Learning to Stop in Structured Prediction for Neural Machine Translation

Mingbo Ma [*§†] **Renjie Zheng** [*†] **Liang Huang** [†,‡]
[†]School of EECS, Oregon State University, Corvallis, OR
[‡]Baidu Research, Sunnyvale, CA
{cosmmb, zrenj11, liang.huang.sh}@gmail.com

Abstract

Beam search optimization (Wiseman and Rush, 2016) resolves many issues in neural machine translation. However, this method lacks principled stopping criteria and does not learn how to stop during training, and the model naturally prefers *longer* hypotheses during the testing time in practice since they use the raw score instead of the probability-based score. We propose a novel ranking method which enables an optimal beam search stopping criteria. We further introduce a structured prediction loss function which penalizes suboptimal finished candidates produced by beam search during training. Experiments of neural machine translation on both synthetic data and real languages (German→English and Chinese→English) demonstrate our proposed methods lead to better length and BLEU score.

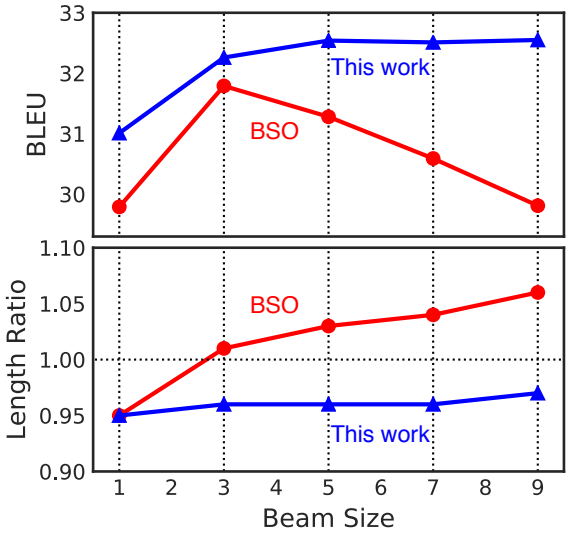

Figure 1: The BLEU score of BSO decreases after beam size 3 as results of increasing length ratio[1] in German→English translation. Our model gets higher BLEU with larger beam.

1 Introduction

Sequence-to-sequence (seq2seq) models based on RNNs (Sutskever et al., 2014; Bahdanau et al., 2014), CNNs (Gehring et al., 2017) and self-attention (Vaswani et al., 2017) have achieved great successes in Neural Machine Translation (NMT). The above family of models encode the source sentence and predict the next word in an autoregressive fashion at each decoding time step. The classical "cross-entropy" training objective of seq2seq models is to maximize the likelihood of each word in the translation reference given the source sentence and all previous words in that reference. This word-level loss ensures efficient and scalable training of seq2seq models.

However, this word-level training objective suffers from a few crucial limitations, namely the *label bias* (Murray and Chiang, 2018), the *exposure bias*, and the *loss-evaluation mismatch* (Lafferty et al., 2001; Bengio et al., 2015a; Venkatraman et al., 2015). In addition, more importantly, at decoding time, *beam search* is universally adopted to improve the search quality, while training is fundamentally local and greedy. Several researchers have proposed different approaches to alleviate above problems, such as reinforcement learning-based methods (Ranzato et al., 2016; Rennie et al., 2017; Zheng et al., 2018b), training with alternative references (Shen et al., 2016; Zheng et al., 2018a). Recently, Wiseman and Rush (2016) attempt to address these issues with a structured training method, Beam Search Optimization (BSO). While BSO outperforms other proposed methods on German-to-English translation, it also brings a different set of problems as partially discussed in (Ma, 2018) which we present with details below.

[*] Equal contribution
[§] Current address: Baidu Research, Sunnyvale, CA.

[1]There are two types of "length ratios" in this paper: (a)

BSO relies on unnormalized raw scores instead of locally-normalized probabilities to get rid of the label bias problem. However, since the raw score can be either positive or negative, the optimal stopping criteria (Huang et al., 2017) no longer holds, e.g., one extra decoding step would increase the entire unfinished hypothesis's model score when we have positive word score. This leads to two consequences: we do not know when to stop the beam search and it could return overlength translations (Fig. 1) or underlength translations (Fig. 3) in practice. As shown in Fig. 1, the BLEU score of BSO drops significantly when beam size gets larger as a result of overlong translations (as evidenced by length ratios larger than 1). Furthermore, BSO performs poorly (shown in Section 4) on hard translation pairs, e.g., Chinese→English (Zh→En) translation, when the target / source ratio is more diverse (Table 1).

