Neural Machine Translation Decoding with Terminology Constraints

Eva Hasler[1], **Adrià de Gispert**[1,2], **Gonzalo Iglesias**[1], **Bill Byrne**[1,2]

[1]SDL Research, Cambridge, UK
[2]Department of Engineering, University of Cambridge, UK
{ehasler,agispert,giglesias,bbyrne}@sdl.com

Abstract

Despite the impressive quality improvements yielded by neural machine translation (NMT) systems, controlling their translation output to adhere to user-provided terminology constraints remains an open problem. We describe our approach to constrained neural decoding based on finite-state machines and multi-stack decoding which supports target-side constraints as well as constraints with corresponding aligned input text spans. We demonstrate the performance of our framework on multiple translation tasks and motivate the need for constrained decoding with attentions as a means of reducing misplacement and duplication when translating user constraints.

1 Introduction

Adapting an NMT system with domain-specific data is one way to adjust its output vocabulary to better match the target domain (Luong and Manning, 2015; Sennrich et al., 2016). Another way to encourage the beam decoder to produce certain words in the output is to explicitly reward n-grams provided by an SMT system (Stahlberg et al., 2017) or language model (Gulcehre et al., 2017) or to modify the vocabulary distribution of the decoder with external suggestions from a terminology (Chatterjee et al., 2017). While providing lexical guidance to the decoder, none of these methods strictly enforces a terminology. This is a requisite, however, for companies wanting to ensure that all brand-related information is rendered correctly and consistently when translating web content or manuals and is often more important than translation quality alone. Although domain adaptation and guided decoding can help to reduce errors in these use cases, they do not provide reliable solutions.

Another recent line of work strictly enforces a given set of words in the output (Anderson et al., 2017; Hokamp and Liu, 2017; Crego et al., 2016). Anderson et al. address the task of image captioning with *constrained beam search* where constraints are given by image tags and constraint permutations are encoded in a finite-state acceptor (FSA). Hokamp and Liu propose *grid beam search* to enforce target-side constraints for domain adaptation via terminology. However, since there is no correspondence between constraints and the source words they cover, correct constraint placement is not guaranteed and the corresponding source words may be translated more than once. Crego et al. replace entities with special tags that remain unchanged during translation and are replaced in a post-processing step using attention weights. Given good alignments, this method can translate entities correctly but it requires training data with entity tags and excludes the entities from model scoring.

We address decoding with constraints to produce translations that respect the terminologies of corporate customers while maintaining the high quality of unconstrained translations. To this end, we apply the constrained beam search of Anderson et al. to machine translation and propose to employ alignment information between target-side constraints and their corresponding source words. The lack of explicit alignments in NMT systems poses an extra challenge compared to statistical MT where alignments are given by translation rules. We address the problem of *constraint placement* by expanding constraints when the NMT model is attending to the correct source span. We also reduce *output duplication* by masking covered constraints in the NMT attention model.

2 Constrained Beam Search

A naive approach to decoding with constraints would be to use a large beam size and select from the set of complete hypotheses the best that satisfies all constraints. However, this is infeasible in practice because it would require searching a potentially very large space to ensure that even hypotheses with low model score due to the inclusion of a constraint would be part of the set of outputs. A better strategy is to force the decoder to produce hypotheses that satisfy the constraints regardless of their score and thus guide the decoder into the right area of the search space. We follow Anderson et al. (2017) in organizing our beam search into multiple stacks corresponding to subsets of satisfied constraints as defined by FSA states.

2.1 Finite-state Acceptors for Constraints

Before decoding, we build an FSA defining the constrained target language for an input sentence. It contains all permutations of constraints interleaved with loops over the remaining vocabulary.

Phrase Constraints: Constraints consisting of multiple tokens are encoded by one state per token. We refer to states within a phrase as intermediate states and restrict their outgoing vocabulary to the next token in the phrase.

Alternative Constraints: Synonyms of constraints can be defined as alternatives and encoded as different arcs connecting the same states. When alternatives consist of multiple tokens, the alternative paths will contain intermediate states.

Figure 1 shows an FSA with constraints C_1 and C_2 where C_1 is a phrase (yielding intermediate states s_1, s_4) and C_2 consists of two single-token alternatives. Both permutations C_1C_2 and C_2C_1 lead to final state s_5 with both constraints satisfied.

