Machine Learning for Hybrid Machine Translation

Sabine Hunsicker
DFKI GmbH
Language Technology Lab
Saarbrücken, Germany
sabine.hunsicker@dfki.de

Chen Yu
DFKI GmbH
Language Technology Lab
Saarbrücken, Germany
yu.chen@dfki.de

Christian Federmann
DFKI GmbH
Language Technology Lab
Saarbrücken, Germany
cfedermann@dfki.de

Abstract

We describe a substitution-based system for hybrid machine translation (MT) that has been extended with machine learning components controlling its phrase selection. The approach is based on a rule-based MT (RBMT) system which creates template translations. Based on the rule-based generation parse tree and target-to-target alignments, we identify the set of "interesting" translation candidates from one or more translation engines which could be substituted into our translation templates. The substitution process is either controlled by the output from a binary classifier trained on feature vectors from the different MT engines, or it is depending on weights for the decision factors, which have been tuned using MERT. We are able to observe improvements in terms of BLEU scores over a baseline version of the hybrid system.

1 Introduction

In recent years, machine translation (MT) systems have achieved increasingly better translation quality. Still each paradigm has its own challenges: while statistical MT (SMT) systems suffer from a lack of grammatical structure, resulting in ungrammatical sentences, RBMT systems have to deal with a lack of lexical coverage. Hybrid architectures intend to combine the advantages of the individual paradigms to achieve an overall better translation.

Federmann et al. (2010) and Federmann and Hunsicker (2011) have shown that using a substitution-based approach can improve the translation quality of a baseline RBMT system. Our submission to WMT12 is a new, improved version following these approaches. The output of an RBMT engine serves as our translation backbone, and we substitute noun phrases by translations mined from other systems.

2 System Architecture

Our hybrid MT system combines translation output from:

a) the Lucy RBMT system, described in more detail in (Alonso and Thurmair, 2003);

b) the Linguatec RBMT system (Aleksic and Thurmair, 2011);

c) Moses (Koehn et al., 2007);

d) Joshua (Li et al., 2009).

Lucy provides us with the translation skeleton, which is described in more detail in Section 2.2 while systems *b)–d)* are aligned to this translation template and mined for substitution candidates. We give more detailed information on these systems in Section 2.3.

2.1 Basic Approach

We first identify "interesting" phrases inside the rule-based translation and then compute the most probable correspondences in the translation output from the other systems. For the resulting phrases, we apply a factored substitution method that decides whether the original RBMT phrase should be kept or rather be replaced by one of the candidate phrases. A schematic overview of our hybrid system and its main components is given in Figure 1.

Proceedings of the 7th Workshop on Statistical Machine Translation, pages 312–316,
Montréal, Canada, June 7-8, 2012.

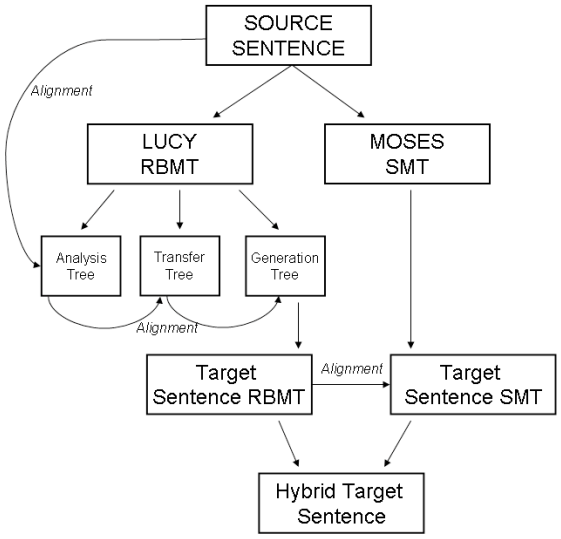

Figure 1: Schematic overview of the architecture of our substitution-based, hybrid MT system.