To overcome the above issues, we propose to use the sigmoid function instead of the raw score at each time step to rank candidates. In this way, the model still has probability properties to hold optimal stopping criteria without label bias effects. Moreover, we also encourage the model to generate the hypothesis which is more similar to gold reference in length. Compared with length reward-based methods (Huang et al., 2017; Yang et al., 2018), our model does not need to tune the predicted length and per-word reward. Experiments on both synthetic and real language translations (De→En and Zh→En) demonstrate significant improvements in BLEU score over strong baselines and other methods.

2 Preliminaries: NMT and BSO

Here we briefly review the conventional NMT and BSO (Wiseman and Rush, 2016) to set up the notations. For simplicity, we choose to use RNN-based model but our methods can be easily applied to other designs of seq2seq model as well.

Regardless of the particular design of different seq2seq models, generally speaking, the decoder always has the following form:

$$p(\mathbf{y} \mid \mathbf{x}) = \prod_{t=1}^{|\mathbf{y}|} p(y_t \mid \mathbf{x}, \mathbf{y}_{<t}) \qquad (1)$$

where $\mathbf{x} \in R^{N \times D}$ represents the D-dimension hidden states from encoder with N words and $\mathbf{y}_{<t}$ denotes the gold prefix $(y_1, ..., y_{(t-1)})$ before t. The conventional NMT model is locally trained to maximize the above probability.

Instead of maximizing each gold word's probability, BSO tries to promote the non-probabilistic scores of gold sequence within a certain beam size b. BSO removes the softmax layer and directly uses the raw score after hidden-to-vocabulary layer, and the non-probabilistic scoring function $f_{\mathbf{x}}(y_t \mid \mathbf{y}_{<t})$ represents the score of word y_t given gold prefix $\mathbf{y}_{<t}$ and $\mathbf{x}$. Similarly, $f_{\mathbf{x}}(\hat{y}_t^b \mid \hat{\mathbf{y}}_{<t}^b)$ is the b^{th} sequence with beam size b at time step t. Then, we have the following loss function to penalize the b^{th} candidate and promote gold sequence:

$$\mathbb{L} = \sum_{t=1}^{|\mathbf{y}|} \Delta(\hat{\mathbf{y}}_{\leq t}^b)(1 + f_{\mathbf{x}}(\hat{y}_t^b \mid \hat{\mathbf{y}}_{<t}^b) - f_{\mathbf{x}}(y_t \mid \mathbf{y}_{<t}))^+ \qquad (2)$$

where $\Delta(\hat{\mathbf{y}}_{\leq t}^b)$ is defined as $(1 - \text{BLEU}(\hat{\mathbf{y}}_{\leq t}^b, \mathbf{y}_{\leq t}))$ which scales the loss according to BLEU score between gold and b^{th} hypothesis in the beam. The notation $(\cdot)^+$ represents a max function between any value and 0, i.e., $z^+ = max(0, z)$.

When Eq. 1 equals to 0 at time step t, then the gold sequence's score is higher than the last hypothesis in the beam by 1, and a positive number otherwise. Finally, at the end of beam search ($t = |\mathbf{y}|$), BSO requires the score of $\mathbf{y}$ exceed the score of the highest incorrect hypothesis by 1.

Note that the above non-probabilistic score function $f_{\mathbf{x}}(\cdot)$ is not bounded as probabilistic score in conventional NMT. In practice, when we have positive word score, then the unfinished candidates always get higher model scores with one extra decoding step and the optimal stopping criteria [2] (Huang et al., 2017) is no longer hold. BSO implements a similar "shrinking beam" strategy which duplicates top unfinished candidate to replace finished hypotheses and terminates the beam search when there are only `</eos>` in the beam. Non-probabilistic score function works well in parsing and Statical MT where we know when to stop beam search. However, in the NMT scenario, without optimal stopping criteria, we don't know when to stop beam search.

3 Learning to Stop

We propose two major improvements to BSO.

target to reference ratio ($|\mathbf{y}|/|\mathbf{y}^*|$), which is used in BLEU, and (b) target to source ratio ($|\mathbf{y}|/|\mathbf{x}|$). By default, the term "length ratio" in this paper refers to the former.

