. This system, called the *tuning system*, is also tuned on the tuning set. The initial weights of any additional features not included in the standard setting are set to 0. We then adapt the weights to the system built with translations corresponding to the test set. The procedure and the settings for building this system must be identical to that of the tuning system.

4 Experiments

The purpose of this exercise is to understand the nature of the system combination task in practice. Therefore, we restrict ourselves to the training data and system translations provided by the shared task. The types of the systems that produced the translations are assumed to be unknown. We report results for six translation directions between four languages.

4.1 Data and baseline

We build an SMT system from release v4 of the Europarl corpus (Koehn, 2005), following a standard routine using the Moses toolkit. The system also includes 5-gram language models trained on in-domain corpora of the respective target languages using SRILM (Stolcke, 2002).

The systems in this paper, including the baseline, are all tuned on the same 501-sentence tuning set. Note also that the provided n-best outputs are excluded in our experiments.

4.2 Results

The experiments include three different setups for direct system combination, involving only hypothesis translation models. System S_0, the baseline for this group, uses a hypothesis translation model built with all available system translations and a hypothesis LM (also from the machine-generated outputs). S_1 differs from S_0 in that the LM in S_1 is generated from a large news corpus. S_2 consists of translation models built with only the five selected systems. The BLEU scores of these systems are shown in Table 3.

	de-en	fr-en	es-en	en-de	en-fr	en-es
Top 1	21.16	30.91	28.54	14.96	26.55	27.84
Mean	17.29	23.78	21.39	12.76	22.96	21.43
S_0	20.46	27.50	23.35	13.95	27.29	25.59
S_1	21.76	28.05	25.49	15.16	27.70	26.09
S_2	21.71	24.98	27.26	15.62	24.28	25.22

Table 3: BLEU scores of direct system combination

When all outputs are included, the combined system can always produce translations better than most of the systems. When only a hypothesis LM is used, the BLEU scores are always higher than the average BLEU scores of the outputs. It even outperforms the top system for English-French. This simple setup (S_0) is certainly a feasible solution when no additional data is available and no system evaluation is possible. This approach appears to be more effective on typically difficult language pairs that involve German.

As for the systems with normal language models, neither of the systems ensure better translations. The translation quality is not completely determined by the number of included translations and their quality. On the other hand, the output set with higher diversity (Table 2) usually leads to better combination results. This observation is consistent with the results from the system integration experiments shown in Table 4.

	de-en	fr-en	es-en	en-de	en-fr	en-es
Bas	19.13	25.07	24.55	13.59	23.67	23.67
Med	17.99	24.56	20.70	13.19	24.19	22.12
All	21.40	28.00	27.75	15.21	27.20	26.41
Top5	21.70	26.01	28.53	15.52	27.87	27.92

Table 4: BLEU scores of integrated SMT systems (Bas: Baseline, Med: Median)

There are two variants in our experiments on system integration. *All* in Table 4 represents the system that integrates the complete hypothesis translation model with the Europarl model, while *Top 5* refers to the system that incorporates the five system-specific models separately. Both setups result in an improvement over the baseline Europarl-based SMT system. BLEU scores increase by up to 4.25 points. The integrated SMT system sometimes produces translations better than the best system (7 out of 12 cases).

5 Conclusion

This work uses the Moses toolkit to combine translations from multiple engines in a simple way. The experiments on six translation directions show interesting results: The final translations are always better than the majority of the given systems, while the combination performs better than the best system in half the cases. A similar approach was applied to improve an existing SMT system which was built in a domain different from the test task. We achieved improvements in all cases.

There are many possible future directions to continue this work. As we have shown, the quality of the combined system is more related to the diversity of the involved systems than to the number of the systems or their quality. Hand-picked systems lead to better combinations than those selected by BLEU scores. It would be interesting to develop a more comprehensive system selection strategy.

Acknowledgments

This work was supported by the EuroMatrix project (IST-034291) which is funded by the European Community under the Sixth Framework Programme for Research and Technological Development.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Yu Chen, Andreas Eisele, Christian Federmann, Eva Hasler, Michael Jellinghaus, and Silke Theison. 2007. Multi-engine machine translation with an open-source SMT decoder. In *Proceedings of*

WMT07, pages 193–196, Prague, Czech Republic, June. Association for Computational Linguistics.

Andreas Eisele, Christian Federmann, Hervé Saint-Amand, Michael Jellinghaus, Teresa Herrmann, and Yu Chen. 2008. Using Moses to integrate multiple rule-based machine translation engines into a hybrid system. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 179–182, Columbus, Ohio, June. Association for Computational Linguistics.

Jesus Gimenez and Lluis Marquez. 2008. A smorgasbord of features for automatic MT evaluation. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 195–198, Columbus, Ohio, June. Association for Computational Linguistics.

Xiaodong He, Mei Yang, Jianfeng Gao, Patrick Nguyen, and Robert Moore. 2008. Indirect-HMM-based hypothesis alignment for combining outputs from machine translation systems. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 98–107, Honolulu, Hawaii, October. Association for Computational Linguistics.

Philipp Koehn and Josh Schroeder. 2007. Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 224–227, Prague, Czech Republic, June. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch Mayne, Christopher Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbs. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of Annual meeting of the Association for Computation Linguistics (acl), demonstration session*, pages 177–180, Prague, Czech, June.

Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of MT Summit 2005*.

Evgeny Matusov, Nicola Ueffing, and Hermann Ney. 2006. Computing consensus translation from multiple machine translation systems using enhanced hypotheses alignment. In *Conference of the European Chapter of the Association for Computational Linguistics*, pages 33–40, Trento, Italy, April.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *ACL '03: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 160–167, Morristown, NJ, USA. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, Morristown, NJ, USA. Association for Computational Linguistics.

Antti-Veikko I. Rosti, Spyridon Matsoukas, and Richard M. Schwartz. 2007. Improved word-level system combination for machine translation. In *ACL*.

Andreas Stolcke. 2002. SRILM - an extensible language modeling toolkit. In *the 7th International Conference on Spoken Language Processing (ICSLP) 2002*, Denver, Colorado.