On NMT Search Errors and Model Errors: Cat Got Your Tongue?

Felix Stahlberg[*] and **Bill Byrne**
University of Cambridge
Department of Engineering
Trumpington St, Cambridge CB2 1PZ, UK
{fs439,wjb31}@cam.ac.uk

Abstract

We report on search errors and model errors in neural machine translation (NMT). We present an exact inference procedure for neural sequence models based on a combination of beam search and depth-first search. We use our exact search to find the global best model scores under a Transformer base model for the entire WMT15 English-German test set. Surprisingly, beam search fails to find these global best model scores in most cases, even with a very large beam size of 100. For more than 50% of the sentences, the model in fact assigns its global best score to the empty translation, revealing a massive failure of neural models in properly accounting for adequacy. We show by constraining search with a minimum translation length that at the root of the problem of empty translations lies an inherent bias towards shorter translations. We conclude that vanilla NMT in its current form requires just the right amount of beam search errors, which, from a modelling perspective, is a highly unsatisfactory conclusion indeed, as the model often prefers an empty translation.

1 Introduction

Neural machine translation (Kalchbrenner and Blunsom, 2013; Sutskever et al., 2014; Bahdanau et al., 2015, NMT) assigns the probability $P(\mathbf{y}|\mathbf{x})$ of a translation $\mathbf{y} = y_1^J \in \mathcal{T}^J$ of length J over the target language vocabulary $\mathcal{T}$ for a source sentence $\mathbf{x} \in \mathcal{S}^I$ of length I over the source language vocabulary $\mathcal{S}$ via a left-to-right factorization using the chain rule:

$$\log P(\mathbf{y}|\mathbf{x}) = \sum_{j=1}^{J} \log P(y_j|y_1^{j-1}, \mathbf{x}). \quad (1)$$

The task of finding the most likely translation $\hat{\mathbf{y}} \in \mathcal{T}^*$ for a given source sentence $\mathbf{x}$ is known as the

---

[*]Now at Google.

decoding or *inference* problem:

$$\hat{\mathbf{y}} = \underset{\mathbf{y} \in \mathcal{T}^*}{\arg\max} \, P(\mathbf{y}|\mathbf{x}). \quad (2)$$

The NMT search space is vast as it grows exponentially with the sequence length. For example, for a common vocabulary size of $|\mathcal{T}| = 32,000$, there are already more possible translations with 20 words or less than atoms in the observable universe ($32,000^{20} \gg 10^{82}$). Thus, complete enumeration of the search space is impossible. The size of the NMT search space is perhaps the main reason why – besides some preliminary studies (Niehues et al., 2017; Stahlberg et al., 2018b; Ott et al., 2018) – analyzing search errors in NMT has received only limited attention. To the best of our knowledge, none of the previous studies were able to quantify the number of search errors in unconstrained NMT due to the lack of an exact inference scheme that – although too slow for practical MT – guarantees to find the global best model score for analysis purposes.

In this work we propose such an exact decoding algorithm for NMT that exploits the monotonicity of NMT scores: Since the conditional log-probabilities in Eq. 1 are always negative, partial hypotheses can be safely discarded once their score drops below the log-probability of any *complete* hypothesis. Using our exact inference scheme we show that beam search does not find the global best model score for more than half of the sentences. However, these *search* errors, paradoxically, often prevent the decoder from suffering from a frequent but very serious *model* error in NMT, namely that the empty hypothesis often gets the global best model score. Our findings suggest a reassessment of the amount of model and search errors in NMT, and we hope that they will spark new efforts in improving NMT modeling capabilities, especially in terms of adequacy.

Algorithm 1 BeamSearch($\mathbf{x}, n \in \mathbb{N}_+$)