[2] Beam search stops when the score of the top unfinished hypothesis is lower than any finished hypothesis, or the `</eos>` is the highest score candidate in the beam.

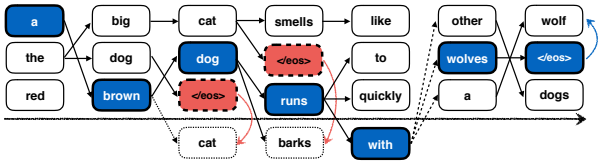

Figure 2: Training illustration with beam size $b = 3$ and gold reference "a brown dog runs with the wolves". The gold reference is highlighted in blue solid boxes. We penalize the under-length translation (short) hypotheses by expelling out the early </eos> out of beam (red dashed boxes). The beam search restarts with gold when gold falls off the beam (at step 5).

3.1 Sigmoid Scoring Function

As mentioned in Section 2, BSO relies on raw score function to eliminate label bias effects. However, without using locally-normalized score does not mean that we should stop using the probabilistic value function. Similar with multi-label classification in (Ma et al., 2017), instead of using locally normalized softmax-based score and non-probabilistic raw scores, we propose to use another form of probabilistic scoring function, sigmoid function, which is defined as follows:

$$g_{\mathbf{x}}(y_t \mid \mathbf{y}_{<t}) = (1 + e^{w \cdot f_{\mathbf{x}}(y_t|\mathbf{y}_{<t})})^{-1} \quad (3)$$

where w is a trainable scalar parameter which shifts the return value of $f_{\mathbf{x}}(y_t \mid \mathbf{y}_{<t})$ into a non-saturated value region of sigmoid function. Eq. 3 measures the probability of each word independently which is different from locally-normalized softmax function. Similar to the scenario in multi-label classification, $g_{\mathbf{x}}(y_t \mid \mathbf{y}_{<t})$ only promotes the words which are preferred by gold reference and does not degrade other words. Eq. 3 enables the model to keep the probability nature of scoring function without introducing label bias effects. After the model regain probability-based scoring function, the optimal stopping criteria can be used in testing time decoding.

3.2 Early Stopping Penalties

Similar to Eq. 1, testing time decoder multiplies the new word's probabilistic score with prefix's score when there is a new word appends to an unfinished hypothesis. Though the new word's probabilistic score is upper bounded by 1, in practice, the score usually far less than one. As described in (Huang et al., 2017; Yang et al., 2018), decoder always prefers short sentence when we use the probabilistic score function.

To overcome the above so-called "*beam search curse*", we propose to penalize early-stopped hypothesis within the beam during training. The procedure during training is illustrated in Fig. 2.

Data	Split	$\overline{\lvert x\rvert}$	$\sigma(\lvert x\rvert)$	$\overline{(\frac{\lvert y\rvert}{\lvert x\rvert})}$	$\sigma(\frac{\lvert y\rvert}{\lvert x\rvert})$	# sents
Synthetic	Train	9.47	5.45	3.0	0.52	5K
	Valid	9.54	5.42	3.0	0.53	1K
	Test	9.51	5.49	3.0	0.52	1K
De→En	Train	17.53	9.93	1.07	0.16	153K
	Valid	17.55	9.97	1.07	0.16	7K
	Test*	18.89	12.82	1.06	0.16	6.5K
Zh→En	Train	23.21	13.44	1.30	0.33	1M
	Valid	29.53	16.62	1.34	0.22	0.6K
	Test	26.53	15.99	1.4	0.24	0.7K

Table 1: Dataset statistics of source sentence length and the ratio between target and source sentences. σ is standard deviation. *shows statistics of cleaned test set.

Different from BSO, to penalize the underlength finished translation hypotheses, we include additional violations when there is an </eos> within the beam before the gold reference finishes and we force the score of that </eos> lower than the $b+1$ candidate by a margin. This underlength translation violation is formally defined as follows:

$$\mathbb{L}^{\text{s}} = \sum_{t=1}^{|\mathbf{y}|} \sum_{j=1}^{b} \mathbb{1}(\hat{y}_t^j = \text{</eos>}) \cdot Q(\hat{y}_t^j, \hat{y}_t^{b+1}) \ , \quad (4)$$
$$Q(\hat{y}_t^j, \hat{y}_t^{b+1}) = (g_{\mathbf{x}}(\hat{y}_t^j \mid \hat{\mathbf{y}}_{<t}^j) - g_{\mathbf{x}}(\hat{y}_t^{b+1} \mid \hat{\mathbf{y}}_{<t}^{b+1}))^+$$

