The Edit Distance Transducer in Action: The University of Cambridge English-German System at WMT16

Felix Stahlberg and **Eva Hasler** and **Bill Byrne**
Department of Engineering
University of Cambridge
Cambridge, UK

Abstract

This paper presents the University of Cambridge submission to WMT16. Motivated by the complementary nature of syntactical machine translation and neural machine translation (NMT), we exploit the synergies of Hiero and NMT in different combination schemes. Starting out with a simple neural lattice rescoring approach, we show that the Hiero lattices are often too narrow for NMT ensembles. Therefore, instead of a hard restriction of the NMT search space to the lattice, we propose to loosely couple NMT and Hiero by composition with a modified version of the edit distance transducer. The loose combination outperforms lattice rescoring, especially when using multiple NMT systems in an ensemble.

1 Introduction

Previous work suggests that syntactic machine translation such as Hiero (Chiang, 2007) and Neural Machine Translation (NMT) (Kalchbrenner and Blunsom, 2013; Sutskever et al., 2014; Cho et al., 2014; Bahdanau et al., 2015) are very different and have complementary strengths and weaknesses (Neubig et al., 2015; Stahlberg et al., 2016). Recent attempts to combine syntactic SMT and NMT report large gains over both baselines. Authors in (Neubig et al., 2015) used NMT to rescore n-best lists which were generated with a syntax-based system. They report that even with 1000-best lists, the gains of using the NMT rescorer often do not saturate. Syntactically Guided NMT (Stahlberg et al., 2016, SGNMT) constrains the NMT search space to Hiero translation lattices which contain significantly more hypotheses than n-best lists. In SGNMT, an NMT beam decoder with a relatively small beam can explore spaces much larger than n-best lists, yielding BLEU score improvements with far fewer expensive NMT evaluations.

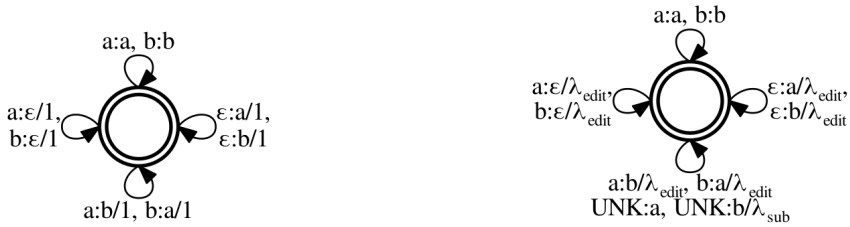
However, these rescoring approaches enforce an exact match between the NMT and syntactic decoders. In general, this kind of hard restriction is best avoided when combining diverse systems (Liu et al., 2009; Frederking et al., 1994). For example, in speech recognition, ROVER (Fiscus, 1997) is a system combination approach based on a soft voting scheme. In machine translation, minimum Bayes-risk (MBR) decoding (Kumar and Byrne, 2004) can be used to combine multiple systems (de Gispert et al., 2009). MBR also does not enforce exact agreement between systems as it distinguishes between the *hypothesis space* and the *evidence space* (Goel and Byrne, 2000; Tromble et al., 2008).

We find that Hiero lattices generated by grammars extracted with the usual heuristics (Chiang, 2007) do not provide enough variety to explore the full potential of neural models, especially when using NMT ensembles. Therefore, we present a "soft" lattice-based combination scheme which uses standard operations on finite state transducers such as composition. Our method replaces the hard combination in previous methods with a similarity measure based on the edit distance, and gives the NMT decoder more freedom to diverge from the Hiero translations. We find that this loose coupling scheme is especially useful when using NMT ensembles.

2 Combining Hiero and NMT via Edit Distance Transducer

In contrast to the strict coupling in SGNMT, we propose to loosely couple Hiero and NMT via an edit distance transducer and shortest distance

Proceedings of the First Conference on Machine Translation, Volume 2: Shared Task Papers, pages 377–384,
Berlin, Germany, August 11-12, 2016.



(a) Standard edit distance transducer.

(b) Modified edit distance transducer E. 'a' is an NMT OOV.

