Topic-Informed Neural Machine Translation

Jian Zhang, Liangyou Li, Andy Way, Qun Liu
ADAPT Centre
School of Computing
Dublin City University, Ireland
jian.zhang,liangyou.li,andy.way,qun.liu@adaptcentre.ie

Abstract

In recent years, neural machine translation (NMT) has demonstrated state-of-the-art machine translation (MT) performance. It is a new approach to MT, which tries to learn a set of parameters to maximize the conditional probability of target sentences given source sentences. In this paper, we present a novel approach to improve the translation performance in NMT by conveying topic knowledge during translation. The proposed *topic-informed* NMT can increase the likelihood of selecting words from the same topic and domain for translation. Experimentally, we demonstrate that topic-informed NMT can achieve a 1.15 (3.3% relative) and 1.67 (5.4% relative) absolute improvement in BLEU score on the Chinese-to-English language pair using NIST 2004 and 2005 test sets, respectively, compared to NMT without topic information.

1 Introduction

In statistical machine translation (SMT) (Och and Ney, 2000; Marcu and Wong, 2002; Koehn et al., 2007), several models are trained separately and then linearly integrated using the log-linear model (Och, 2003). Neural machine translation (NMT) (Cho et al., 2014; Sutskever et al., 2014; Bahdanau et al., 2015; Luong et al., 2015), being a new approach, employs an individual large neural network to maximize the translation probability directly.

One observation we obtain from machine translation (MT) training data is that some of the words within the same sentence often belong to the same or similar topic. Examine the example presented in Figure 1, where words *Commercial*, *market*, *prices* and *bank* have higher probabilities of being in the *Financial* domain. Intuitively, by allowing the topic information of the source input sentence and previous translated words to be provided to the decoder, we can maintain the same topic in the translations during the decoding phase, and consequently better translations can be produced. Specifically, when a source word has more than one translation option available (e.g. the word *bank* in Figure 1), it is clear that the translation in the *Financial* domain is more likely to be selected because many of the source words are from the same topic.

Topic models have been applied successfully in many SMT works. For example, similar "topic consistent" behaviour is also observed by Su (2015). In his work, a context-aware topic model is integrated into SMT for better lexical selection. Xiao et al. (2012) and Zhang et al. (2014) focus on document translations and propose a topic-similarity model and a topic-sensitivity model for hierarchical phrase-based SMT (Chiang, 2005) on the document level. The topic-similarity model is used to encourage or penalize topic-sensitive rules, and the topic-sensitivity model is applied to balance topic-insensitive rules. However, these approaches cannot be directly used in NMT.

In this work, we propose our novel topic-informed NMT model. In topic models, word-topic distributions can be viewed as a vector. In NMT, words are represented as vector-space representations. Thus, it is very natural to use them together. Word expressions in neural models are derived by its near context words while the topic vectors represent the document-level topic information. For this reason, we see

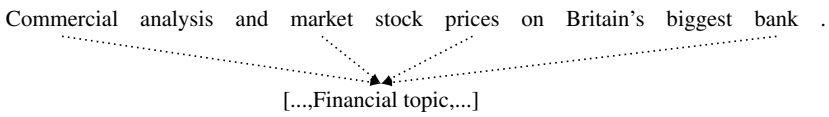

Figure 1: Some words within the same sentence belong to the same topic.

them as complementary to one another. Our intuition is that incorporating topic information will benefit NMT. Thus, we explore incorporating topic information into NMT in either/both the source side and target side.

The remainder of this paper is structured as follows. Section 2 presents related work. Section 3 gives review of the NMT architecture we use. Section 4 introduces our topic-informed NMT system, while Section 5 presents experimental results and some observations. Finally, the conclusions and future directions are provided in Section 6.

2 Related Work

Topic modelling has been applied successfully in many SMT works, especially in the domain adaptation literature. The motivation for introducing topic-model information in MT is that the translation performance decreases when there are dissimilarities between the training and the testing domains. A better approach is to make the use of the topic knowledge learned during training. Such knowledge can yield a better word or phrase choice in translation. Early work shows that the lexical translation table conditioned on the provenance of each domain can significantly improve translation quality (Chiang et al., 2011). Eidelman (2012) extends the provenance idea by including topic-dependent lexical weighting probabilities on the source side. Hasler (2012) successfully combines sparse word-pair and phrase-pair features with topic models. Later, Hasler (2014) also reports that translation performance is improved by using similarity features, which are computed from training phrase-pair and test-context vectors generated from the phrase-pair topic model. However, these ideas cannot be directly applied in NMT given the different training algorithms used in SMT and NMT.