where notation $\mathbb{1}$ is identification function which only equals to 1 when i^{th} candidate in beam $\hat{y}_t^j$ is </eos>, e.g. in Fig. 2. We only have non-zero loss when the model score of underlength translation candidates are greater than the $b+1$ candidate by a margin. In this way, we penalize all the short hypotheses during training time. Note that during both training and testing time, the decoder stops beam search when it satisfies the optimal stopping criteria (Huang et al., 2017). Therefore, we do not need to penalize the overlength translations since we have already promoted the gold reference to the top of the beam at time step $|y|$ during training.

4 Experiments

We showcase the performance comparisons over three different datasets. We implement seq2seq model, BSO and our proposed model based on PyTorch-based OpenNMT (Klein et al., 2017). We use a two-layer bidirectional LSTM as the encoder and a two layer LSTM as the decoder. We train Seq2seq model for 20 epochs to minimize perplexity on the training dataset, with a batch size of 64, word embedding size of 512, the learning rate of 0.1, learning rate decay of 0.5 and dropout rate of 0.2. Following Wiseman and Rush (2016),

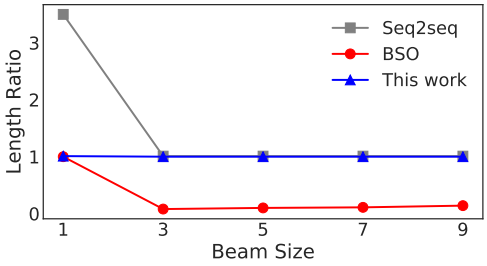

Figure 3: Length ratio on synthetic test dataset.

we then train BSO and our model based on the previous Seq2seq model with the learning rate of 0.01 and learning rate decay of 0.75, batch size of 40. Note that our pretrained model is softmax-based, and we only replace the softmax layer with the sigmoid layer for later training for simplicity. The performance will have another boost when our pretrained model is sigmoid-based. We use Adagrad (Duchi et al., 2011) as the optimizer.

In Zh→En task, we employ BPE (Sennrich et al., 2015) which reduces the source and target language vocabulary sizes to 18k and 10k. Following BSO, we set the decoding beam size smaller than the training beam size by 1.

4.1 Synthetic Task

Table 1 shows the statistics of source sentence length and the ratio between target and source sentences. The synthetic dataset is a simple translation task which generates target sentences from this grammar: $\{a \to x, b \to x\ x, c \to x\ x\ x, d \to x\ x\ x\ x, e \to x\ x\ x\ x\ x\}$. For example:

1. source sentence $[b\, c\, a]$ will generate the target sentence $[x\ x\ x\ x\ x]$ (2 x from b, 3 x from c and 1 x from a).
2. source sentence $[a, b, c, d, e]$ will be translated into $[x\ x\ x\ x\ x\ x\ x\ x\ x\ x\ x\ x\ x\ x\ x]$ in target side (1 x from a, 2 x from b, 3 x from c, 4 x from d and 5 x from e).

This dataset is designed to evaluate the length prediction ability of different models. Fig. 3 shows the length ratio of different models on the test set. Only our model can predict target sentence length correctly with all beam sizes which shows a better ability to learn target length.

4.2 De→En Translation

The De→En dataset is previously used in BSO and MIXER (Ranzato et al., 2016), which is from IWSLT 2014 machine translation evaluation campaign (Cettolo et al., 2014) [3].

[3]The test set of De→En involves some mismatched source-reference pairs. We have cleaned this test set and report the statistics based on the cleaned version.

Decode	Seq2seq†		Train	BSO†		This work	
Beam	BLEU	Len.	Beam	BLEU	Len.	BLEU	Len.
1	30.65	1.00	2	29.79	0.95	31.01	0.95
3	31.38	0.97	4	31.79	1.01	32.26	0.96
5	31.38	0.97	6	31.28	1.03	32.54	0.96
7	31.42	0.96	8	30.59	1.04	32.51	0.96
9	31.44	0.96	10	29.81	1.06	32.55	0.97

Table 2: BLEU and length ratio on the De→En validation set. †indicates our own implementation.