In previous years, it turned out that the alignment of the candidate translations to the source contained too many errors. In this version of our system, we thus changed the alignment method that connects the other translations. Only the rule-based template is aligned to the source. As we make use of the Lucy RBMT analysis parse trees, this alignment is very good. The other translations are now connected to the rule-based template using a confusion network approach. This also reduces computational efforts, as we now can compute the substitution candidates directly from the template without detouring over the source. During system training and tuning, this new approach has resulted in a reduced number of erroneous alignment links.

Additionally, we also changed our set of decision factors, increasing their total number. Whereas an older version of this system only used four factors, we now consider the following twelve factors:

1. **frequency:** frequency of a given candidate phrase compared to total number of candidates for the current phrase;
2. **LM(phrase):** language model (LM) score of the phrase;
3. **LM(phrase+1):** phrase with right-context;
4. **LM(phrase-1):** phrase with left-context;
5. **Part-of-speech match?:** checks if the part-of-speech tags of the left/right context match the current candidate phrase's context;
6. **LM(pos)** LM score for part-of-speech (PoS);
7. **LM(pos+1)** PoS with right-context;
8. **LM(pos-1)** PoS with left-context;
9. **Lemma** checks if the lemma of the candidate phrase fits the reference;
10. **LM(lemma)** LM score for the lemma;
11. **LM(lemma+1)** lemma with right-context;
12. **LM(lemma-1)** lemma with left-context.

The language model was trained using the SRILM toolkit (Stolcke, 2002), on the EuroParl (Koehn, 2005) corpus, and lemmatised or part-of-speech tagged versions, respectively. We used the TreeTagger (Schmid, 1994) for lemmatisation as well as part-of-speech tagging.

The substitution algorithm itself was also adapted. We investigated two machine learning approaches. In the previous version, the system used a hand-written decision tree to perform the substitution:

1. the first of the two new approaches consisted of machine learning this decision tree from annotated data;
2. the second approach was to assign a weight to each factor and using MERT tuning of these weights on a development set.

Both approaches are described in more detail later in Section 2.4.

2.2 Rule-Based Translation Templates

The Lucy RBMT system provides us with parse tree structures for each of the three phases of its transfer-based translation approach: *analysis*, *transfer* and *generation*. Out of these structures, we can extract linguistic phrases which later represent the "slots" for substitution. Previous work has shown that these structures are of a good grammatical quality due to the grammar Lucy uses.

2.3 Substitution Candidate Translations

Whereas in our previous work, we solely relied on candidates retrieved from SMT systems, this time we also included an additional RBMT system into the architecture. Knowing that statistical systems make similar errors, we hope to balance out this fact by exploiting also a system of a different paradigm, namely RBMT.

To create the statistical translations, we used state-of-the-art SMT systems. Both our Moses and Joshua systems were trained on the EuroParl corpus and News Commentary[1] training data. We performed tuning on the "newstest2011" data set using MERT.

We compile alignments between translations with the alignment module of MANY (Barrault, 2010). This module uses a modified version of TERp (Snover et al., 2009) and a set of different costs to create the best alignment between any two given sentences. In our case, each single candidate translation is aligned to the translation template that has been produced by the Lucy RBMT system. As we do not use the source in this alignment technique, we can use any translation system, regardless of whether this system provides us with a source-to-target alignment.

In earlier versions of this system, we compiled the source-to-target alignments for the candidate translations using GIZA++ (Och and Ney, 2003), but these alignments contained many errors. By using target-to-target alignments, we are able to reduce the amount of those errors which is, of course, preferred.

2.4 Substitution Approaches

Using the parse tree structures provided by Lucy, we extract "interesting" phrases for substitution. This includes noun phrases of various complexity, then simple verb phrases consisting of only the main verb, and finally adjective phrases. Through the target-to-target alignments we identify and collect the set of potential substitution candidates. Phrase substitution can be performed using two methods.