2.2 Multi-Stack Decoding

When extending a hypothesis to satisfy a constraint which is not among the top-k vocabulary items in the current beam, the overall likelihood may drop and the hypothesis may be pruned in subsequent steps. To prevent this, the extended hypothesis is placed on a new stack along with other hypotheses that satisfy the same set of constraints. Each stack maps to an acceptor state which helps to keep track of the permitted extensions for hypotheses on this stack. The stack where a hypothesis should be placed is found by following the appropriate arc leaving the current acceptor state. The stack mapping to the final state is used to generate complete hypotheses. At each time step, all stacks are pruned to the beam size k and therefore

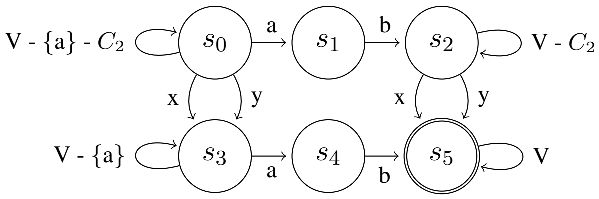
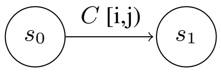


Figure 1: Example of FSA for two constraints $C_1 = ab$ and $C_2 = \{x, y\}$.

the actual beam size for constrained decoding depends on the number of acceptor states.

2.3 Decoding with Attentions

Since an acceptor encoding c single-token constraints has 2^c states, the constrained search of Anderson et al. (2017) can be inefficient for large numbers of constraints. In particular, all unsatisfied constraints are expanded at each time step t which increases decoding complexity from $\mathcal{O}(tk)$ for normal beam search to $\mathcal{O}(tk2^c)$. Hokamp and Liu (2017) organize their grid beam search into beams that group hypotheses with the same number of constraints, thus their decoding time is $\mathcal{O}(tkc)$. However, this means that different constraints will compete for completion of the same hypothesis and their placement is determined locally. We assume that a target-side constraint can come with an aligned source phrase which is encoded as a span in source sentence S and stored with the acceptor arc label:



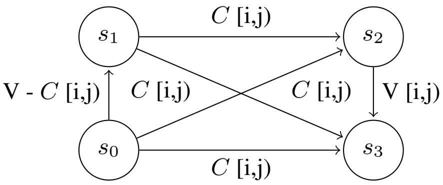
$j > i, \quad 0 \leq i, j \leq |S|$

Because the attention weights in attention-based decoders function as soft alignments from the target to the source sentence (Alkhouli and Ney, 2017), we use them to decide at which position a constraint should be inserted in the output. At each time step in a hypothesis, we determine the source position with the maximum attention. If it falls into a constrained source span and this span matches an outgoing arc in the current acceptor state, we extend the current hypothesis with the arc label. Thus, the outgoing arcs in non-intermediate states are active or inactive depending on the current attentions. This reduces the complexity from $\mathcal{O}(tk2^c)$ to $\mathcal{O}(tkc)$ by ignoring all but one constraint permutation and in practice, disabling vocabulary loops saves extra time.

State-specific Attention Mechanism: Once a constraint has been completed, we need to ensure that its source span will not be translated again. We force the decoder to respect covered constraints by masking their spans during all future expansions of the hypothesis. This is done by zeroing out the attention weights on covered positions to exclude them from the context vector computed by the attention mechanism.

Implications: Constrained decoding with aligned source phrases relies on the quality of the source-target pairs. Over- and under-translation can occur as a result of incomplete source or target phrases in the terminology.

Special Cases: Monitoring the source position with the maximum attention is a relatively strict criterion to decide where a constraint should be placed in the output. It turns out that depending on the language pair, the decoder may produce translations of neighbouring source tokens when attending to a constrained source span.[1] The strict requirement of only producing constraint tokens can be relaxed to accommodate such cases, for example by allowing extra tokens before (s_1) or after (s_2) constraint C while attending to span $[i, j)$,



Conversely, the decoder may never place the maximum attention on a constraint span which can lead to empty translations. Relaxing this requirement using thresholding on the attention weights to determine positions with secondary attention can help in those cases.

3 Experimental Setup

We build attention-based neural machine translation models (Bahdanau et al., 2015) using the Blocks implementation of van Merriënboer et al. (2015) for English-German and English-Chinese translation in both directions. We combine three models per language pair as ensembles and further combine the NMT systems with n-grams extracted from SMT lattices using Lattice minimum Bayes-risk as described by Stahlberg et al. (2017), referred to as LNMT. We decode with a beam size of 12 and length normalization (Johnson et al., 2017) and back off to constrained decoding without attentions when decoding with attentions fails.[2] We report lowercase BLEU using mteval-v13.pl.