Model	BLEU	Len.
This work (full model)	32.54	0.96
This work w/ softmax	32.29	0.98
This work w/o scale augment	31.97	0.95
This work w/o early stopping loss	31.19	0.93

Table 3: Ablation study on the De→En validation set with training beam size $b = 6$

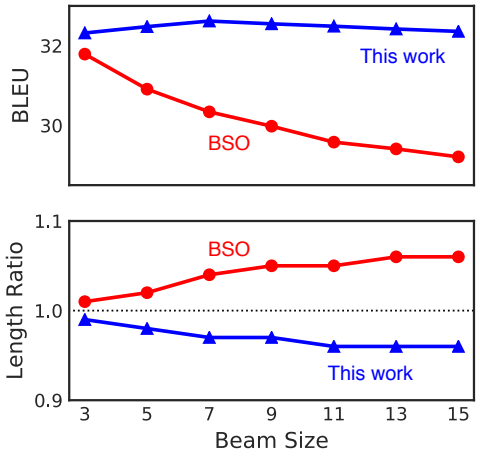

Figure 4: BLEU and length ratio of models with training beam size $b = 6$ and decode with different beam size on De→En dataset.

Table 2 shows the BLEU score and length ratio of different models on dev-set. Similar to seq2seq, our proposed model achieves better BLEU score with larger beam size and outperforms the best BSO $b = 4$ model with 0.76 BLEU. The ablation study in Table 3 shows that the model produces shorter sentence without scale augment (term $\Delta(\hat{y}^b_{\leq t})$ in Eq. 2) and early stopping loss. The model also performs worse when replacing softmax to sigmoid because of the label bias problem. Fig. 4 shows BLEU score and length ratio of BSO and our models trained with beam size $b = 6$ with different decoding beam size. Compared with BSO, whose BLEU score degrades dramatically when increasing beam size, our model performs much more stable. Moreover, BSO achieves much better BLEU score with decoding beam $b = 3$ while trained with $b = 6$ because of a better

Model	Original Test Set		Cleaned Test Set	
	BLEU	Len.	BLEU	Len.
BSO‡	26.35	-	-	
DAD‡	22.40	-	-	
MIXER‡	21.83	-	-	
Seq2seq†	29.54	0.97	30.08	0.97
BSO†	29.63	1.02	30.08	1.02
This work	30.29	0.98	30.85	0.98

Table 4: BLEU and length ratio on the De→En test set. †indicates our own implementation. ‡results from (Wiseman and Rush, 2016).

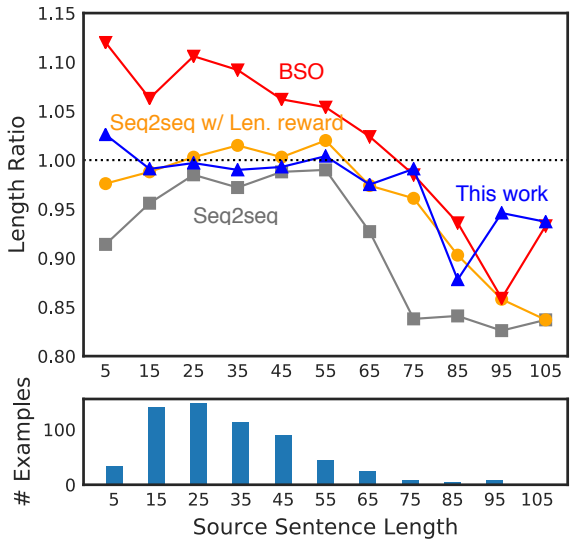

Figure 5: Length ratio of examples on Zh→En dev-set with different source sentence length.

length ratio, this is inconsistent with their claim that decoding beam size should smaller than training beam size by 1.

Table 4 shows better accuracy of our proposed model than not only published test results of BSO (Wiseman and Rush, 2016), DAD (Bengio et al., 2015b) and MIXER (Ranzato et al., 2016), but also our implemented seq2seq and BSO model.

4.3 Zh→En Translation

Model	Train Beam	Decode Beam	BLEU	Len.
Seq2Seq†	-	7	37.74	0.96
w/ Len. reward†	-	7	38.28	0.99
BSO†	4	3	36.91	1.03
BSO†	8	7	35.57	1.07
This work	4	3	38.41	1.00
This work	8	7	39.51	1.00

Table 5: BLEU and length ratio of models on Zh→En validation set. †indicates our own implementation.