Input: $\mathbf{x}$: Source sentence, n: Beam size
1: $\mathcal{H}_{cur} \leftarrow \{(\epsilon, 0.0)\}$ {Initialize with empty translation prefix and zero score}
2: **repeat**
3:    $\mathcal{H}_{next} \leftarrow \emptyset$
4:    **for all** $(\mathbf{y}, p) \in \mathcal{H}_{cur}$ **do**
5:       **if** $y_{|\mathbf{y}|} = </\mathbf{s}>$ **then**
6:          $\mathcal{H}_{next} \leftarrow \mathcal{H}_{next} \cup \{(\mathbf{y}, p)\}$ {Hypotheses ending with $</\mathbf{s}>$ are not expanded}
7:       **else**
8:          $\mathcal{H}_{next} \leftarrow \mathcal{H}_{next} \cup \bigcup_{w \in \mathcal{T}} (\mathbf{y} \cdot w, p + \log P(w|\mathbf{x}, \mathbf{y}))$ {Add all possible continuations}
9:       **end if**
10:    **end for**
11:    $\mathcal{H}_{cur} \leftarrow \{(\mathbf{y}, p) \in \mathcal{H}_{next} : |\{(\mathbf{y}', p') \in \mathcal{H}_{next} : p' > p\}| < n\}$ {Select n-best}
12:    $(\tilde{\mathbf{y}}, \tilde{p}) \leftarrow \arg\max_{(\mathbf{y}, p) \in \mathcal{H}_{cur}} p$
13: **until** $\tilde{y}_{|\tilde{\mathbf{y}}|} = </\mathbf{s}>$
14: **return** $\tilde{\mathbf{y}}$

Algorithm 2 DFS($\mathbf{x}, \mathbf{y}, p \in \mathbb{R}, \gamma \in \mathbb{R}$)

Input: $\mathbf{x}$: Source sentence
       $\mathbf{y}$: Translation prefix (default: ϵ)
       p: $\log P(\mathbf{y}|\mathbf{x})$ (default: 0.0)
       γ: Lower bound
1: **if** $y_{|\mathbf{y}|} = </\mathbf{s}>$ **then**
2:    **return** $(\mathbf{y}, p)$ {Trigger γ update}
3: **end if**
4: $\tilde{\mathbf{y}} \leftarrow \perp$ {Initialize $\tilde{\mathbf{y}}$ with dummy value}
5: **for all** $w \in \mathcal{T}$ **do**
6:    $p' \leftarrow p + \log P(w|\mathbf{x}, \mathbf{y})$
7:    **if** $p' \geq \gamma$ **then**
8:       $(\mathbf{y}', \gamma') \leftarrow \text{DFS}(\mathbf{x}, \mathbf{y} \cdot w, p', \gamma)$
9:       **if** $\gamma' > \gamma$ **then**
10:          $(\tilde{\mathbf{y}}, \gamma) \leftarrow (\mathbf{y}', \gamma')$
11:       **end if**
12:    **end if**
13: **end for**
14: **return** $(\tilde{\mathbf{y}}, \gamma)$

2 Exact Inference for Neural Models

Decoding in NMT (Eq. 2) is usually tackled with beam search, which is a time-synchronous approximate search algorithm that builds up hypotheses from left to right. A formal algorithm description is given in Alg. 1. Beam search maintains a set of active hypotheses $\mathcal{H}_{cur}$. In each iteration, all hypotheses in $\mathcal{H}_{cur}$ that do not end with the end-of-sentence symbol $</\mathbf{s}>$ are expanded and collected in $\mathcal{H}_{next}$. The best n items in $\mathcal{H}_{next}$ constitute the set of active hypotheses $\mathcal{H}_{cur}$ in the next iteration (line 11 in Alg. 1), where n is the beam size. The algorithm terminates when the best hypothesis in $\mathcal{H}_{cur}$ ends with the end-of-sentence symbol $</\mathbf{s}>$. Hypotheses are called *complete* if they end with $</\mathbf{s}>$ and *partial* if they do not.

Beam search is the ubiquitous decoding algorithm for NMT, but it is prone to search errors as the number of active hypotheses is limited by n. In particular, beam search never compares partial hypotheses of different lengths with each other. As we will see in later sections, this is one of the main sources of search errors. However, in many cases, the model score found by beam search is a reasonable approximation to the global best model score. Let γ be the model score found by beam search ($\tilde{p}$ in line 12, Alg. 1), which is a lower bound on the global best model score: $\gamma \leq \log P(\hat{\mathbf{y}}|\mathbf{x})$. Furthermore, since the conditionals $\log P(y_j|y_1^{j-1}, \mathbf{x})$ in Eq. 1 are log-probabilities and thus non-positive, expanding a partial hypothesis is guaranteed to result in a lower model score, i.e.:[1]

$$\forall j \in [2, J] : \log P(y_1^{j-1}|\mathbf{x}) > \log P(y_1^j|\mathbf{x}). \quad (3)$$

Consequently, when we are interested in the global best hypothesis $\hat{\mathbf{y}}$, we only need to consider partial hypotheses with scores greater than γ. In our exact decoding scheme we traverse the NMT search space in a depth-first order, but cut off branches along which the accumulated model score falls below γ. During depth-first search (DFS), we update γ when we find a better complete hypothesis.