Despite the fact that neural network training has been shown to be advantageous in many natural language processing tasks, little work has been proposed on using additional knowledge in NMT. Gulcehre et al. (2015) leverage monolingual corpora for NMT and propose two approaches to integrate a neural language model into the encoder-decoder architecture. He et al. (2016) integrate SMT features into NMT in order to solve the out-of-vocabulary problem. Furthermore, they use a n-gram language model trained on large monolingual data to enhance the local fluency of translation outputs. Linguistic information can also be used during NMT training (Sennrich and Haddow, 2016; García-Martínez et al., 2016). There are also works that include topic modelling in neural language model training. Mikolov and Zweig (2012) use a contextual real-valued input vector associated with each word in a recurrent neural network (RNN) language model.

3 Neural Machine Translation

3.1 Encoder-Decoder Architecture

In a nutshell, the fundamental job of the encoder-decoder architecture (Cho et al., 2014) in NMT is to probabilistically decode a target sequence given the encoded source sequence, where the two sequences can be of different lengths. Given a sentence pair (S, T), S is the foreign input sentence and T is the translation, where $S = (s_1, s_2, ..., s_{m-1}, s_m)$ and $T = (t_1, t_2, ..., t_{n-1}, t_n)$ are the words in the sentence pair. The encoder-decoder architecture needs to find the translation probabilities for each word in T, as in Equation (1):

$$p(T|S) = \sum_{j=1}^{n} p(t_j|t_{1:j-1}, S) \tag{1}$$

and the conditional probability is given by the decoder, which uses the $softmax$ function outputting the probability distribution on all words in the target, as in Equation (2):

$$p(t_j = t|t_{1:j-1}, S) = softmax(f(h_j)) \tag{2}$$

where f is a function that can transform the target context h_j into a vector, which has the same size with the size of target vocabulary. h_j is defined in Equation (3):

$$h_j = g(t_{j-1}, h_{j-1}, c) \tag{3}$$

where c is the source context vector computed by the encoder, t_{j-1} is the word vector representation of word $j-1$, h_{j-1} is the hidden state for time $j-1$ and g is a non-linear activation function. Thus, we use the source input sentence and previous translated words to predict the next word.

3.2 Attention Model

The encoder-decoder architecture uses a fixed-sized vector to represent the whole source input. Although RNNs are known to be better at capturing long range dependencies, experimental results (Bahdanau et al., 2015) show that translation quality decreases for long input sentences. Bahdanau (2015) use an attention mechanism to learn dynamic *soft-alignment* during the network training, as in Equation (4):

$$e_{ij} = a(h_{j-1}, h_i) \tag{4}$$

where e_{ij} is the alignment model to score the alignment at position i and j in S and T, respectively. h_{j-1} is the target hidden state as seen in Equation (3), and h_i is the source hidden state at time i.

Thus, a distinct context vector c_j can be computed for each word in T, and the source context vector c is rewritten as in (5):

$$c_j = \sum_{i=1}^{m} \alpha_{ij} h_i \tag{5}$$

where α_{ij} is a normalized weight for each hidden state in S, computed as in (6):

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_{i=1}^{m} exp(e_{ij})} \tag{6}$$

With the attentional model, source information can be spread across the source context vector, and the decoder can selectively pay attention to different parts of the source context during decoding.

3.3 Bidirectional RNN

During the encoding phase, words can also be fed into the encoder in both directions, in which case we are using a bidirectional RNN. The intuition behind such a model is to include both *positive* and *negative* time stamps of the source input during encoding, which can shorten the distance between the decoding part with the relevant encoded part. Sutskever et al. (2014) claim that it is "extremely valuable" and can "greatly boost the performance" by using a bidirectional RNN in NMT.

In this paper, following Cho et al. (2014) and Bahdanau et al. (2015), we use the encoder-decoder architecture with a single layer bidirectional gated recurrent unit (GRU) (Chung et al., 2014) as the encoder, and a single layer GRU with attention model as the decoder in our NMT system.[1]

4 Topic-Informed Neural Machine Translation

In this section, we provide an explanation of how to include topic knowledge in NMT.

[1]More hidden layers usually result in much better quality. However, the training time increases. Thus, we use only a single layer in the encoder or the decoder.

