**LIMSI @ WMT11**

**Alexandre Allauzen He´le`ne Bonneau-Maynard Hai-Son Le**

**Aure´lien Max Guillaume Wisniewski Franc¸ois Yvon**

Univ. Paris-Sud and LIMSI-CNRS B.P. 133, 91403 Orsay cedex, France

**Gilles Adda Josep M. Crego Adrien Lardilleux Thomas Lavergne Artem Sokolov**

LIMSI-CNRS

B.P. 133, 91403 Orsay cedex, France

**Abstract**

This paper describes LIMSI’s submissions to the Sixth Workshop on Statistical Machine Translation. We report results for the French- English and German-English shared transla- tion tasks in both directions. Our systems use *n*-code, an open source Statistical Ma- chine Translation system based on bilingual *n*-grams. For the French-English task, we fo- cussed on finding efficient ways to take ad- vantage of the large and heterogeneous train- ing parallel data. In particular, using a sim- ple filtering strategy helped to improve both processing time and translation quality. To translate from English to French and Ger- man, we also investigated the use of the SOUL language model in Machine Trans- lation and showed significant improvements with a 10-gram SOUL model. We also briefly report experiments with several alternatives to the standard *n*-best MERT procedure, leading to a significant speed-up.

**1 Introduction**

This paper describes LIMSI’s submissions to the Sixth Workshop on Statistical Machine Translation, where LIMSI participated in the French-English and German-English tasks in both directions. For this evaluation, we used *n*-code, our in-house Statistical Machine Translation (SMT) system which is open- source and based on bilingual *n*-grams.

This paper is organized as follows. Section 2 pro- vides an overview of *n*-code, while the data pre- processing and filtering steps are described in Sec- tion 3. Given the large amount of parallel data avail-

able, we proposed a method to filter the French- English *GigaWord* corpus (Section 3.2). As in our previous participations, data cleaning and filtering constitute a non-negligible part of our work. This includes detecting and discarding sentences in other languages; removing sentences which are also in- cluded in the provided development sets, as well as parts that are repeated (for the monolingual news data, this can reduce the amount of data by a fac- tor 3 or 4, depending on the language and the year); normalizing the character set (non-utf8 characters which are aberrant in context, or in the case of the *GigaWord* corpus, a lot of non-printable and thus in- visible control characters such as *EOT (end of trans- mission)*1).

For target language modeling (Section 4), a stan- dard back-off *n*-gram model is estimated and tuned as described in Section 4.1. Moreover, we also in- troduce in Section 4.2 the use of the SOUL lan- guage model (LM) (Le et al., 2011) in SMT. Based on neural networks, the SOUL LM can handle an arbitrary large vocabulary and a high order marko- vian assumption (up to 10-gram in this work). Fi- nally, experimental results are reported in Section 5 both in terms of BLEU scores and translation edit rates (TER) measured on the provided *newstest2010* dataset.

**2 System Overview**

Our in-house *n*-code SMT system implements the bilingual *n*-gram approach to Statistical Machine Translation (Casacuberta and Vidal, 2004). Given a

1 This kind of characters was used for Teletype up to the sev- enties or early eighties.

309

*Proceedings of the 6th Workshop on Statistical Machine Translation*, pages 309–315, Edinburgh, Scotland, UK, July 30–31, 2011. *Q*c 2011 Association for Computational Linguistics

source sentence *sJ* , a translation hypothesis *t*ˆ*I* is de-

a word-aligned corpus (using MGIZA++2 with de-

1 1

fined as the sentence which maximizes a linear com-

bination of feature functions:

fault settings) in such a way that a unique segmenta-

tion of the bilingual corpus is achieved, allowing to estimate the *n*-gram model. Figure 1 presents a sim-

*t*ˆ*I*

1

( *M* )

*J I*

ple example illustrating the unique tuple segmenta-

= arg max

*I*

*t*

1

∑ *λmhm*(*s*1*, t*1)

*m*=1

(1)

tion for a given word-aligned pair of sentences (top).