Figure 1: "Flower automata" for calculating edit distances over the alphabet $\{\text{a}, \text{b}, \text{UNK}\}$.

search. With loose coupling, the NMT decoder is not restricted to the Hiero lattice as in previous work, but runs independently to produce translation lattices on its own, which are then combined with the Hiero lattices. The combination does not require an exact match. Instead, we will describe a procedure for combining NMT and Hiero that captures similarity under the edit distance and both the NMT and Hiero translation system scores. This scheme is implemented efficiently using standard FST operations (Allauzen et al., 2007). First, we introduce the FST composition operation and the edit distance transducer. We will describe the whole pipeline in Sec. 2.3.

2.1 Composition of Finite State Transducers

The composition of two weighted transducers T_1, T_2 (denoted as $T_1 \circ T_2$) over a semiring $(\mathbb{K}, \oplus, \otimes)$ is defined following (Mohri, 2004)

$$[T_1 \circ T_2](x, y) = \bigoplus_z T_1(x, z) \otimes T_2(z, y). \quad (1)$$

We will make extensive use of this operation as tool for building complex automata which make use of both the NMT and Hiero translation lattices.

2.2 The Edit Distance Transducer

Composition can be used together with a "flower automaton" to calculate the edit distance between two sequences (Mohri, 2003). The edit distance transducer shown in Fig. 1(a) transduces a sequence x to another sequence y over the alphabet $\{\text{a}, \text{b}\}$ and accumulates the number of edit operations via the transitions with cost 1. In our case, x corresponds to an NMT hypothesis which is to be combined with a Hiero hypothesis y. In contrast to SGNMT, where we require an exact match between NMT and Hiero (up to UNKs), our edit-distance-based scheme allows different hypotheses to be combined. We replaced the standard definition of the edit distance transducer (Mohri, 2003) by a finer-grained model designed to work well for combining NMT and Hiero. Instead of uniform costs, we lower the cost for UNK substitutions as we want to encourage substituting NMT UNKs by words in the Hiero translation. We distinguish between three types of edit operations.

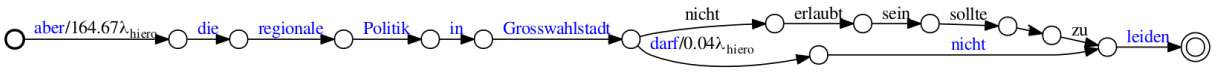
- **Type I**: Substituting UNK with a word outside the NMT vocabulary is free.
- **Type II**: For substitutions of UNK with a word inside the NMT vocabulary we add the cost λ_{sub}.
- **Type III**: All other edit operations are penalized with cost λ_{edit} (and $\lambda_{edit} > \lambda_{sub}$).

We will refer to the modified edit distance transducer as E. Fig. 1(b) shows E over the alphabet $\{\text{a}, \text{b}, \text{UNK}\}$, with 'a' being an NMT OOV.

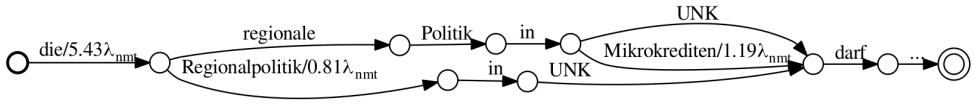
2.3 Loose Coupling of Hiero and NMT

Our edit-distance-based scheme combines an NMT translation lattice N with a Hiero translation lattice H. Weights in N and H are scaled by λ_{nmt} and λ_{hiero}, respectively. The similarity measure between NMT and Hiero translations is parametrized with λ_{ins}, λ_{edit}, and λ_{sub}. We keep the various costs separated by using transducers with tropical sparse tuple vector semirings (Iglesias et al., 2015). Instead of single real-valued arc weights, this semiring uses vectors which can hold multiple features. The inner product of these vectors with a constant parameter vector determines the final weights on the arcs[1]. The sparse tuple vector semiring enables us to optimize the λ-parameters with LMERT (Macherey et al., 2008) on a development set.