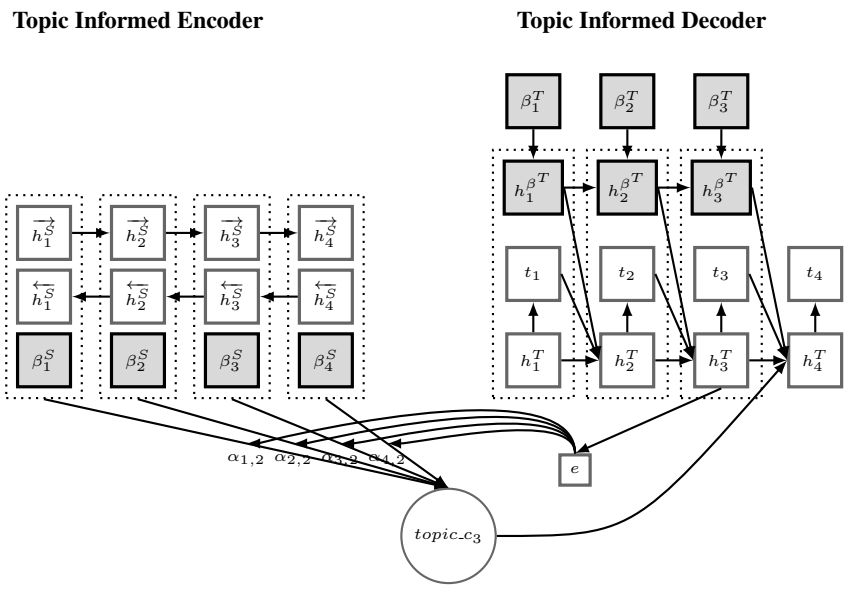

Figure 2: A graphical illustration of the proposed topic-informed NMT model, with 4 input words and 4 output words. The shaded units indicate the topic information used in the encoder and decoder. In order to distinguish the hidden states in the encoder-decoder architecture, we use h^S and h^T to represent the hidden states of the encoder and decoder, respectively, e.g. h_1^S indicates the encoder hidden state at time 1. Furthermore, in order to keep the notation consistent with Section 4.1, e.g. β_i^S denotes the word distributions over topics for the source word at position i, a similar notation is also used for the target words. For example, β_1^T indicates the word distributions over topics for the target word at position 1.

4.1 Topic-Informed Encoder

The encoder in standard NMT uses only word embedding to compute the source context vector. By using topic information on the source side, the decoder can have an overview of the source topics during decoding. Furthermore, the attention model can implicitly pay attention to the topic distributions of each source word. Topic information on the source side can be helpful to generate more accurate translations. Therefore, we first compute the topic distributions (word distributions over topics) for each source word of the input sentence. We then concatenate the topic distributions with each corresponding hidden state of the input source words. Finally, the hidden states with the topic information are used to compute the topic-informed source context vector $topic\_c_j$, as in Equation (7):

$$topic\_c_j = \sum_{i=1}^{m} \alpha_{ij}[h_i, \beta_i^S] \tag{7}$$

to obtain our topic-informed encoder, where β_i^S denotes the word distributions over topics for each source word in S, and $[h_i, \beta_i^S]$ denotes the concatenation operation on the corresponding h_i and β_i^S. Thus, Equation (3) is updated as in (8):

$$h_j = g(t_{j-1}, h_{j-1}, topic\_c_j) \tag{8}$$

in order to obtain the source topic-informed NMT model.

4.2 Topic-Informed Decoder

While the source topic-informed NMT model is useful for generating accurate translations, introducing topic information on the target side can help topic consistency between target words. A natural choice for maintaining this topic consistency is to use the same architecture as the decoder (i.e. GRU), to obtain the topic hidden state for the corresponding target word. We use $h_{j-1}^{\beta^T}$ to represent the hidden state of target topics for time $j-1$. We then can update Equation (3) as in (9):

$$h_j = g(t_{j-1}, h_{j-1}, c, h_{j-1}^{\beta^T}) \tag{9}$$

	Source	Target
Sentence Num.	1,501,652	1,501,652
Token Num.	38,388,118	44,901,788

Table 1: Training data statistics.

to obtain the target topic-informed NMT model. Consequently, the decoder can then use the topic knowledge of previous translated words to increase the likelihood of selecting words from the same topic.

4.3 Topic-Informed NMT

We now can combine the topic-informed encoder and the topic-informed decoder to obtain the overall topic-informed NMT system, as in Equation (10):

$$h_j = g(t_{j-1}, h_{j-1}, topic\_c_j, h_{j-1}^{\beta^T}) \tag{10}$$

Figure 2 provides a the graphical illustration of this novel topic-informed NMT model.