where *sJ* and *tI* respectively denote the source and



1 1



the target sentences, and *λm* is the weight associated

with the feature function *hm*. The translation fea- ture is the log-score of the translation model based on bilingual units called *tuples*. The probability as-



signed to a sentence pair by the translation model is



estimated by using the *n*-gram assumption:



*K*

*p*(*sJ , tI* ) =

*p*((*s, t*)*k |*(*s, t*)*k*

1 *. . .* (*s, t*) )

1 1 ∏

*k*=1

*− k−n*+1

Figure 1: Tuple extraction from a sentence pair.

where *s* refers to a source symbol (*t* for target) and (*s, t*)*k* to the *kth* tuple of the given bilingual sentence pair. It is worth noticing that, since both languages are linked up in tuples, the context information pro- vided by this translation model is bilingual. In ad- dition to the translation model, *eleven* feature func- tions are combined: a *target-language model* (see Section 4 for details); four *lexicon models*; two *lex- icalized reordering models* (Tillmann, 2004) aim- ing at predicting the orientation of the next transla- tion unit; a “weak” distance-based *distortion model*; and finally a *word-bonus model* and a *tuple-bonus model* which compensate for the system preference for short translations. The four *lexicon models* are similar to the ones used in a standard phrase-based system: two scores correspond to the relative fre- quencies of the tuples and two lexical weights are estimated from the automatically generated word alignments. The weights associated to feature func- tions are optimally combined using a discriminative training framework (Och, 2003) (Minimum Error Rate Training (MERT), see details in Section 5.4), using the provided *newstest2009* data as develop- ment set.

**2.1 Training**

Our translation model is estimated over a training

The resulting sequence of tuples *(1)* is further re- fined to avoid *NULL* words in the source side of the tuples *(2)*. Once the whole bilingual training data is segmented into tuples, *n*-gram language model prob- abilities can be estimated. In this example, note that the English source words *perfect* and *translations* have been reordered in the final tuple segmentation, while the French target words are kept in their orig- inal order.

**2.2 Inference**

During decoding, source sentences are encoded in the form of word lattices containing the most promising reordering hypotheses, so as to reproduce the word order modifications introduced during the tuple extraction process. Hence, at decoding time, only those encoded reordering hypotheses are trans- lated. Reordering hypotheses are introduced using a set of reordering rules automatically learned from the word alignments.

In the previous example, the rule [*perfect transla- tions ❀ translations perfect*] produces the swap of

the English words that is observed for the French and English pair. Typically, part-of-speech (POS) information is used to increase the generalization power of such rules. Hence, rewriting rules are built

using POS rather than surface word forms. Refer

corpus composed of tuple sequences using classi-

cal smoothing techniques. Tuples are extracted from

2 <http://geek.kyloo.net/software>

to (Crego and Marin˜ o, 2007) for details on tuple ex- traction and reordering rules.

**3 Data Pre-processing and Selection**

We used all the available parallel data allowed in the constrained task to compute the word align- ments, except for the French-English tasks where the United Nation corpus was not used to train our translation models. To train the target language models, we also used all provided data and mono- lingual corpora released by the LDC for French and English. Moreover, all parallel corpora were POS-tagged with the TreeTagger (Schmid, 1994). For German, the fine-grained POS information used for pre-processing was computed by the RFTag- ger (Schmid and Laws, 2008).

**3.1 Tokenization**

We took advantage of our in-house text process- ing tools for the tokenization and detokenization steps (De´chelotte et al., 2008). Previous experi- ments have demonstrated that better normalization tools provide better BLEU scores (Papineni et al.,

2002). Thus all systems are built in “true-case.”

As German is morphologically more complex than English, the default policy which consists in treating each word form independently is plagued with data sparsity, which poses a number of diffi- culties both at training and decoding time. Thus, to translate from German to English, the German side was normalized using a specific pre-processing scheme (described in (Allauzen et al., 2010)), which aims at reducing the lexical redundancy and splitting complex compounds.