---

[1]Equality in Eq. 3 is impossible since probabilities are modeled by the neural model via a softmax function which never predicts a probability of *exactly* 1.

Alg. 2 specifies the DFS algorithm formally. An important detail is that elements in $\mathcal{T}$ are ordered such that the loop in line 5 considers the $</\text{s}>$ token first. This often updates γ early on and leads to better pruning in subsequent recursive calls.[2]

Exact inference under length constraints Our admissible pruning criterion based on γ relies on the fact that the model score of a (partial) hypothesis is always lower than the score of any of its translation prefixes. While this monotonicity condition is true for vanilla NMT (Eq. 3), it does not hold for methods like length normalization (Jean et al., 2015; Boulanger-Lewandowski et al., 2013; Wu et al., 2016) or word rewards (He et al., 2016): Length normalization gives an advantage to longer hypotheses by dividing the score by the sentence length, while a word reward directly violates monotonicity as it rewards each word with a positive value. In Sec. 4 we show how our exact search can be extended to handle arbitrary length models (Murray and Chiang, 2018; Huang et al., 2017; Yang et al., 2018) by introducing length dependent lower bounds γ_k and report initial findings on exact search under length normalization. However, despite being of practical use, methods like length normalization and word penalties are rather heuristic as they do not have any justification from a probabilistic perspective. They also do not generalize well as (without re-tuning) they often work only for a specific beam size. It would be much more desirable to fix the length bias in the NMT model itself.

3 Results without Length Constraints

We conduct all our experiments in this section on the entire English-German WMT `news-test2015` test set (2,169 sentences) with a Transformer base (Vaswani et al., 2017) model trained with Tensor2Tensor (Vaswani et al., 2018) on parallel WMT18 data excluding ParaCrawl. Our pre-processing is as described by Stahlberg et al. (2018a) and includes joint subword segmentation using byte pair encoding (Sennrich et al., 2016) with 32K merges. We report cased BLEU scores.[3] An open-source implementation of our exact inference scheme is available in the

Search	BLEU	Ratio	#Search errors	#Empty
Greedy	29.3	1.02	73.6%	0.0%
Beam-10	30.3	1.00	57.7%	0.0%
Exact	2.1	0.06	0.0%	51.8%

Table 1: NMT with exact inference. In the absence of search errors, NMT often prefers the empty translation, causing a dramatic drop in length ratio and BLEU.
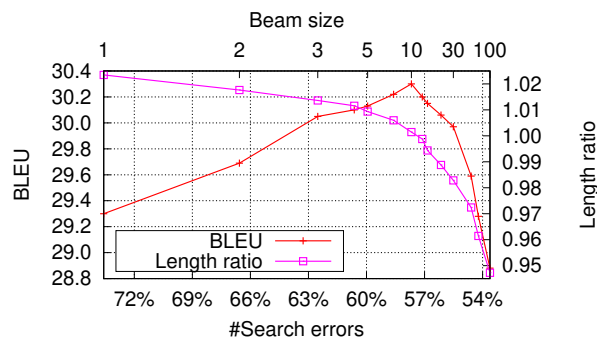

Figure 1: BLEU over the percentage of search errors. Large beam sizes yield fewer search errors but the BLEU score suffers from a length ratio below 1.
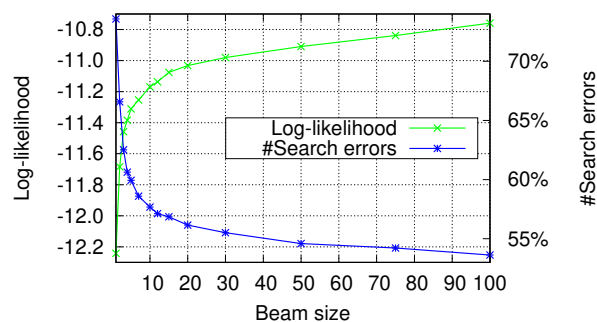

Figure 2: Even large beam sizes produce a large number of search errors.