4.4 Topic Modeling

Instead of documents, sentences are often used to represent the mixture of topics in MT. For example, given the source or target training corpus, we can choose a fixed N topics to learn. The effectiveness of such an approach is that the topic representations can be learned directly using off-the-shelf algorithms. We take the same approach in this work, and use the translation training data directly to learn the topic representations. We use the Latent Dirichlet Allocation (LDA) implementation in the topic modeling toolkit (Řehůřek and Sojka, 2010) to train the topic model and compute the topic distributions of words.[2] We train the models with 200 iterations. Training takes approximately 40 hours on average for all the LDA models used in this work. For a detailed explanation of LDA, we refer the reader to Blei et al. (2003). Other options are to use Latent Semantic Analysis (LSA) (Deerwester et al., 1990) or Hidden Topic Markov Models (HTMM) (Gruber et al., 2007), which have not been studied in this work. We leave them as part of our future research.

5 Experiments

5.1 Data and Experiment Models

We report our experimental results on the NIST evaluation data set in the Chinese-to-English translation direction. Our MT training data are extracted from LDC corpora,[3] and NIST 2002 is used as our development set. Table 1 shows the details of the training data used. We use NIST 2004 and 2005 as our test sets. The English training data is tokenized and lowercased using scripts in Moses (Koehn et al., 2007). The Stanford Chinese word segmenter (Tseng et al., 2005) is used to segment the Chinese training data. In NMT, we limit our vocabularies to be the top 16,000 most frequent words, which covers 97.57% and 98.77% of the original words in the source and target training corpora, respectively. Outside the 16,000 threshold, all tokens are mapped to *UNK*.

We compare our approach against two baselines. The SMT baseline is trained using Moses, with a lexicalized reordering model (Koehn et al., 2005; Galley and Manning, 2008) and a 5-gram KenLM (Heafield, 2011) language model trained using the target side of the parallel training data. We use all the default parameters in Moses. Our second baseline is an NMT system. We use the encoder-decoder architecture with a single layer bidirectional GRU as the encoder, and a single layer GRU with attention model as the decoder. Each word in the training corpora is converted into a 512-dimensional vector during training. The hidden layers of the encoder and decoder each contain 1,024 hidden units. The bidirectional RNN described in Section 3.3 is also used. We use beam search during translation, with a beam size of 5. We use a minibatch (with batch size 32) stochastic gradient descent algorithm together with Adadelta (Zeiler, 2012) to train our models. All NMT models are trained up to 320,000 updates

[2] `https://radimrehurek.com/gensim/models/ldamulticore.html`

[3] LDC2002E18, LDC2003E07, LDC2003E14, LDC2004T07, the Hansards portion of LDC2004T08 and LDC2005T06

Systems	NIST 2002 (dev)	NIST 2004 (test)	NIST 2005 (test)
SMT	33.42	32.36	30.11
NMT	34.33	34.76	31.12
Source Topic-Informed NMT (40)	35.39	35.17†	31.95‡
Target Topic-Informed NMT (10)	36.31	35.43‡	32.50‡
Topic-Informed NMT (40,10)	34.86	**35.91**‡	**32.79**‡

Table 2: BLEU scores of the trained SMT and NMT models. We use ‡ and † to indicate significant (Koehn, 2004) improvements upon the baseline NMT using bootstrapping method at the level p = 0.01 and p = 0.05 level, respectively (with 1000 iterations).

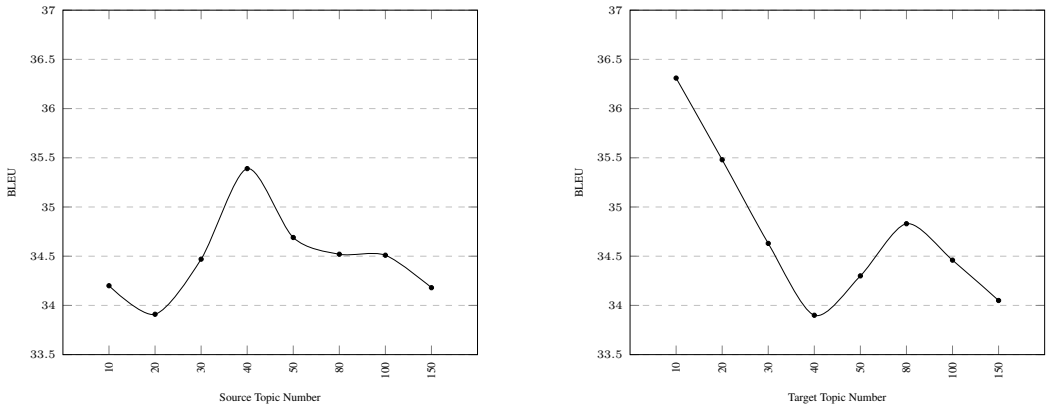

Figure 3: Topic numbers vs. translation BLEU scores on the NIST 2002 development dataset.