Using the same pre-processing scheme to trans- late from English to German would require to post- process the output to undo the pre-processing. As in our last year’s experiments (Allauzen et al., 2010), this pre-processing step could be achieved with a two-step decoding. However, by stacking two de- coding steps, we may stack errors as well. Thus, for this direction, we used the German tokenizer pro- vided by the organizers.

**3.2 Filtering the *GigaWord* Corpus**

The available parallel data for English-French in- cludes a large Web corpus, referred to as the *Giga- Word* parallel corpus. This corpus is very noisy, and

contains large portions that are not useful for trans- lating news text. The first filter aimed at detecting foreign languages based on perplexity and lexical coverage. Then, to select a subset of parallel sen- tences, trigram LMs were trained for both French and English languages on a subset of the available News data: the French (resp. English) LM was used to rank the French (resp. English) side of the cor- pus, and only those sentences with perplexity above a given threshold were selected. Finally, the two se- lected sets were intersected. In the following exper- iments, the threshold was set to the median or upper quartile value of the perplexity. Therefore, half (or

75%) of this corpus was discarded.

**4 Target Language Modeling**

Neural networks, working on top of conventional *n*-gram models, have been introduced in (Bengio et al., 2003; Schwenk, 2007) as a potential means to improve conventional *n*-gram language models (LMs). However, probably the major bottleneck with standard NNLMs is the computation of poste- rior probabilities in the output layer. This layer must contain one unit for each vocabulary word. Such a design makes handling of large vocabularies, con- sisting of hundreds thousand words, infeasible due to a prohibitive growth in computation time. While recent work proposed to estimate the *n*-gram dis- tributions only for the most frequent words (short- list) (Schwenk, 2007), we explored the use of the SOUL (Structured OUtput Layer Neural Network) language model for SMT in order to handle vocabu- laries of arbitrary sizes.

Moreover, in our setting, increasing the order of standard *n*-gram LM did not show any significant improvement. This is mainly due to the data spar- sity issue and to the drastic increase in the number of parameters that need to be estimated. With NNLM however, the increase in context length at the input layer results in only a linear growth in complexity in the worst case (Schwenk, 2007). Thus, training longer-context neural network models is still feasi- ble, and was found to be very effective in our system.

**4.1 Standard** *n***-gram Back-off Language**

**Models**

To train our language models, we assumed that the test set consisted in a selection of news texts dat- ing from the end of 2010 to the beginning of 2011. This assumption was based on what was done for the 2010 evaluation. Thus, for each language, we built a development corpus in order to optimize the vocabulary and the target language model.

**Development set and vocabulary** In order to cover different periods, two development sets were used. The first one is *newstest2008*. This corpus is two years older than the targeted time period; there- fore, a second development corpus named *dev2010-*

*2011* was collected by randomly sampling bunches of 5 consecutive sentences from the provided news data of 2010 and 2011.

To estimate such large LMs, a vocabulary was first defined for each language by including all tokens observed in the Europarl and News- Commentary corpora. For French and English, this vocabulary was then expanded with all words that occur more than 5 times in the French-English *Gi- gaWord* corpus, and with the most frequent proper names taken from the monolingual news data of

2010 and 2011. As for German, since the amount of training data was smaller, the vocabulary was ex- panded with the most frequent words observed in the monolingual news data of 2010 and 2011. This pro- cedure resulted in a vocabulary containing around

500k words in each language.

**Language model training** All the training data al- lowed in the constrained task were divided into sev- eral sets based on dates or genres (resp. 9 and 7 sets for English and French). On each set, a stan- dard 4-gram LM was estimated from the 500k words vocabulary using absolute discounting interpolated with lower order models (Kneser and Ney, 1995; Chen and Goodman, 1998).

All LMs except the one trained on the news cor- pora from 2010-2011 were first linearly interpolated. The associated coefficients were estimated so as to minimize the perplexity evaluated on *dev2010-2011*. The resulting LM and the 2010-2011 LM were fi- naly interpolated with *newstest2008* as development data. This procedure aims to avoid overestimating

the weight associated to the 2010-2011 LM.