SGNMT decoder (Stahlberg et al., 2017, 2018b).[4]

Our main result is shown in Tab. 1. Greedy and beam search both achieve reasonable BLEU scores but rely on a high number of search errors[5] to not be affected by a serious NMT model error: For 51.8% of the sentences, NMT assigns the global best model score to the empty translation, i.e. a single $</\text{s}>$ token. Fig. 1 visualizes the relationship between BLEU and the number of search errors. Large beam sizes reduce the number of search errors, but the BLEU score drops because translations are too short. Even a large beam size of 100 produces 53.62% search errors. Fig. 2 shows that beam search effectively reduces search

---

[2]Note that the order in which the for-loop in line 5 of Alg. 2 iterates over $\mathcal{T}$ may be important for efficiency but does not affect the correctness of the algorithm.

[3]Comparable with http://matrix.statmt.org/

[4]http://ucam-smt.github.io/sgnmt/html/, simpledfs decoding strategy.

[5]A sentence is classified as search error if the decoder does not find the global best model score.

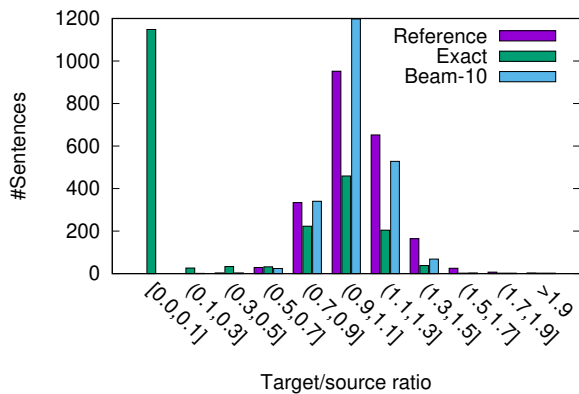Figure 3: Histogram over target/source length ratios.

Model	Beam-10		Exact
	BLEU	#Search err.	#Empty
LSTM*	28.6	58.4%	47.7%
SliceNet*	28.8	46.0%	41.2%
Transformer-Base	30.3	57.7%	51.8%
Transformer-Big*	31.7	32.1%	25.8%

Table 2: *: The recurrent LSTM, the convolutional SliceNet (Kaiser et al., 2017), and the Transformer-Big systems are strong baselines from a WMT'18 shared task submission (Stahlberg et al., 2018a).

errors with respect to greedy decoding to some degree, but is ineffective in reducing search errors even further. For example, Beam-10 yields 15.9% fewer search errors (absolute) than greedy decoding (57.68% vs. 73.58%), but Beam-100 improves search only slightly (53.62% search errors) despite being 10 times slower than beam-10.

The problem of empty translations is also visible in the histogram over length ratios (Fig. 3). Beam search – although still slightly too short – roughly follows the reference distribution, but exact search has an isolated peak in $[0.0, 0.1]$ from the empty translations.

Tab. 2 demonstrates that the problems of search errors and empty translations are not specific to the
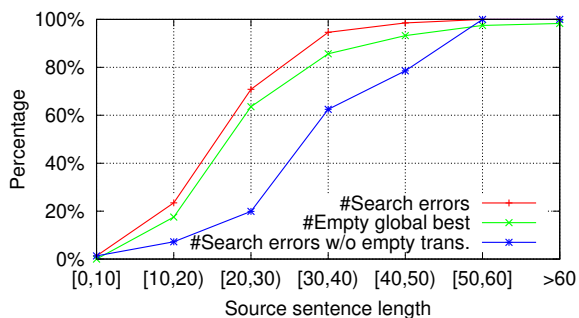

Figure 4: Number of search errors under Beam-10 and empty global bests over the source sentence length.
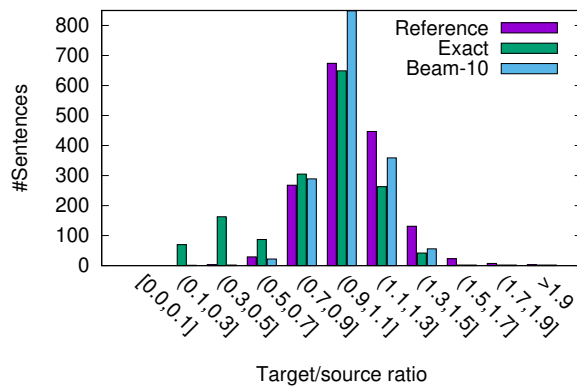
Figure 5: Histogram over length ratios with minimum translation length constraint of 0.25 times the source sentence length. Experiment conducted on 73.0% of the test set.