and the models are saved at each 1,000 updates. The training takes approximately 3 days on an NVIDIA GeForce GTX TITAN X GM200 GPU machine. We then choose the final model based on the BLEU[4] (Papineni et al., 2002) score on the development data.

5.2 Results

Table 2 presents the experiment results on the development and test data. In Table 2, the number next to each topic-informed NMT system indicates the number of topics used in the system, i.e. we use 40 source topics in the source topic-informed NMT system, and 10 target topics in the target topic-informed NMT model. The source and target topic numbers are experimentally chosen from $\{10, 20, 30, 40, 50, 80, 100, 150\}$ according to the development BLEU scores, as seen in Figure 3. We then leverage topic information on both source (with 40 topics) and target (with 10 topics) sides, which produces the topic-informed system (40,10) in Table 2. We think that the source topic number and the target topic number are not necessarily to be the same as the topic distributions are used differently in the proposed NMT model. The source topic distribution is appended to each word in the encoder, and the target distribution is passed via the RNN in the decoder.

We first compare the system performance on the development data. According to Table 2, using topic information can improve system performance over the two baseline systems. The source topic-informed NMT (40) system can gain absolute improvements of 1.97 (5.8% relative) and 1.06 (3.1% relative) BLEU scores compared to the SMT and NMT baseline systems, respectively. When using the topic information on the target side, we can observe absolute BLEU score improvements of 2.89 (8.6% relative) and 1.98 (5.8% relative) compared to the SMT and NMT baseline systems, respectively, which is the best-performed system on the development data. The topic-informed NMT (40, 10) system can only gain absolute 0.53 (1.5% relative) and 1.44 (4.3% relative) improvements compared to the SMT and NMT baseline systems, respectively.

In Table 2, significant improvements can be observed on the test data. There is a gain of 0.41 (absolute, 1.8% relative) and 0.83 (absolute, 2.7% relative) BLEU scores compared with the NMT baseline systems when the topic information is employed on the source side, on the NIST 2004 and NIST 2005 data sets,

[4] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-bleu.perl

Source:	他/He 当即/immediately 被/sent 送往/to 医院/hospital 抢救/emergency treatment
Translation 1:	He was sent to hospital for **rescue**
Translation 2:	He was sent to hospital for **emergency treatment**

Source:	过半/Over half 英国人/British 不/disagree 赞同/disagree 政府/government 支持/support 美/US 打/attack 伊拉克/Iraq
Translation 1:	The British people **does not agree** to support United States to **fight** Iraq
Translation 2:	The British people **disagree** with the government 's support for US **war against** Iraq

Figure 4: The examples shows our observations that better word choices can be made in the topic informed NMT. Translation 1 is produced by the baseline NMT, and Translate 2 is the topic-informed NMT output.

Source:	印尼/Indonesia 国会/parliament **议长/speaker** 出庭/stands 受审/trial
Translation 1:	UNK of Indonesia 's congress in court
Translation 2:	Indonesian parliament **speaker** is trial

Source:	**卡伊达/Qaida** 组织/Qaida 领导层/leadership 也/also 开始/began to **活跃/active** 起来/be 。/.
Translation 1:	The leadership of the UNK city began to UNK .
Translation 2:	The leadership of the UNK organization has begun to be **active** .

Figure 5: The examples shows our observations that less number of *UNK* can be produced in the topic informed NMT. Translation 1 is produced by the baseline NMT, and Translate 2 is the topic-informed NMT output.

respectively. Furthermore, we find further absolute BLEU score improvements of 0.67 (absolute, 1.9% relative) and 1.38 (absolute, 4.4% relative) on the target topic-informed NMT (10) system, on the NIST 2004 and NIST 2005 data sets, respectively. The best-performing system on the test data is the topic-informed NMT (40,10) system, which achieves 35.91 and 32.79 BLEU scores on the NIST 2004 and NIST 2005 data sets, respectively. These results are 1.15 (absolute, 3.3% relative) and 1.67 (absolute, 5.4% relative) higher compared with the NMT baseline systems. Overall, the topic-informed NMT system significantly improves upon the baseline translation performance.