3.1 Data

Our models are trained on the data provided for the 2017 Workshop for Machine Translation (Bojar et al., 2017). We tokenize and truecase the English-German data and apply compound splitting when the source language is German. The training data for the NMT systems is augmented with backtranslation data (Sennrich et al., 2016). For English-Chinese, we tokenize and lowercase the data. We apply byte-pair encoding (Sennrich et al., 2017) to all data.

3.2 Terminology Constraints

We run experiments with two types of constraints to evaluate our constrained decoder.

Gold Constraints: For each input sentence, we extract up to two tokens from the reference which were not produced by the baseline system, favouring rarer words. This aims at testing the performance in a setup where users may provide corrections to the NMT output which are to be incorporated into the translation. These reference tokens may consist of one or more subwords. Similarly, we extract phrases of up to five subwords surrounding a reference token missing from the baseline output. We do not have access to aligned source words for gold constraints.

Dictionary Constraints: We automatically extract bilingual user dictionary entries using terms and phrases from the reference translations as candidates in order to ensure that the entries are relevant for the inputs. In a real setup, these entries would be provided by customers and would be expected to be correct translations without ambiguity. We apply a filter of English stop words and verbs to the candidates and look them up in a pruned phrase table to find likely pairs. This results in entries as shown below:[3]

English	**German**
ICJ	IGH
The Wall Street Journal	The Wall Street Journal
Dead Sea	Tote Meer|Toten Meer

For evaluation purposes, we ensure that dictionary entries match the reference when applying them to an input sentence.

[1] For example, to produce an article before a noun when the constrained source span includes just the noun.

[2] This usually applies to less than 2% of the inputs.

[3] Our dictionaries are available on request.

	dev (lr)	rep	test15	test16	test17
eng-ger-wmt17					
LNMT	24.9 (1.00)	443	28.1	34.7	27.0
+ 2 gold tokens	29.2 (1.14)	1141	33.4	40.9	32.3
+ 1 gold phrase	36.8 (1.09)	880	40.5	46.7	39.6
+ dictionary (v1)	26.4 (1.03)	610	29.6	36.4	28.8
+ dictionary (v2)	26.6 (1.02)	471	29.9	37.0	29.1
ger-eng-wmt17					
LNMT	31.2 (1.01)	307	33.5	40.7	34.6
+ 2 gold tokens	34.6 (1.14)	745	37.7	44.8	38.5
+ 1 gold phrase	42.3 (1.08)	550	45.7	51.3	46.4
+ dictionary (v1)	32.4 (1.02)	353	34.7	41.8	36.2
+ dictionary (v2)	32.5 (1.01)	320	34.6	41.9	36.0

(a) Results for English-German language pairs

	dev (lr)	test17
eng-chi-wmt17		
LNMT	30.8 (0.95)	31.0
+ 2 gold tokens	33.8 (1.10)	34.2
+ 1 gold phrase	40.6 (1.06)	41.2
+ dictionary (v1)	34.0 (1.01)	33.7
+ dictionary (v2)	33.9 (0.98)	34.1
chi-eng-wmt17		
LNMT	21.2 (1.00)	23.5
+ 2 gold tokens	23.3 (1.13)	25.5
+ 1 gold phrase	30.1 (1.09)	32.3
+ dictionary (v1)	23.0 (1.06)	25.5
+ dictionary (v2)	23.4 (1.03)	25.4

(b) Results for English-Chinese language pairs

Table 1: BLEU scores and dev length ratios for decoding with gold constraints (without attentions) followed by results for dictionary constraints without (v1) or with (v2) attentions. The column *rep* shows the number of repeated character 7-grams within the same sentence, see Section 4.3.

4 Results

The results for decoding with terminology constraints are shown in Tab. 1a and 1b where each section contains the results for gold constraints followed by dictionary constraints.

4.1 Results with Gold Constraints

Decoding with gold constraints yields large BLEU gains over LNMT for all language pairs. However, the length ratio on the dev set increases significantly. Inspecting the output reveals that this is often caused by constraints being translated more than once which can lead to whole passages being retranslated. Phrase constraints seem to integrate better into the output than single token constraints which may be due to the longer gold context being fed back to the NMT state.

4.2 Results with Dictionary Constraints

Decoding with up to two dictionary constraints per sentence yields gains of up to 3 BLEU. This is partly because we do not control whether LNMT already produced the constraint tokens and because not all sentences have dictionary matches. The length ratios are better compared to the gold experiments which we attribute to our filtering of tokens such as verbs which tend to influence the general word order more than nouns, for example.

Decoding with or without attentions yields similar BLEU scores overall and a consistent improvement for English-German. Note that decoding with attentions is sensitive to errors in the automatically extracted dictionary entries.