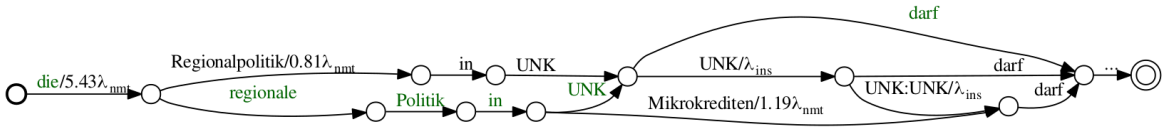
[1]The ucam-smt tutorial contains details to the tropical sparse tuple vector semiring: http://ucam-smt.github.io/tutorial/basictrans.html#lmert_veclats_tst



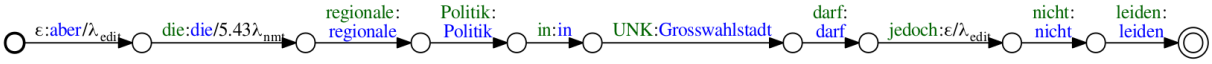
(a) Example Hiero lattice H.



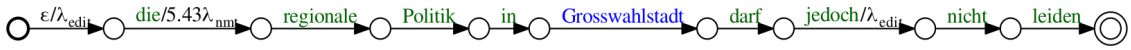
(b) Example NMT lattice N.



(c) Transducer with UNK insertion arcs: Replace$(N, \texttt{UNK}, U)$.



(d) Best path in the combined transducer C. Hiero scores are omitted in this figure.



(e) Projection of the best path: $\Pi_{UNK}(\text{ShortestPath}(C))$. The final hypothesis is *die regionale Politik in Grosswahlstadt darf jedoch nicht leiden*.

Figure 2: Combining Hiero and NMT via edit distance transducer.

Examples for H and N are shown in Fig. 2(a) and Fig. 2(b). The shortest path in H containing the string *nicht erlaubt sein sollte zu* has grammatical and stylistic flaws but is complete, whereas there is a better path in N with an UNK. Our goal is to merge these two hypotheses by using the NMT translation in N with the UNK replaced by a word from the Hiero lattice H.

1. **Adding UNK insertions**. We found that often NMT produces an isolated UNK token, even if multiple tokens are required. Therefore, we allow extending a single UNK token to a sequence of up to three UNK tokens. This is realized by replacing UNK arcs in N with the transducer U shown in Fig. 3 using OpenFST's `Replace` operation. Fig. 2(c) shows the result of the replace operation when applied to the example lattice N in Fig. 2(b). We denote this operation as follows:

$$\text{Replace}(N, \texttt{UNK}, U) \tag{2}$$

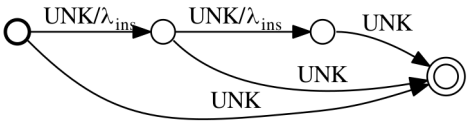


Figure 3: UNK extension transducer U.

2. **Composition with the edit distance transducer**. The next step finds the edit distances to the Hiero hypotheses as described in Sec. 2.2.

$$C := \text{Replace}(N, \texttt{UNK}, U) \circ E \circ H \tag{3}$$

3. **Shortest path**. The above operation generates very large lattices, and dumping all of them is not feasible. We could use disambiguation (Iglesias et al., 2015; Mohri and Riley, 2015) on the combined transducer C to find the best alignment for each unique NMT hypothesis. However, we only need the single shortest path in order to generate the combined translation.

$$\text{ShortestPath}(C) \tag{4}$$

4. **Projection**. A complete path in the transducer C has an NMT hypothesis on the input labels (marked green in Fig. 2(d)) and a Hiero hypothesis on the output labels (marked blue in Fig. 2(d)). Therefore, we can generate different translations from the best path in C. If we project the input labels on the output labels with OpenFST's `Project`, we obtain a hypothesis $\widehat{t}_{NMT}$ in the NMT lattice N.

$$\widehat{t}_{NMT} = \Pi_1(\text{ShortestPath}(C)) \tag{5}$$

However, $\widehat{t}_{NMT}$ still contains UNKs. If we project on the input labels, we end up with the aligned Hiero hypothesis without UNKs (blue labels in Fig. 2(d))

$$\widehat{t}_{Hiero} = \Pi_2(\text{ShortestPath}(C)) \quad (6)$$

but we do not use the NMT translation directly. Therefore, we introduce a new projection function Π_{UNK} which switches between preserving symbols on the input and output tapes: if the input label on an arc is UNK, we write the output label over the input label. Otherwise, we write the input label over the output label. This is equivalent to projecting the output labels to the input labels only if the input label is UNK, and then projecting the input labels to the output labels. As shown in Fig. 2(e), we obtain the NMT hypothesis, but the UNK is replaced by the matching word *Grosswahlstadt* from the Hiero lattice. Thus, the final combined translation is described by the following term:

$$\widehat{t}_{comb} = \Pi_{UNK}(\text{ShortestPath}(C)) \quad (7)$$

In general, the final hypothesis $\widehat{t}_{comb}$ is a mix of an NMT and a Hiero hypothesis. We do not search for $\widehat{t}_{comb}$ directly but for pairs of NMT and Hiero translations which optimize the individual model scores as well as the distance between them. Stated more formally, the shortest path in C yields a pair $(\widehat{t}_{NMT}, \widehat{t}_{Hiero})$ for which holds

$$\widehat{t}_{NMT}, \widehat{t}_{Hiero} = \underset{(t_N,t_H)\in N\times H}{\text{argmin}} \Big(d_{edit}(t_N, t_H) + \lambda_{nmt} \cdot S_N(t_N|s) + \lambda_{hiero} \cdot S_H(t_H|s) \Big) \quad (8)$$

where $d_{edit}(t_N, t_H)$ is the modified edit distance between t_N and t_H (according E and U), and $S_N(t_N|s)$ and $S_H(t_H|s)$ are the scores NMT and Hiero assign to the translations given source sentence s. If we interpret these scores as negative log-likelihoods, we arrive at a probabilistic interpretation of Eq. 8.

$$\widehat{t}_{NMT}, \widehat{t}_{Hiero} = \underset{(t_N,t_H)\in N\times H}{\text{argmax}} \Big(\mathrm{e}^{-d_{edit}(t_N,t_H)} \cdot P(t_N, t_H|s) \Big) \quad (9)$$

with (assuming independence)

$$P(t_N, t_H|s) := P_N(t_N|s)^{\lambda_{nmt}} \cdot P_H(t_H|s)^{\lambda_{hiero}}.$$

Eq. 9 suggests that we maximize the product of two quantities – the similarity between Hiero and NMT hypotheses and their joint probability. The FST operations allow to optimize over the set $N \times H$ efficiently. Note that the NMT lattice N is rather small in our case ($|N| \leq 20$) due to the small beam size used in NMT decoding. This makes it possible to solve Eq. 8 almost always without pruning [2].

3 Experimental Setup

The parallel training data includes *Europarl v7*, *Common Crawl*, and *News Commentary v10*. Sentence pairs with sentences longer than 80 words or length ratios exceeding 2.4:1 were deleted, as were *Common Crawl* sentences from other languages (Shuyo, 2010). We use *news-test2014* (the filtered version) as a development set, and keep *news-test2015* and *news-test2016* as test sets.

The NMT systems are built using the Blocks framework (van Merriënboer et al., 2015) based on the Theano library (Bastien et al., 2012) with the network architecture and hyper-parameters as in (Bahdanau et al., 2015): the encoder and decoder networks consist of 1000 gated recurrent units (Cho et al., 2014). The decoder uses a single maxout (Goodfellow et al., 2013) output layer with the feed-forward attention model described in (Bahdanau et al., 2015). In our final ensemble, we use 8 independently trained NMT systems with vocabulary sizes between 30,000 and 60,000.

Rules for our En-De Hiero system were extracted as described in (de Gispert et al., 2010). A 5-gram language model for the Hiero system was trained on WMT16 parallel and monolingual data (Heafield et al., 2013).

We apply gentle post-processing to the German output for fixing small number and currency formatting issues. The English source sentences in the training corpus are lower-cased. During decoding, we lower case only in-vocabulary words, and pass through OOVs with correct casing. We apply a simple heuristic for recognizing surnames to avoid literal translation of them into German[3].

[2] We limit the Hiero lattices to a maximum of 100,000 nodes with OpenFST's `Prune` to remove the worst outliers.

[3] We mark a word as surname if it has occurred after a first name, is on a census list of known surnames, and is written with a capitalized initial letter.

Setup		news-test2014	news-test2015	news-test2016
Best in competition[4]		20.6	25.2	34.8
Hiero baseline		18.9	21.2	26.0
Single NMT	Pure NMT	17.5	19.6	23.2
	SGNMT (lattice rescoring)	21.2	23.5	28.7
	Edit distance transducer based combination	21.7	24.1	28.6
Ensemble NMT	Pure NMT	19.4	21.7	25.4
	SGNMT (lattice rescoring)	21.9	24.6	29.7
	Edit distance transducer based combination	**22.9**	**25.7**	**31.3**

Table 1: English-German lower-cased BLEU scores calculated with Moses `mteval-v13a.pl`.