5.3 Observations

As we described earlier, by allowing topic information into the decoder, topic-consistent behaviour can be maintained, and consequently better translations can be produced. We find that the translations produced by the baseline NMT system is more fluent compared to SMT. However, two types of errors are still commonly made.

The first type of error produced in NMT translations is the lexical selection error. We find that when words can be translated by NMT, some of the translations are not appropriate in the context, i.e. words are not suitable for the domain. For example, in the baseline NMT translation examples in Figure 4, 抢救/"emergency treatment" is translated into *rescue* and 打/attack is translated into *fight*. In topic-informed NMT translations in Figure 4, the source input sentences and previous translations, e.g. *hospital* in the first example, and *British*, *government* and *US* in the second example, can give strong indications to the decoder about the current topics, namely "*Hospital*" and "*Politics*" topics in the two examples, respectively, that better word choices can thus be produced in the translations.

The second type of error that can be observed in the NMT translations is that the lexical coverage is low. Some of the source words are translated as *UNK* even though the correct translations can be found in the target vocabulary list. Comparing the *UNK* numbers produced by the baseline NMT and the topic-informed NMT, we observe that the number of *UNK* tokens has been reduced in the topic-informed NMT translations, as presented in Table 2. Interestingly, we also find that the topic-informed NMT system tends to produce more words in translations, namely 50,913 and 34,695 words in NIST 2004 and NIST 2005, respectively. In contrast, the NMT baseline produces 44,552 and 30,558 words in NIST 2004 and NIST 2005, respectively. In conclusion, topic-informed NMT can reduce the *UNK* number even when more words appear in the translation outputs. We think the reason for this the topic distribution of the *UNK* token favours to one particular topic. If the source inputs do not belong to the

	Baseline NMT	Topic-Informed NMT
NIST 2004	2.3%	1.9%
NIST 2005	2.7%	2.3%

Table 3: The percentage of *UNK* tokens produced in translation outputs by baseline NMT and topic-informed NMT systems.

same topic in which *UNK* appears, the *UNK* token will have less chance to be chosen as the translation output. Therefore, other word choices than *UNK* can be made and the overall *UNK* number is reduced. Inspecting the translation examples in Figure 5, 议长/speaker and 活跃/active fail to be translated by the baseline NMT system. However, topic-informed NMT is able to use the known topic information, either from the source sentences or previous translations, to produce correct translations. For example, the source words 议长/speaker and 活跃/active are translated into *speaker* and *active*, respectively. The source word 卡伊达/Qaida does not appear in the parallel training data, so both systems produce *UNK* in this case.

Figure 6 compares the word alignments produced by the two systems. In this example, we can find that the word alignments of topic-informed NMT can give more weights (α_{ij} in Equation (6)) to the correct aligned words, which consequently indicates to the decoder to pay more attentions on the correct source works to translate.