2.4.1 Machine-Learned Decision Tree

Previous work used hand-crafted rules. These are now replaced by a classifier which was trained on annotated data. Our training set D can formally be represented as

$$D = \{(x_i, y_i) | x_i \in \mathbb{R}^p, y_i \in \{-1, 1\}\}_{i=1}^{n} \quad (1)$$

where each x_i represents the *feature vector* for some sentence i while the y_i value contains the annotated class information. We use a binary classification scheme, simply defining 1 as "good" and -1 as "bad" translations.

In order to make use of machine (ML) learning methods such as decision trees (Breiman et al., 1984), Support Vector Machines (Vapnik, 1995), or the Perceptron (Rosenblatt, 1958) algorithm, we have to prepare our training set with a sufficiently large amount of annotated training instances.

To create the training data set, we computed the feature vectors and all possible substitution candidates for the WMT12 "newstest2011" development set. Human annotators were then given the task to assign to each candidate whether it was a "good" or a "bad" substitution. We used Appraise (Federmann, 2010) for the annotation, and collected a set of 24,996 labeled training instances with the help of six human annotators. Table 1 gives an overview of the data sets characteristics. The decision tree learned from this data replaces the hand-crafted rules.

2.4.2 Weights Tuned with MERT

Another approach we followed was to assign weights to the chosen decision factors and to use Minimal Error Rate Training to get the best weights. Using the twelve factors described in Section 2.1, we assign uniformly distributed weights and create n-best lists. Each n-best lists contains a total of $n+2$ hypotheses, with n being the number of candidate systems. It contains the Lucy template translations, the hybrid translation using the best candidates as well as a hypothesis for each candidate system. In the latter translation, each potential candidate for substitution is selected and replaces the original sub phrase in the baseline. The n-best list is

	Translation Candidates		
	Total	"good"	"bad"
Count	24,996	10,666	14,330

Table 1: Training data set characteristics

[1] Available at `http://www.statmt.org/wmt12/`

	Hybrid Systems			Baseline Systems			
	Baseline	+Decision Tree	+MERT	Lucy	Linguatec	Joshua	Moses
BLEU	13.9	14.2	**14.3**	14.0	14.7	14.6	**15.9**
BLEU-cased	13.5	13.8	**13.9**	13.7	14.2	13.5	**14.9**
TER	0.776	0.773	**0.768**	0.774	0.775	**0.772**	0.774

Table 2: Experimental results for all component and hybrid systems applied to the WMT12 "newstest2012" test set data for language pair English→German.

sorted by the final score of the feature vectors making up each hypothesis. We used Z-MERT (Zaidan, 2009) to optimise the set of feature weights on the "newstest2011" development set.

3 Evaluation

Using the "newstest2012" test set, we created baseline translations for the four MT systems used in our hybrid system. Then we performed three runs of our hybrid system:

a) a baseline run, using the factors and uniformly distributed weights;

b) a run using the weights trained on the development set;

c) a run using the decision tree learned from annotated data.

Table 2 shows the results for automatic metrics' scores. Besides BLEU (Papineni et al., 2001), we also report its case-sensitive variant, BLEU-cased, and TER (Snover et al., 2006) scores.

Comparing the scores, we see that both advanced hybrid methods perform better than the original, baseline hybrid as well as the Lucy baseline system. The MERT approach performs slightly better than the decision tree. This proves that using machine-learning to adapt the substitution approach results in better translation quality.

Other baseline systems, however, still outperform the hybrid systems. In part this is due to the fact that we are preserving the basic structure of the RBMT translation and do not reorder the new hybrid translation. To improve the hybrid approach further, there is more research required.

4 Conclusion and Outlook

In this paper, we have described how machine-learning approaches can be used to improve the phrase substitution component of a hybrid machine translation system.

We reported on two different approaches, the first using a binary classifier learned from annotated data, and the second using feature weights tuned with MERT. Both systems achieved improved automatic metrics' scores on the WMT12 "newstest2012" test set for the language pair English→German.