Method	BLEU
NMT baseline: ShortestPath(N)	25.4
Hiero baseline: ShortestPath(H)	26.4
NMT hypothesis used for combination: $\widehat{t}_{NMT}$	26.7
Hiero hypothesis used for combination: $\widehat{t}_{Hiero}$	30.4
Combined translation: $\widehat{t}_{comb}$	31.3

Table 2: Projection methods on *news-test2016* with NMT 8-ensemble.

4 Results

Tab. 1 reports performance on *news-test2014*, *news-test2015*, and *news-test2016*[5]. Similarly to previous work (Stahlberg et al., 2016), we observe that rescoring Hiero lattices with NMT (SGNMT) outperforms both NMT and Hiero baselines significantly on all test sets. For SGNMT, we see further improvements of between +0.7 BLEU (*news-test2014*) and +1.1 BLEU (*news-test2015*) by using NMT ensembles rather than single NMT. However, these gains are rather small considering the improvements from using ensembles for the (pure) NMT baseline (between +1.9 BLEU and +2.2 BLEU). Our combination scheme makes better use of the ensembles. We report 31.3 BLEU on *news-test2016*, which in the English-German WMT'16 evaluation is among the best systems (within 0.1 BLEU) which do not use back-translation (Sennrich et al., 2016a). Back-translation is a technique for making use of monolingual data in NMT training, and we expect our system could benefit from back-translation, although we leave this analysis to future work.

The combination procedure we propose is non-trivial. It is not immediately clear how the gains arise as the final scores are mixtures between edit distance costs, NMT scores, and Hiero scores. In the remainder we will try to provide some insight. Unless stated otherwise, we report investigations into the Hiero + NMT 8-system ensemble which yields the best results in Tab. 1.

First, we focus on the projection function $\Pi_{UNK}(\cdot)$ which switches between preserving the input and output label at the UNK symbol to produce the combined translation $\widehat{t}_{comb}$ (Eq. 7). As explained in Sec. 2.3, we can use OpenFST's `Project` operation to fetch the NMT and Hiero hypotheses $\widehat{t}_{NMT}$ and $\widehat{t}_{Hiero}$ which have been used to produce the combined translation (Eq. 5 and 6). Tab. 2 shows that the hypotheses that are aligned in the final transducer are often not the 1-best translations of any of the baseline systems. Remarkably, using the $\widehat{t}_{Hiero}$ translations results in 30.4 BLEU, which is a very substantial improvement over the baseline Hiero system (26.0 BLEU). Note that this BLEU score is achieved with hypotheses from the original Hiero lattice H but weighted in combination with the NMT scores and the edit distance. However, these selected paths are often given very low scores by Hiero: in only 8.6% of the sentences is the Hiero hypothesis left unchanged. If we look for $\widehat{t}_{Hiero}$ in the Hiero n-best list, we find that even very deep 20,000-best lists contain only 63.5% of the Hiero hypotheses which were selected by the combination scheme (Fig. 4). This indicates the benefit in using lattice-

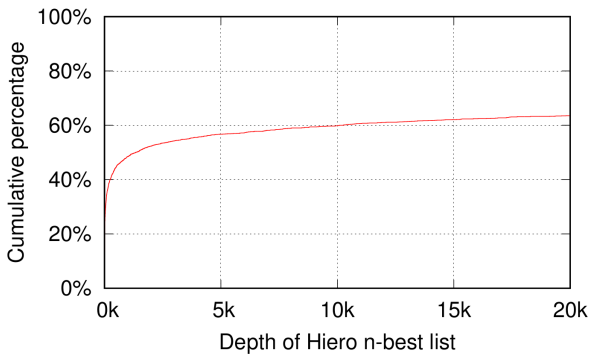


Figure 4: Percentage of $\widehat{t}_{Hiero}$ hypotheses found in the baseline Hiero n-best list.

[4] http://matrix.statmt.org/

[5] The code we used for SGNMT and ensembling is available at http://ucam-smt.github.io/sgnmt/html/.