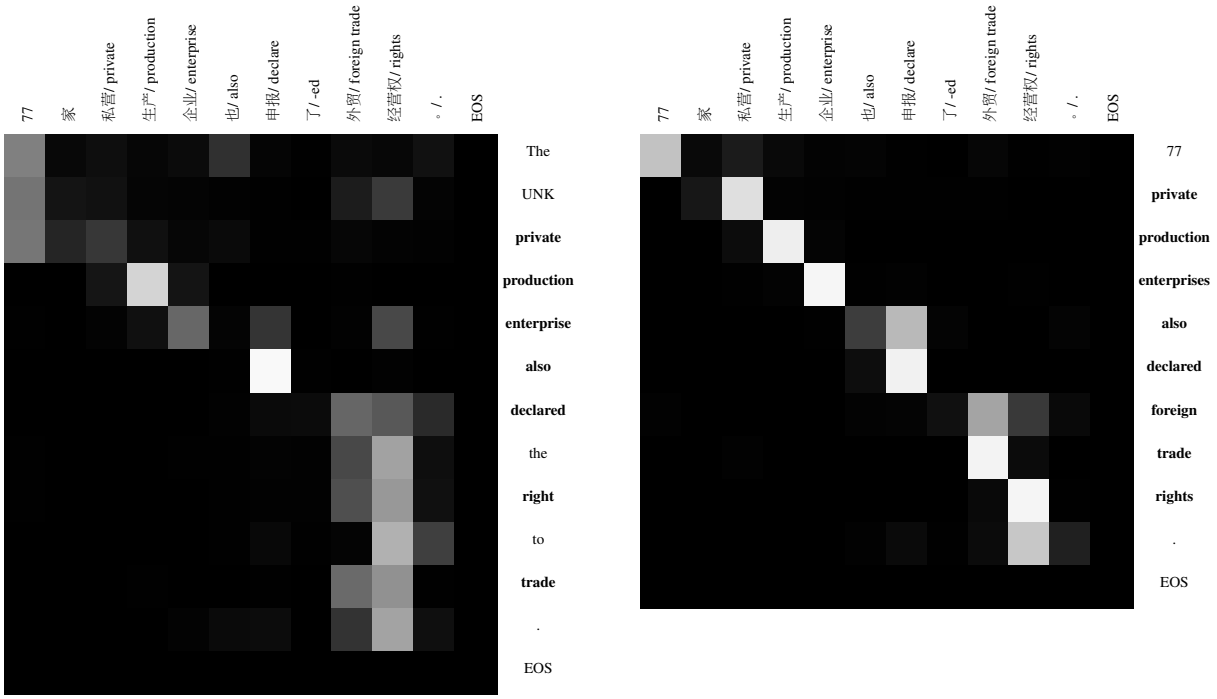

Figure 6: The comparison of alignments generated without (left) and with (right) topic knowledge. The example is chosen from the development data, where the x-axis sentence is the source training sentence (Chinese), and the y-axis sentence is the target training sentence (English). In the heatmap, we use grayscale colour schemes, where black means the word alignment probability is low, and white means the word alignment probability is high.

Our baseline NMT system contains 58,427,521 parameters to learn. Since we concatenate the source topic information and use a GRU network to store the target topic information, the topic-informed NMT system contains 817,984 more parameters to learn. During our experiments, we see that all of the trained NMT models (baseline NMT and topic-informed NMT systems listed in Table 2) produce their best-performing models between 280,000 and 320,000 in terms of update numbers. The baseline NMT and the topic-informed NMT systems require 284,000 and 289,000 updates to learn the best-performing models on the development data, respectively.

6 Conclusion

NMT has a lot of potential as a new approach to MT. In this paper, we present a novel approach of integrating topic knowledge into the existing NMT architecture. Through our experiments, we show that translation quality can be improved. We demonstrate that our topic-informed NMT can achieve 1.15 and 1.67 absolute improvements in BLEU score on two different test sets. The experimental results not only demonstrate the effectiveness of the proposed model, but also show an approach to enrich the representation of the context vector produced by the encoder and decoder. We show that introducing topic information in NMT can produce translations with better lexical selection, and a lower number of UNKs. We give concrete examples to support our observations. Furthermore, we argue that better word alignments can also be learned which consequently benefits translation quality.

In the future, we want to experiment on other topic learning approaches, e.g. LSA or HTMM, or jointly train neural topic model (Cao et al., 2015) and translation. We are also interested in further exploring the correlation between NMT performance and the quality of the topic modeling itself.

Acknowledgments

We would like to thank the three anonymous reviewers for their valuable and constructive comments and the Irish Center for High-End Computing (`www.ichec.ie`) for providing computational infrastructures. The ADAPT Centre for Digital Content Technology (`www.adaptcentre.ie`) at Dublin City University is funded under the Science Foundation Ireland Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proceedings of the International Conference on Learning Representations 2015, ICLR 2015*, San Diego, California, USA.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022.

Ziqiang Cao, Sujian Li, Yang Liu, Wenjie Li, and Heng Ji. 2015. A Novel Neural Topic Model and Its Supervised Extension. In Blai Bonet and Sven Koenig, editors, *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015*, pages 2210–2216, Austin, Texas, USA.

David Chiang, Steve DeNeefe, and Michael Pust. 2011. Two Easy Improvements to Lexical Weighting. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, pages 455–460, Portland, Oregon, USA.

David Chiang. 2005. A Hierarchical Phrase-Based Model for Statistical Machine Translation. In *ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, pages 263–270, University of Michigan, USA.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1724–1734, Doha, Qatar.

Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *Computing Research Repository (CoRR)*, abs/1412.3555.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

Vladimir Eidelman, Jordan L. Boyd-Graber, and Philip Resnik. 2012. Topic Models for Dynamic Translation Model Adaptation. In *The 50th Annual Meeting of the Association for Computational Linguistics*, pages 115–119, Jeju Island, Korea.