Output Duplication The first three examples in Tab. 2 show English↔German translations where decoding without attentions has generated both the target side of the constraint and the translation preferred by the NMT system. When using attentions, the constraint is only translated once.

Constraint Placement The fourth example demonstrates the importance of tying constraints to source words. Decoding without attentions fails to translate *Zeichen* as *signs* because the alternative *sign* already appears in the translation of *Zeichensprache* as *sign language*. With attentions, *signs* is generated at the correct position.

4.3 Output length ratio and repetitions

To back up our hypothesis that increases in length ratio are related to output duplication, Tab. 1a column *rep* shows the number of repeated character 7-grams within the same sentence, ignoring stop words and overlapping n-grams. This confirms that constrained decoding with attentions reduces the number of repeated n-grams in the output. While this does not take alignments to the source into account nor does it capture duplicated translations with unrelated source forms, it provides some evidence that the outputs are not just shorter than for decoding without attentions but in fact contain fewer repetitions and likely fewer duplicated translations.

eng-ger-wmt17	*Example 1*	*Example 2*
Source	It already has the **budget** ...	And it often costs over a hundred dollars to obtain the required **identity card**.
Constraints	Budget [4,5)	Ausweis [12,14)
LNMT	Es hat bereits den **Haushalt**...	Und es kostet oft mehr als hundert Dollar, um die erforderliche **Personalausweis** zu erhalten.
+ dictionary (v1)	Das **Budget** hat bereits den **Haushalt**...	Und es kostet oft mehr als hundert Dollar, um den **Ausweis** zu erhalten, um die erforderliche **Personalausweis** zu erhalten.
+ dictionary (v2)	Es verfügt bereits über das **Budget**...	Und es kostet oft mehr als hundert Dollar, um den gewünschten **Ausweis** zu erhalten.
ger-eng-wmt17	*Example 3*	*Example 4*
Source	Der **Pokal** war die einzige **Möglichkeit** , etwas zu gewinnen .	Aber es ist keine typische Zeichensprache – sagt sie . Edmund hat einige **Zeichen** alleine erfunden .
Constraints	cup [1,2), chance [5,6)	sign|signs [13,14)
LNMT	The **trophy** was the only **way** to win something.	But it's not a typical sign language – says, Edmund invented some **characters** alone.
+ dictionary (v1)	The **cup** was the only **way** to get something to win a **chance**.	But it's not a typical sign language – says, Edmund invented some **characters** alone.
+ dictionary (v2)	The **cup** was the only **chance** to win something.	But it is not a typical sign language – she says, Edmund invented some **signs** alone.

Table 2: English↔German translation outputs for constrained decoding.

	BLEU/speed ratio					
eng-ger-wmt17	c=2		c=3		c=4	
LNMT	26.7	1.00	26.7	1.00	26.7	1.00
+ dict (v1)	28.2	0.20	28.4	0.14	28.5	0.11
+ dict (v2*)	27.8	0.69	28.0	0.66	28.1	0.59
+ A	28.0	0.65	28.2	0.61	28.2	0.54
+ B	28.4	0.27	28.6	0.24	28.7	0.21
+ C	28.5	0.21	28.6	0.19	28.7	0.17

Table 3: BLEU scores and speed ratios relative to unconstrained LNMT for production system with up to c constraints per sentence (newstest2017). A: secondary attention, B, C: allow 1 or 2 extra tokens, respectively (Section 2.3). Dict (v2*) refers to decoding with attentions but without A, B or C.

4.4 Comparison of decoding speeds

To evaluate the speed of constrained decoding with and without attentions, we decode newstest-2017 on a single GPU using our English-German production system (Iglesias et al., 2018) which in comparison to the systems described in Section 3 uses a beam size of 4 and an early pruning strategy similar to that described in Johnson et al. (2017), amongst other differences. About 89% of the sentences have at least one dictionary match and we allow up to two, three or four matches per sentence. Because the constraints result from dictionary application, the number of constraints per sentence varies and not all sentences contain the maximum number of constraints. Tab. 3 reports BLEU scores and speed ratios for different decoding configurations. Rows two and three confirm that the reduced computational complexity of our approach yields faster decoding speeds than the approach of Anderson et al. (2017) while incurring a small decrease in BLEU. Moreover, it compares favourably for larger numbers of constraints per sentence: v2* is 3.5x faster than v1 for c=2 and more than 5x faster for c=4. Relaxing the restrictions of decoding with attentions improves the BLEU scores but increases runtime. However, the slowest v2 configuration is still faster than v1. The optimal trade-off between quality and speed is likely to differ for each language pair.