Distance measure component	Avg. number per sentence	Percentage of affected sentences
UNK insertions (U)	0.16	12.9%
UNK→non-OOV substitutions (Type II)	1.34	55.9%
Other edit operations (Type III)	1.74	61.7%

Table 3: Breakdown of the distances measured between NMT and Hiero along the shortest path in C on *news-test2016*.

based approaches over n-best lists.

Next, we investigate the distance measure between NMT and Hiero translations, which is realized with the UNK insertion transducer U and the modified edit distance transducer E (Sec. 2.3). Tab. 3 shows that UNK insertions are relatively rare compared to the edit operations of types II and III allowed by E (Sec. 2.3). The average edit distance between NMT and Hiero disregarding UNKs on the best path (type III) is 1.74. In 61.7% of the cases the input and output labels differ not only at UNK – i.e. in only 38.3% of the sentences do we have an exact match between NMT and Hiero. We note that UNK is often replaced with an NMT in-vocabulary word (55.9% of the sentences). It seems that NMT often produces an UNK even if a better word is in the NMT vocabulary. This could be due to the over-representation of UNK in the NMT training corpus.

To study the effectiveness of our edit distance transducer based combination scheme in correcting NMT UNKs, we trained individual NMT systems with vocabulary sizes between 10,000 and 60,000. Tab. 4 shows that nearly one in six tokens (16.3%) produced by our pure NMT system with a vocabulary size of 30,000 are UNKs. Increasing the NMT vocabulary to 50k or 60k does improve pure NMT very significantly, but results show that these improvements are already captured by the combination scheme with Hiero. As in the literature, we see large variation in performance over individual NMT systems even with the same vocabulary size (Sennrich et al., 2016b), which could explain the small performance drop when increasing the vocabulary size from 50k to 60k.

One important practical issue for system building is the number of systems to be ensembled as training each individual NMT system takes a significant amount of time. Fig. 5 indicates that even for 8-ensembles the gains for pure NMT do not seem to saturate. The combination with Hiero via edit distance transducer also greatly benefits from using ensembles, but most of the gains are gotten with fewer systems.

Vocabulary size	Pure NMT		NMT+Hiero
	BLEU	# of UNKs	BLEU
10,000	18.9	18.0%	28.1
30,000	21.6	16.3%	28.8
50,000	23.2	9.1%	28.6
60,000	22.9	9.9%	28.5

Table 4: BLEU scores on *news-test2016* for different vocabulary sizes (single NMT). Each individual NMT system is combined with Hiero as described in Sec. 2.3.

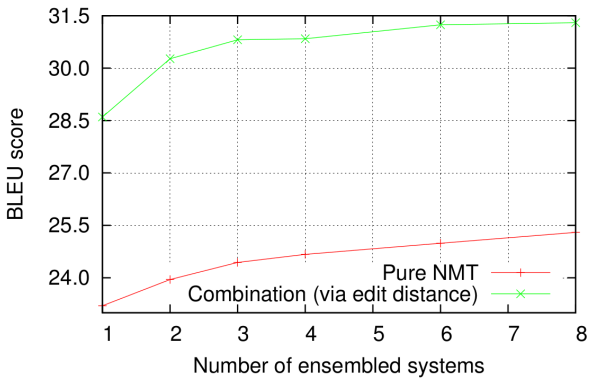


Figure 5: BLEU score over the number of systems in the ensemble on *news-test2016*.

5 Conclusion and Future Work

We have presented a method based on the edit distance that is effective in combining Hiero SMT systems with NMT ensembles. Our approach makes use of standard WFST operations, and we showed the effectiveness of the approach with a successful WMT'16 submission for English-German. In the future, we are planning to add back-translation (Sennrich et al., 2016a) and investigate the use of character- or subword-based NMT (Sennrich et al., 2016b; Chitnis and DeNero, 2015; Ling et al., 2015; Chung et al., 2016; Luong and Manning, 2016) within our combination framework.

Acknowledgements

This work was supported by the U.K. Engineering and Physical Sciences Research Council (EPSRC grant EP/L027623/1).

References

Cyril Allauzen, Michael Riley, Johan Schalkwyk, Wojciech Skut, and Mehryar Mohri. 2007. OpenFst: A general and efficient weighted finite-state transducer library. In *Implementation and Application of Automata*, pages 11–23. Springer.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, James Bergstra, Ian Goodfellow, Arnaud Bergeron, Nicolas Bouchard, David Warde-Farley, and Yoshua Bengio. 2012. Theano: new features and speed improvements. In *NIPS*.

David Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2):201–228.

Rohan Chitnis and John DeNero. 2015. Variable-length word encodings for neural translation models. In *EMNLP*, pages 2088–2093.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *EMNLP*.

Junyoung Chung, Kyunghyun Cho, and Yoshua Bengio. 2016. A character-level decoder without explicit segmentation for neural machine translation. In *ACL*.

Adrià de Gispert, Sami Virpioja, Mikko Kurimo, and William Byrne. 2009. Minimum Bayes risk combination of translation hypotheses from alternative morphological decompositions. In *NAACL*, pages 73–76.

Adrià de Gispert, Gonzalo Iglesias, Graeme Blackwood, Eduardo R Banga, and William Byrne. 2010. Hierarchical phrase-based translation with weighted finite-state transducers and shallow-n grammars. *Computational Linguistics*, 36(3):505–533.

Jonathan G Fiscus. 1997. A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER). In *ASRU*, pages 347–354.

Robert Frederking, Sergei Nirenburg, David Farwell, Stephen Helmreich, Eduard Hovy, Kevin Knight, Stephen Beale, Constantine Domashnev, Donalee Attardo, Dean Grannes, et al. 1994. Integrating translations from multiple sources within the Pangloss Mark III machine translation system. In *AMTA*, pages 73–80.

Vaibhava Goel and William J Byrne. 2000. Minimum Bayes-risk automatic speech recognition. *Computer Speech & Language*, 14(2):115–135.

Ian Goodfellow, David Warde-farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. 2013. Maxout networks. In *ICML*, pages 1319–1327.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. 2013. Scalable modified Kneser-Ney language model estimation. In *ACL*, pages 690–696.

Gonzalo Iglesias, Adrià de Gispert, and William Byrne. 2015. Transducer disambiguation with sparse topological features. In *EMNLP 2015*, pages 2275–2280.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *EMNLP*, page 413.

Shankar Kumar and William Byrne. 2004. Minimum Bayes-risk decoding for statistical machine translation. Technical report, DTIC Document.

Wang Ling, Isabel Trancoso, Chris Dyer, and Alan W Black. 2015. Character-based neural machine translation. *arXiv preprint arXiv:1511.04586*.

Yang Liu, Haitao Mi, Yang Feng, and Qun Liu. 2009. Joint decoding with multiple translation models. In *ACL*, pages 576–584.

Minh-Thang Luong and Christopher D Manning. 2016. Achieving open vocabulary neural machine translation with hybrid word-character models. In *ACL*.

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

Mehryar Mohri and Michael D Riley. 2015. On the disambiguation of weighted automata. In *Implementation and Application of Automata*, pages 263–278. Springer.

Mehryar Mohri. 2003. Edit-distance of weighted automata: General definitions and algorithms. *International Journal of Foundations of Computer Science*, 14(06):957–982.

Mehryar Mohri. 2004. Weighted finite-state transducer algorithms. An overview. In *Formal Languages and Applications*, pages 551–563. Springer.

Graham Neubig, Makoto Morishita, and Satoshi Nakamura. 2015. Neural reranking improves subjective quality of machine translation: NAIST at WAT2015. *arXiv preprint arXiv:1510.05203*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *ACL*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *ACL*.

Nakatani Shuyo. 2010. Language detection library for Java. `http://code.google.com/p/language-detection/`. [Online; accessed 1-June-2016].

Felix Stahlberg, Eva Hasler, Aurelien Waite, and Bill Byrne. 2016. Syntactically guided neural machine translation. In *ACL*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, pages 3104–3112.

Roy W Tromble, Shankar Kumar, Franz Och, and Wolfgang Macherey. 2008. Lattice minimum Bayes-risk decoding for statistical machine translation. In *EMNLP*, pages 620–629.

Bart van Merriënboer, Dzmitry Bahdanau, Vincent Dumoulin, Dmitriy Serdyuk, David Warde-Farley, Jan Chorowski, and Yoshua Bengio. 2015. Blocks and fuel: Frameworks for deep learning. *arXiv preprint arXiv:1506.00619*.