Transformer base model and also occur with other architectures. Even a highly optimized Transformer Big model from our WMT18 shared task submission (Stahlberg et al., 2018a) has 25.8% empty translations.

Fig. 4 shows that long source sentences are more affected by both beam search errors and the problem of empty translations. The global best translation is empty for almost all sentences longer than 40 tokens (green curve). Even without sentences where the model prefers the empty translation, a large amount of search errors remain (blue curve).

4 Results with Length Constraints

To find out more about the length deficiency we constrained exact search to certain translation lengths. Constraining search that way increases the run time as the γ-bounds are lower. Therefore, all results in this section are conducted on only a subset of the test set to keep the runtime under control.[6] We first constrained search to translations longer than 0.25 times the source sentence length and thus excluded the empty translation from the search space. Although this mitigates the problem slightly (Fig. 5), it still results in a peak in the $(0.3, 0.5]$ cluster. This suggests that the problem of empty translations is the consequence of an inherent model bias towards shorter hypotheses and cannot be fixed with a length constraint.

We then constrained exact search to either the

---

[6]We stopped decoding if the decoder took longer than a day for a single sentence on a single CPU. Exact search *without* length constraints is much faster and does not need maximum execution time limits.

Search	BLEU	Ratio
Beam-10	37.0	1.00
Exact for Beam-10 length	37.0	1.00
Exact for reference length	37.9	1.01

Table 3: Exact search under length constraints. Experiment conducted on 48.3% of the test set.

Search	W/o length norm.		With length norm.	
	BLEU	Ratio	BLEU	Ratio
Beam-10	37.0	1.00	36.3	1.03
Beam-30	36.7	0.98	36.3	1.04
Exact	27.2	0.74	36.4	1.03

Table 4: Length normalization fixes translation lengths, but prevents exact search from matching the BLEU score of Beam-10. Experiment conducted on 48.3% of the test set.

length of the best Beam-10 hypothesis or the reference length. Tab. 3 shows that exact search constrained to the Beam-10 hypothesis length does not improve over beam search, suggesting that any search errors between beam search score and global best score for that length are insignificant enough so as not to affect the BLEU score. The oracle experiment in which we constrained exact search to the correct reference length (last row in Tab. 3) improved the BLEU score by 0.9 points.

A popular method to counter the length bias in NMT is *length normalization* (Jean et al., 2015; Boulanger-Lewandowski et al., 2013) which simply divides the sentence score by the sentence length. We can find the global best translations under length normalization by generalizing our exact inference scheme to *length dependent* lower bounds γ_k. The generalized scheme[7] finds the best model scores for each translation length k in a certain range (e.g. zero to 1.2 times the source sentence length). The initial lower bounds are derived from the Beam-10 hypothesis $\mathbf{y}_{\text{beam}}$ as follows:[8]

$$\gamma_k = (k+1)\frac{\log P(\mathbf{y}_{\text{beam}}|\mathbf{x})}{|\mathbf{y}_{\text{beam}}| + 1}. \quad (4)$$

Exact search under length normalization does not suffer from the length deficiency anymore (last row in Tab. 4), but it is not able to match our best BLEU score under Beam-10 search. This suggests that while length normalization biases search towards translations of roughly the correct length, it does not fix the fundamental modelling problem.

---

[7]Available in our SGNMT decoder (Stahlberg et al., 2017, 2018b) as `simplelendfs` strategy.

[8]We add 1 to the lengths to avoid division by zero errors.

5   Related Work

Other researchers have also noted that large beam sizes yield shorter translations (Koehn and Knowles, 2017). Sountsov and Sarawagi (2016) argue that this model error is due to the locally normalized maximum likelihood training objective in NMT that underestimates the margin between the correct translation and shorter ones if trained with regularization and finite data. A similar argument was made by Murray and Chiang (2018) who pointed out the difficulty for a locally normalized model to estimate the "budget" for all remaining (longer) translations. Kumar and Sarawagi (2019) demonstrated that NMT models are often poorly calibrated, and that that can cause the length deficiency. Ott et al. (2018) argued that uncertainty caused by noisy training data may play a role. Chen et al. (2018) showed that the consistent best string problem for RNNs is decidable. We provide an alternative DFS algorithm that relies on the monotonic nature of model scores rather than consistency, and that often converges in practice.