**4.2 The SOUL Model**

We give here a brief overview of the SOUL LM; refer to (Le et al., 2011) for the complete training procedure. Following the classical work on dis- tributed word representation (Brown et al., 1992), we assume that the output vocabulary is structured by a clustering tree, where each word belongs to only one class and its associated sub-classes. If *wi* denotes the *i*-th word in a sentence, the sequence *c*1:*D*(*wi*) = *c*1*, . . . , cD* encodes the path for the word *wi* in the clustering tree, with *D* the depth of the tree, *cd* (*wi*) a class or sub-class assigned to *wi*, and *cD*(*wi*) the leaf associated with *wi* (the word itself). The *n*-gram probability of *wi* given its history *h* can then be estimated as follows using the chain rule:

*D*

*P*(*wi|h*) = *P*(*c*1(*wi*)*|h*) ∏ *P*(*cd* (*wi*)*|h, c*1:*d−*1)

*d*=2

Figure 2 represents the architecture of the NNLM to estimate this distribution, for a tree of depth *D* = 3. The SOUL architecture is the same as for the standard model up to the output layer. The main difference lies in the output structure which in- volves several layers with a softmax activation func- tion. The first softmax layer *(class layer)* estimates the class probability *P*(*c*1(*wi*)*|h*), while other out- put *sub-class layers* estimate the sub-class proba- bilities *P*(*cd* (*wi*)*|h, c*1:*d−*1). Finally, the *word layers* estimate the word probabilities *P*(*cD*(*wi*)*|h, c*1:*D−*1). Words in the short-list are a special case since each of them represents its own class without any sub- classes (*D* = 1 in this case).

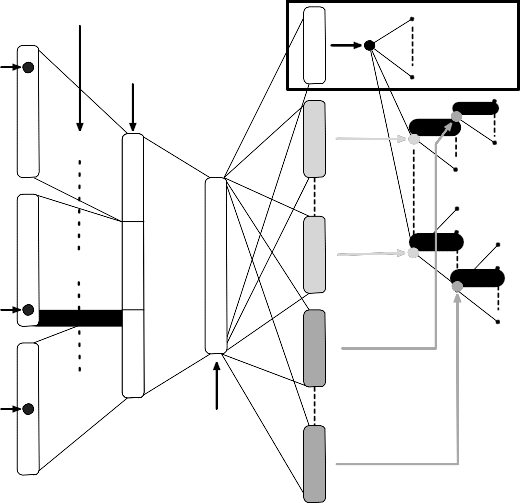
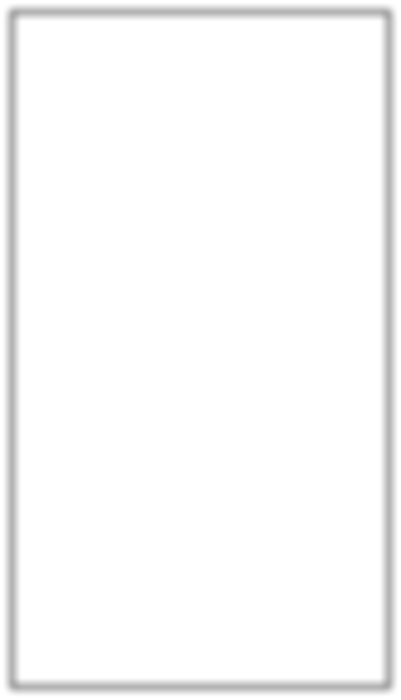
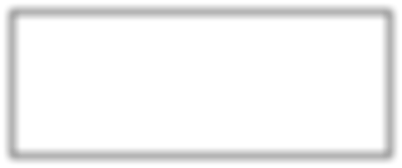
**5 Experimental Results**

The experimental results are reported in terms of BLEU and translation edit rate (TER) using the *newstest2010* corpus as evaluation set. These auto- matic metrics are computed using the scripts pro- vided by the NIST after a detokenization step.