Future work will have to investigate ways how to achieve a closer integration of the individual baseline translations. This might be done by also taking into account reordering of the linguistic phrases as shown in the tree structures. We will also need to examine the differences between the classifier and MERT approach, to see whether we can integrate them to improve the selection process even further.

Also, we have to further evaluate the machine learning performance via, e.g., cross-validation-based tuning, to improve the prediction rate of the classifier model. We intend to explore other machine learning techniques such as SVMs as well.

Acknowledgments

This work has been funded under the Seventh Framework Programme for Research and Technological Development of the European Commission through the T4ME contract (grant agreement no.: 249119). It was also supported by the EuroMatrix-Plus project (IST-231720). We are grateful to the anonymous reviewers for their valuable feedback. Special thanks go to Hervé Saint-Amand for help with fixing the automated metrics scores.

References

Vera Aleksic and Gregor Thurmair. 2011. Personal Translator at WMT 2011. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 303–308, Edinburgh, Scotland, July. Association for Computational Linguistics.

Juan A. Alonso and Gregor Thurmair. 2003. The Comprendium Translator System. In *Proceedings of the Ninth Machine Translation Summit*.

Loïc Barrault. 2010. MANY : Open Source Machine Translation System Combination. *Prague Bulletin of Mathematical Linguistics, Special Issue on Open Source Tools for Machine Translation*, 93:147–155, January.

L. Breiman, J. Friedman, R. Olshen, and C. Stone. 1984. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA.

Christian Federmann and Sabine Hunsicker. 2011. Stochastic parse tree selection for an existing rbmt system. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 351–357, Edinburgh, Scotland, July. Association for Computational Linguistics.

Christian Federmann, Andreas Eisele, Yu Chen, Sabine Hunsicker, Jia Xu, and Hans Uszkoreit. 2010. Further experiments with shallow hybrid mt systems. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 77–81, Uppsala, Sweden, July. Association for Computational Linguistics.

Christian Federmann. 2010. Appraise: An Open-Source Toolkit for Manual Phrase-Based Evaluation of Translations. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May. European Language Resources Association (ELRA).

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of ACL Demo and Poster Sessions*, pages 177–180. Association for Computational Linguistics, June.

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

Zhifei Li, Chris Callison-Burch, Chris Dyer, Sanjeev Khudanpur, Lane Schwartz, Wren Thornton, Jonathan Weese, and Omar Zaidan. 2009. Joshua: An Open Source Toolkit for Parsing-Based Machine Translation. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 135–139, Athens, Greece, March. Association for Computational Linguistics.

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, Stroudsburg, PA, USA, April. Association for Computational Linguistics.

Franz Josef Och and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51, March.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. Bleu: a Method for Automatic Evaluation of Machine Translation. IBM Research Report RC22176(W0109-022), IBM.

F. Rosenblatt. 1958. The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. *Psychological Review*, 65:386–408.

Helmut Schmid. 1994. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of the International Conference on New Methods in Language Processing*, Manchester, UK.

Toby Segaran. 2007. *Programming Collective Intelligence: Building Smart Web 2.0 Applications*. O'Reilly, Beijing.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of the Conference of the Association for Machine Translation in the Americas*, pages 223–231.

Matthew Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz. 2009. Fluency, Adequacy, or HTER? Exploring Different Human Judgments with a Tunable MT Metric. In *Proceedings of the Fourth Workshop on Statistical Machine Translation at the 12th Meeting of the European Chapter of the Association for Computational Linguistics (EACL-2009)*, Athens, Greece, March.

Andreas Stolcke. 2002. SRILM - An Extensible Language Modeling Toolkit. In *Proceedings of the International Conference on Spoken Language Processing*, pages 257–286, November.

V. N. Vapnik. 1995. *The Nature of Statistical Learning Theory*. Springer, New York.

Omar F. Zaidan. 2009. Z-MERT: A fully configurable open source tool for minimum error rate training of machine translation systems. *The Prague Bulletin of Mathematical Linguistics*, 91:79–88, January.