To the best of our knowledge, this is the first work that reports the exact number of search errors in NMT as prior work often relied on approximations, e.g. via n-best lists (Niehues et al., 2017) or constraints (Stahlberg et al., 2018b).

6   Conclusion

We have presented an exact inference scheme for NMT. Exact search may not be practical, but it allowed us to discover deficiencies in widely used NMT models. We linked deteriorating BLEU scores of large beams with the reduction of search errors and showed that the model often prefers the empty translation – an evidence of NMT's failure to properly model adequacy. Our investigations into length constrained exact search suggested that simple heuristics like length normalization are unlikely to remedy the problem satisfactorily.

References

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

Nicolas Boulanger-Lewandowski, Yoshua Bengio, and Pascal Vincent. 2013. Audio chord recognition with recurrent neural networks. In *ISMIR*, pages 335–340. Citeseer.

Yining Chen, Sorcha Gilroy, Andreas Maletti, Jonathan May, and Kevin Knight. 2018. Recurrent neural networks as weighted language recognizers. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2261–2271, New Orleans, Louisiana. Association for Computational Linguistics.

Wei He, Zhongjun He, Hua Wu, and Haifeng Wang. 2016. Improved neural machine translation with SMT features. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 151–157. AAAI Press.

Liang Huang, Kai Zhao, and Mingbo Ma. 2017. When to finish? Optimal beam search for neural text generation (modulo beam size). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2134–2139, Copenhagen, Denmark. Association for Computational Linguistics.

Sébastien Jean, Orhan Firat, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. 2015. Montreal neural machine translation systems for WMT'15. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 134–140, Lisbon, Portugal. Association for Computational Linguistics.

Lukasz Kaiser, Aidan N Gomez, and Francois Chollet. 2017. Depthwise separable convolutions for neural machine translation. *arXiv preprint arXiv:1706.03059*.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709. Association for Computational Linguistics.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Aviral Kumar and Sunita Sarawagi. 2019. Calibration of encoder decoder models for neural machine translation. *arXiv preprint arXiv:1903.00802*.

Kenton Murray and David Chiang. 2018. Correcting length bias in neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 212–223, Belgium, Brussels. Association for Computational Linguistics.

Jan Niehues, Eunah Cho, Thanh-Le Ha, and Alex Waibel. 2017. Analyzing neural MT search and model performance. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 11–17.

Myle Ott, Michael Auli, David Grangier, et al. 2018. Analyzing uncertainty in neural machine translation. In *International Conference on Machine Learning*, pages 3953–3962.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Pavel Sountsov and Sunita Sarawagi. 2016. Length bias in encoder decoder models and a case for global conditioning. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1516–1525, Austin, Texas. Association for Computational Linguistics.

Felix Stahlberg, Adrià de Gispert, and Bill Byrne. 2018a. The University of Cambridge's machine translation systems for WMT18. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 504–512, Belgium, Brussels. Association for Computational Linguistics.

Felix Stahlberg, Eva Hasler, Danielle Saunders, and Bill Byrne. 2017. SGNMT – A flexible NMT decoding platform for quick prototyping of new models and search strategies. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 25–30. Association for Computational Linguistics.

Felix Stahlberg, Danielle Saunders, Gonzalo Iglesias, and Bill Byrne. 2018b. Why not be versatile? Applications of the SGNMT decoder for machine translation. In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers)*, pages 208–216. Association for Machine Translation in the Americas.

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, Samy Bengio, Eugene Brevdo, Francois Chollet, Aidan N. Gomez, Stephan Gouws, Llion Jones, Łukasz Kaiser, Nal Kalchbrenner, Niki Parmar, Ryan Sepassi, Noam Shazeer, and Jakob Uszkoreit. 2018. Tensor2tensor for neural machine translation. *CoRR*, abs/1803.07416.

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*, pages 5998–6008.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

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*, pages 3054–3059, Brussels, Belgium. Association for Computational Linguistics.