**5.1 English-French**

Compared with last year evaluation, the amount of available parallel data has drastically increased with about 33M of sentence pairs. It is worth noticing

shared context space



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *System* | en2fr | | fr2en | |
| BLEU | *TER* | BLEU | *TER* |
| All  Upper quartile  Median | 27.4  27.8  28.1 | 56.6  56.3  56.0 | 26.8  28.4  28.6 | 55.0  53.8  53.5 |

wi-1

input layer

}short list

w i-2 w i-3

R

R W

ih

R

hidden layer:

tanh activation

sub-class layers

word layers

Table 1: **English-French** translation results in terms of BLEU score and TER estimated on *newstest2010* with the NIST script. *All* means that the translation model is trained on *news-commentary*, *Europarl*, and the whole *GigaWord*. The rows *upper quartile* and *median* corre- spond to the use of a filtered version of the *GigaWord*.

of data is the mismatch between the target vocab- ulary derived from the translation model and that of the LM. The translation model may generate words

Figure 2: Architecture of the Structured Output Layer

Neural Network language model.

that the provided corpora are not homogeneous, nei- ther in terms of genre nor in terms of topics. Never- theless, the most salient difference is the noise car- ried by the *GigaWord* and the *United Nation* cor- pora. The former is an automatically collected cor- pus drawn from different websites, and while some parts are indeed relevant to translate news texts, us- ing the whole *GigaWord* corpus seems to be harm- ful. The latter *(United Nation)* is obviously more homogeneous, but clearly out of domain. As an il- lustration, discarding the *United Nation* corpus im- proves performance slightly.

Table 1 summarizes some of our attempts at deal- ing with such a large amount of parallel data. As stated above, translation models are trained with the *news-commentary*, *Europarl*, and *GigaWord* cor- pora. For this last data set, results show the reward of sentence pair selection as described in Section 3.2. Indeed, filtering out 75% of the corpus yields to a significant BLEU improvement when translating from English to French and of 1 point in the other direction (line *upper quartile* in Table 1). More- over, a larger selection (50% in the *median* line) still increases the overall performance. This shows the room left for improvement by a more accurate data selection process such as a well optimized thresh- old in our approach, or a more sophisticated filtering strategy (see for example (Foster et al., 2010)).

Another issue when using such a large amount

which are unknown to the LM, and their probabili- ties could be overestimated. To avoid this behaviour, the probability of unknown words for the target LM is penalized during the decoding step.

**5.2 English-German**

For this translation task, we compare the impact of two different POS-taggers to process the German part of the parallel data. The results are reported in Table 2. Results show that to translate from En- glish to German, the use of a fine-grained POS infor- mation (RFTagger) leads to a slight improvement, whereas it harms the source reordering model in the other direction. It is worth noticing that to translate from German to English, the RFTagger is always used during the data pre-processing step, while a dif- ferent POS tagger may be involved for the source reordering model training.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *System* | en2de | | de2en | |
| BLEU | *TER* | BLEU | *TER* |
| RFTagger  TreeTagger | 22.8  23.1 | 60.1  59.4 | 16.3  16.2 | 66.0  66.0 |

Table 2: Translation results in terms of BLEU score and translation edit rate (TER) estimated on *newstest2010* with the NIST scoring script.

**5.3 The SOUL Model**

As mentioned in Section 4.2, the order of a con- tinuous *n*-gram model such as the SOUL LM can be raised without a prohibitive increase in complex- ity. We summarize in Table 3 our experiments with

SOUL LMs of orders 4, 6, and 10. The SOUL LM is introduced in the SMT pipeline by rescoring the *n*-best list generated by the decoder, and the asso- ciated weight is tuned with MERT. We observe for the English-French task: a BLEU improvement of

0.3, as well as a similar trend in TER, when intro- ducing a 4-gram SOUL LM; an additional BLEU improvement of 0.3 when increasing the order from

4 to 6; and a less important gain with the 10-gram SOUL LM. In the end, the use of a 10-gram SOUL LM achieves a 0.7 BLEU improvement and a TER decrease of 0.8. The results on the English-German task show the same trend with a 0.5 BLEU point improvement.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SOUL LM | en2fr | | en2de | |
| BLEU | *TER* | BLEU | *TER* |
| without | 28.1 | 56.0 | 16.3 | 66.0 |
| 4-gram  6-gram  10-gram | 28.4  28.7  28.8 | 55.5  55.3  55.2 | 16.5  16.7  16.8 | 64.9  64.9  64.6 |

Table 3: Translation results from English to French and

English to German measured on *newstest2010* using a

100-best rescoring with SOUL LMs of different orders.