We also perform experiments on NIST Zh→En translation dataset. We use the NIST 06 and 08 dataset with 4 references as the validation and test set respectively. Table 1 shows that the characteristic of Zh→En translation is very different from De→En in source length and variance in target/source length ratio.

Model	BLEU	Len.
Seq2Seq†	34.19	0.95
w/ Len. reward†	34.60	0.99
BSO†	31.78	1.04
This work	35.40	0.99

Table 6: BLEU and length ratio of models on Zh→En test set. †indicates our own implementation.

We compare our model with seq2seq, BSO and seq2seq with length reward (Huang et al., 2017) which involves hyper-parameter to solve neural model's tendency for shorter hypotheses (our proposed method does not require tuning of hyper-parameter). Fig. 5 shows that BSO prefers over-length hypotheses in short source sentences and underlength hypotheses when the source sentences are long. This phenomenon degrades the BLEU score in dev-set from Table 5. Our proposed model comparatively achieves better length ratio on almost all source sentence length in dev-set.

5 Future Works and Conclusions

Our proposed methods are general techniques which also can be applied to the Transformer (Vaswani et al., 2017). As part of our future works, we plan to adapt our techniques to the Transformer to further evaluate our model's performance.

There are some scenarios that decoding time beam search is not applicable, such as the simultaneous translation system proposed by Ma et al. (2018) which does not allow for adjusting the committed words, the training time beam search still will be helpful to the greedy decoding performance. We plan to further investigate the performance of testing time greedy decoding with beam search optimization during training.

We propose two modifications to BSO to provide better scoring function and under-translation penalties, which improves the accuracy in De-En and Zh-En by 0.8 and 3.7 in BLEU respectively.

Acknowledgments

This work was supported in part by DARPA grant N66001-17-2-4030 (M. M.), and NSF grants IIS-1656051 and IIS-1817231 (R. Z.).

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015a. Scheduled sampling for sequence prediction with recurrent neural networks. In *NIPS*.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015b. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*, pages 1171–1179.

Mauro Cettolo, Jan Niehues, Sebastian Stüker, Luisa Bentivogli, and Marcello Federico. 2014. Report on the 11th iwslt evaluation campaign, iwslt 2014. In *Proceedings of the International Workshop on Spoken Language Translation, Hanoi, Vietnam*.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. 2017. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*.

Liang Huang, Kai Zhao, and Mingbo Ma. 2017. When to finish? optimal beam search for neural text generation (modulo beam size). In *EMNLP*.

G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. M. Rush. 2017. OpenNMT: Open-Source Toolkit for Neural Machine Translation. *ArXiv e-prints*.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML*.

Mingbo Ma. 2018. Structured neural models for natural language processing. In *Ph.D thesis, Oregon State University*.

Mingbo Ma, Liang Huang, Bing Xiang, and Bowen Zhou. 2017. Group sparse cnns for question classification with answer sets. In *ACL*.

Mingbo Ma, Liang Huang, Hao Xiong, Kaibo Liu, Chuanqiang Zhang, Zhongjun He, Hairong Liu, Xing Li, and Haifeng Wang. 2018. Stacl: Simultaneous translation with integrated anticipation and controllable latency. *ArXiv*, abs/1810.08398.

Kenton Murray and David Chiang. 2018. Correcting length bias in neural machine translation. *WMT 2018*, page 212.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2016. Sequence level training with recurrent neural networks. *ICLR*.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7008–7024.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Minimum risk training for neural machine translation. In *Proceedings of 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.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30*.

Arun Venkatraman, Martial Hebert, and J. Andrew Bagnell. 2015. Improving multi-step prediction of learned time series models. In *AAAI*.

Sam Wiseman and Alexander M Rush. 2016. Sequence-to-sequence learning as beam-search optimization. In *Proceedings of EMNLP*.

Yilin Yang, Liang Huang, and Mingbo Ma. 2018. Breaking the beam search curse: A study of (re-) scoring methods and stopping criteria for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Renjie Zheng, Mingbo Ma, and Liang Huang. 2018a. Multi-reference training with pseudo-references for neural translation and text generation. In *EMNLP*.

Renjie Zheng, Yilin Yang, Mingbo Ma, and Liang Huang. 2018b. Ensemble sequence level training for multimodal mt: Osu-baidu wmt18 multimodal machine translation system report. In *WMT*.