Michel Galley and Christopher D. Manning. 2008. A Simple and Effective Hierarchical Phrase Reordering Model. In *EMNLP 2008: 2008 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, pages 848–856, Honolulu, Hawaii, USA.

Mercedes García-Martínez, Loïc Barrault, and Fethi Bougares. 2016. Factored Neural Machine Translation. *Computing Research Repository (CoRR)*, abs/1609.04621.

Amit Gruber, Yair Weiss, and Michal Rosen-zvi. 2007. Hidden Topic Markov Models. In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics (AISTATS-07)*, volume 2, pages 163–170.

Çaglar Gülçehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loïc Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On Using Monolingual Corpora in Neural Machine Translation. *Computing Research Repository (CoRR)*, abs/1503.03535.

Eva Hasler, Barry Haddow, and Philipp Koehn. 2012. Sparse Lexicalised Features and Topic Adaptation for SMT. In *2012 International Workshop on Spoken Language Translation, IWSLT*, pages 268–275, Hong Kong.

Eva Hasler, Barry Haddow, and Philipp Koehn. 2014. Dynamic Topic Adaptation for SMT using Distributional Profiles. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 445–456, Baltimore, Maryland, USA.

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, Phoenix, Arizona, USA.

Kenneth Heafield. 2011. KenLM: Faster and Smaller Language Model Queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197, Edinburgh, Scotland.

Philipp Koehn, Amittai Axelrod, Alexandra Birch, Chris Callison-Burch, Miles Osborne, and David Talbot. 2005. Edinburgh System Description for the 2005 IWSLT Speech Translation Evaluation. In *2005 International Workshop on Spoken Language Translation*, pages 68–75, Pittsburgh, PA, USA.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, Prague, Czech Republic.

Philipp Koehn. 2004. Statistical Significance Tests for Machine Translation Evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing , EMNLP 2004, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 388–395, Barcelona, Spain.

Thang Luong, Ilya Sutskever, Quoc V. Le, Oriol Vinyals, and Wojciech Zaremba. 2015. Addressing the Rare Word Problem in Neural Machine Translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, pages 11–19, Beijing, China.

Daniel Marcu and William Wong. 2002. A Phrase-Based, Joint Probability Model for Statistical Machine Translation. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 133–139, University of Pennsylvania, Philadelphia, PA, USA.

Tomas Mikolov and Geoffrey Zweig. 2012. Context Dependent Recurrent Neural Network Language Model. In *2012 IEEE Spoken Language Technology Workshop (SLT)*, pages 234–239, Miami, Florida ,USA.

Franz Josef Och and Hermann Ney. 2000. Improved Statistical Alignment Models. In *38th Annual Meeting of the Association for Computational Linguistics*, pages 440–447, Hong Kong, China.

Franz Josef Och. 2003. Minimum Error Rate Training in Statistical Machine Translation. In *41st Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, pages 160–167, Sapporo Convention Center, Japan.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, Philadelphia, PA, USA.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta.

Rico Sennrich and Barry Haddow. 2016. Linguistic Input Features Improve Neural Machine Translation. In *Proceedings of the First Conference on Machine Translation, WMT 2016, colocated with ACL 2016*, pages 83–91, Berlin, Germany.

Jinsong Su, Deyi Xiong, Yang Liu, Xianpei Han, Hongyu Lin, Junfeng Yao, and Min Zhang. 2015. A Context-Aware Topic Model for Statistical Machine Translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, pages 229–238, Beijing, China.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. In Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014*, pages 3104–3112, Montreal, Quebec, Canada.

Huihsin Tseng, Pichuan Chang, Galen Andrew, Daniel Jurafsky, and Christopher Manning. 2005. A Conditional Random Field Word Segmenter for Sighan Bakeoff 2005. In *In Fourth SIGHAN Workshop on Chinese Language Processing*.

Xinyan Xiao, Deyi Xiong, Min Zhang, Qun Liu, and Shouxun Lin. 2012. A Topic Similarity Model for Hierarchical Phrase-based Translation. In *The 50th Annual Meeting of the Association for Computational Linguistics*, pages 750–758, Jeju Island, Korea.

Matthew D. Zeiler. 2012. ADADELTA: An Adaptive Learning Rate Method. *Computing Research Repository (CoRR)*, Volume: abs/1212.5701.

Min Zhang, Xinyan Xiao, Deyi Xiong, and Qun Liu. 2014. Topic-based Dissimilarity and Sensitivity Models for Translation Rule Selection. *Journal of Artificial Intelligence Research*, 50(1):1–30.