**5.4 Optimization Issues**

Along with MIRA (Margin Infused Relaxed Al- gorithm) (Watanabe et al., 2007), MERT is the most widely used algorithm for system optimiza- tion. However, standard MERT procedure is known to suffer from instability of results and very slow training cycle with approximate estimates of one de- coding cycle for each training parameter. For this year’s evaluation, we experimented with several al- ternatives to the standard *n*-best MERT procedure, namely, MERT on word lattices (Macherey et al.,

2008) and two differentiable variants to the BLEU objective function optimized during the MERT cy- cle. We have recast the former in terms of a spe- cific semiring and implemented it using a general- purpose finite state automata framework (Sokolov and Yvon, 2011). The last two approaches, hereafter referred to as ZHN and BBN, replace the BLEU objective function, with the usual BLEU score on *expected n-gram counts* (Rosti et al., 2010) and with an *expected BLEU score* for normal *n*-gram counts (Zens et al., 2007), respectively. All expecta-

tions (of the *n*-gram counts in the first case and the BLEU score in the second) are taken over all hy- potheses from *n*-best lists for each source sentence.

Experiments with the alternative optimization methods achieved virtually the same performance in terms of BLEU score, but 2 to 4 times faster. Neither approach, however, showed any consistent and sig- nificant improvement for the majority of setups tried (with the exception of the BBN approach, that had almost always improved over *n*-best MERT, but for the sole French to English translation direction). Ad- ditional experiments with 9 complementary transla- tion models as additional features were performed with lattice-MERT, but neither showed any substan- tial improvement. In the view of these rather incon- clusive experiments, we chose to stick to the classi- cal MERT for the submitted results.

**6 Conclusion**

In this paper, we described our submissions to WMT’11 in the French-English and German- English shared translation tasks, in both directions. For this year’s participation, we only used *n*-code, our open source Statistical Machine Translation sys- tem based on bilingual *n*-grams. Our contributions are threefold. First, we have shown that *n*-gram based systems can achieve state-of-the-art perfor- mance on large scale tasks in terms of automatic metrics such as BLEU. Then, as already shown by several sites in the past evaluations, there is a signifi- cant reward for using data selection algorithms when dealing with large heterogeneous data sources such as the *GigaWord*. Finally, the use of a large vocab- ulary continuous space language model such as the SOUL model has enabled to achieve significant and consistent improvements. For the upcoming evalua- tion(s), we would like to suggest that the important work of data cleaning and pre-processing could be shared among all the participants instead of being done independently several times by each site. Re- ducing these differences could indeed help improve the reliability of SMT systems evaluation.

**Acknowledgment**

This work was achieved as part of the Quaero Pro- gramme, funded by OSEO, French State agency for innovation.

**References**

Alexandre Allauzen, Josep M. Crego, ˙Ilknur Durgar El- Kahlout, and Francois Yvon. 2010. LIMSI’s statis- tical translation systems for WMT’10. In *Proc. of the Joint Workshop on Statistical Machine Translation and MetricsMATR*, pages 54–59, Uppsala, Sweden.

Yoshua Bengio, Re´jean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic lan- guage model. *JMLR*, 3:1137–1155.

P.F. Brown, P.V. de Souza, R.L. Mercer, V.J. Della Pietra, and J.C. Lai. 1992. Class-based n-gram models of nat- ural language. *Computational Linguistics*, 18(4):467–

479.

Francesco Casacuberta and Enrique Vidal. 2004. Ma- chine translation with inferred stochastic finite-state transducers. *Computational Linguistics*, 30(3):205–

225.

Stanley F. Chen and Joshua T. Goodman. 1998. An empirical study of smoothing techniques for language modeling. Technical Report TR-10-98, Computer Sci- ence Group, Harvard University.