5 Conclusion

We have presented our approach to NMT decoding with terminology constraints using decoder attentions which enables reduced output duplication and better constraint placement compared to existing methods. Our results on four language pairs demonstrate that terminology constraints as provided by customers can be respected during NMT decoding while maintaining the overall translation quality. At the same time, empirical results confirm that our improvements in computational complexity translate into faster decoding speeds. Future work includes the application of our approach to more recent architectures such as Vaswani et al. (2017) which will involve extracting attentions from one or more decoding layers.

Acknowledgments

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

References

Tamer Alkhouli and Hermann Ney. 2017. Biasing Attention-Based Recurrent Neural Networks Using External Alignment Information. In *Proceedings of the Conference on Machine Translation (WMT), Volume 1: Research Papers*. Association for Computational Linguistics, pages 108–117. http://www.statmt.org/wmt17/pdf/WMT11.pdf.

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2017. Guided Open Vocabulary Image Captioning with Constrained Beam Search. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pages 936–945. https://aclweb.org/anthology/D17-1098.

Dzmitry Bahdanau, KyungHyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proceedings of ICLR 2015*. https://arxiv.org/pdf/1409.0473.pdf.

Ondrej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Raphael Rubino, Lucia Specia, and Marco Turchi. 2017. Findings of the 2017 Conference on Machine Translation (WMT17). In *Proceedings of the Conference on Machine Translation (WMT), Volume 2: Shared Task Papers*. Association for Computational Linguistics, pages 169–214. http://www.statmt.org/wmt17/pdf/WMT17.pdf.

Rajen Chatterjee, Matteo Negri, Marco Turchi, Marcello Federico, Lucia Specia, and Frédéric Blain. 2017. Guiding Neural Machine Translation Decoding with External Knowledge. In *Proceedings of the Conference on Machine Translation (WMT), Volume 1: Research Papers*. Association for Computational Linguistics, pages 157–168. https://aclweb.org/anthology/W17-4716.

Josep Crego, Jungi Kim, Guillaume Klein, Anabel Rebollo, Kathy Yang, Jean Senellart, Egor Akhanov, Patrice Brunelle, Aurélien Coquard, Yongchao Deng, Satoshi Enoue, Chiyo Geiss, Joshua Johanson, Ardas Khalsa, Raoum Khiari, Byeongil Ko, Catherine Kobus, Jean Lorieux, Leidiana Martins, Dang-Chuan Nguyen, Alexandra Priori, Thomas Riccardi, Natalia Segal, Christophe Servan, Cyril Tiquet, Bo Wang, Jin Yang, Dakun Zhang, Jing Zhou, and Peter Zoldan. 2016. SYSTRAN's Pure Neural Machine Translation Systems. Arxiv preprint. https://arxiv.org/pdf/1610.05540v1.pdf.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, and Yoshua Bengio. 2017. On integrating a language model into neural machine translation. *Computer Speech & Language* 45:137–148. https://doi.org/10.1016/j.csl.2017.01.014.

Chris Hokamp and Qun Liu. 2017. Lexically Constrained Decoding for Sequence Generation Using Grid Beam Search. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, pages 1535–1546. https://doi.org/10.18653/v1/P17-1141.

Gonzalo Iglesias, William Tambellini, Adrià de Gispert, Eva Hasler, and Bill Byrne. 2018. Accelerating NMT Batched Beam Decoding with LMBR Posteriors for Deployment. In *Proceedings of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Industry Track)*. Association for Computational Linguistics.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation. *Transactions of the Association for Computational Linguistics* 5:339–351. https://aclweb.org/anthology/Q17-1024.

Minh-Thang Luong and Christopher D Manning. 2015. Stanford Neural Machine Translation Systems for Spoken Language Domains. In *Proceedings of the 12th International Workshop on Spoken Language Translation*. pages 76–79. https://nlp.stanford.edu/pubs/luong-manning-iwslt15.pdf.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving Neural Machine Translation Models with Monolingual Data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. pages 86–96. https://aclweb.org/anthology/P16-1009.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2017. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, pages 1715–1725. https://aclweb.org/anthology/P16-1162.

Felix Stahlberg, Adrià de Gispert, Eva Hasler, and Bill Byrne. 2017. Neural Machine Translation by Minimising the Bayes-risk with Respect to Syntactic Translation Lattices. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2,* *Short Papers*. Association for Computational Linguistics, pages 362–368. https://aclweb.org/anthology/E/E17/E17-2058.pdf.

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. In *Proceedings of ICLR 2015*. Arxiv preprint. https://arxiv.org/abs/1506.00619.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, Curran Associates, Inc., pages 5998–6008. http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.