Josep Maria Crego and Jose´ Bernardo Marin˜ o. 2007. Im- proving statistical MT by coupling reordering and de- coding. *Machine Translation*, 20(3):199–215.

Daniel De´chelotte, Gilles Adda, Alexandre Allauzen, Olivier Galibert, Jean-Luc Gauvain, He´le`ne Mey- nard, and Franc¸ois Yvon. 2008. LIMSI’s statisti- cal translation systems for WMT’08. In *Proc. of the NAACL-HTL Statistical Machine Translation Work- shop*, Columbus, Ohio.

George Foster, Cyril Goutte, and Roland Kuhn. 2010.

Discriminative instance weighting for domain adapta- tion in statistical machine translation. In *Proceedings of the 2010 Conference on Empirical Methods in Natu- ral Language Processing*, pages 451–459, Cambridge, MA, October.

Reinhard Kneser and Herman Ney. 1995. Improved backing-off for m-gram language modeling. In *Pro- ceedings of the International Conference on Acoustics, Speech, and Signal Processing, ICASSP’95*, pages

181–184, Detroit, MI.

Hai-Son Le, Ilya Oparin, Alexandre Allauzen, Jean-Luc Gauvain, and Franc¸ois Yvon. 2011. Structured output layer neural network language model. In *IEEE Inter- national Conference on Acoustics, Speech and Signal Processing (ICASSP 2011)*, Prague (Czech Republic),

22-27 May.

Wolfgang Macherey, Franz Josef Och, Ignacio Thayer, and Jakob Uszkoreit. 2008. Lattice-based minimum error rate training for statistical machine translation. In *Proc. of the Conf. on EMNLP*, pages 725–734.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *ACL ’03: Proc. of*

*the 41st Annual Meeting on Association for Computa- tional Linguistics*, pages 160–167.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei- Jing Zhu. 2002. BLEU: a method for automatic eval- uation of machine translation. In *ACL ’02: Proc. of the 40th Annual Meeting on Association for Compu- tational Linguistics*, pages 311–318. Association for Computational Linguistics.

Antti-Veikko I. Rosti, Bing Zhang, Spyros Matsoukas, and Richard Schwartz. 2010. BBN system description for wmt10 system combination task. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Trans- lation and MetricsMATR*, WMT ’10, pages 321–326, Stroudsburg, PA, USA. Association for Computational Linguistics.

Helmut Schmid and Florian Laws. 2008. Estimation of conditional probabilities with decision trees and an application to fine-grained POS tagging. In *Proceed- ings of the 22nd International Conference on Com- putational Linguistics (Coling 2008)*, pages 777–784, Manchester, UK, August.

Helmut Schmid. 1994. Probabilistic part-of-speech tag- ging using decision trees. In *Proc. of International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK.

Holger Schwenk. 2007. Continuous space language models. *Computer, Speech & Language*, 21(3):492–

518.

Artem Sokolov and Franc¸ois Yvon. 2011. Minimum er- ror rate training semiring. In *Proceedings of the 15th Annual Conference of the European Association for Machine Translation*, EAMT’2011, May.

Christoph Tillmann. 2004. A unigram orientation model for statistical machine translation. In *Proceedings of HLT-NAACL 2004*, pages 101–104. Association for Computational Linguistics.

Taro Watanabe, Jun Suzuki, Hajime Tsukada, and Hideki Isozaki. 2007. Online large-margin training for sta- tistical machine translation. In *Proceedings of the*

*2007 Joint Conference on Empirical Methods in Nat- ural Language Processing and Computational Natu- ral Language Learning (EMNLP-CoNLL)*, pages 764–

773, Prague, Czech Republic.

Richard Zens, Sasa Hasan, and Hermann Ney. 2007.

A systematic comparison of training criteria for sta- tistical machine translation. In *Proceedings of the*

*2007 Joint Conference on Empirical Methods in Nat- ural Language Processing and Computational Natu- ral Language Learning (EMNLP-CoNLL)*, pages 524–

532.