Minimum Risk Training in improving domain robustness of Neural Machine Translation

Chaojun Wang

Master of Science
Artificial Intelligence
School of Informatics
University of Edinburgh
2019

Abstract

Although Neural Machine Translation (NMT) improves state-of-the-art Machine Translation significantly, it still underperforms on some conditions than Statistical Machine Translation (SMT). One of the conditions is out-of-domain translations. Previous works mainly focus on improving the out-of-domain translation of NMT systems by leveraging out-of-domain monolingual data. However, we believe that improving model generalisation ability to unseen domains, namely domain robustness is also important since monolingual corpus would be unavailable or the domain that NMT systems would encounter in actual use would be unpredictable. Therefore, this study mainly aims to explore the approach to improving domain robustness of NMT systems.

A previous study showed that 'hallucination translation' (translations that are fluent but unrelated to source) is common in out-of-domain translation. According to theoretical analysis, we claim that exposure bias derived by token-level training objective would be one of the reasons that result in hallucinations. Therefore, we propose to use Minimum Risk Training (MRT), a sentence-level training objective, to alleviate exposure bias and reduce the hallucinations, and thereby, improve domain robustness. The experiments on the domain-specific dataset from OPUS repository demonstrated the effectiveness of MRT and our best model also exceeds SMT and achieves the state-of-the-art. Our hypotheses about the correlation between exposure bias and hallucination and the relation between hallucinations and domain robustness are also supported in a series of analysis experiments.

Moreover, inspired by the results of the analysis experiments, we speculate and by experiments provide the evidence for the correlation between exposure bias and the phenomenon that increasing beam size does not improve translation quality. MRT fine-tuning would mitigate this phenomenon and extended the optimal beam size of NMT systems.

Acknowledgements

First of all, I would like to thank my supervisor, Dr Rico Sennrich, for his valuable and patient guidance during the project, from which I have learned a lot about theoretical knowledge, critical thinking and academic writing.

Next, I would like to thank my families for selfless support and help, my parents for giving me a warm family and endless love, and for supporting me to do what I would like to do.

Lastly, I want to thank my friends, who accompanied me in my study and growth.

Table of Contents

Chapter 1

Introduction

Machine Translation (MT) which aims to use of the algorithm to automatically translate the text from one language to another language is an important application in Natural Language Processing. Recently, Neural Machine Translation (NMT), which is a neural network-based MT [33], has significantly improved state-of-art in machine translation. However, NMT does not perform well under all of the conditions. One such condition is out-of-domain translation [1]. However, Statistical Machine Translation (SMT), which is the dominant MT algorithm before NMT, exhibits a better performance on out-of-domain translation than NMT. Therefore, Koehn and Knowles [15] pointed out that the out-of-domain translation is one of the key challenges in NMT.

Most of the existing works aim to improve the out-of-domain translation of NMT systems by leveraging out-of-domain monolingual data and in-domain parallel data, namely domain adaptation [7] and have achieved good results [8, 29, 13]. However, the out-of-domain monolingual data may not exist for some language pairs, and the domains that will be encountered in actual use would be unpredictable. Therefore, to improve user satisfaction and model reliability, the ability of NMT systems that showing good generalisation to unseen domains, namely domain robustness [19] needs to be improved. Our study aims to explore the approach to this issue.

According to the previous study [19], 'hallucination translations' (hallucinations), which mean that the translations that are fluent but unrelated to the source sentence, are more pronounced in the out-of-domain translation than the in-domain translation of NMT systems. Therefore, alleviating the hallucinations would indirectly improve the domain robustness. By thinking of the reasons that result in hallucinations, we spec-

[1] Out-of-domain translation refers to that the model is trained with the training sentences from one domain, but is used to translate the sentences from another domain.

ulate that *exposure bias* [26] would be one of the reasons that lead to hallucinations. Exposure bias refers to a mismatch between training and testing stage of vanilla NMT systems (of which the training objective is a token-level training objective, Maximum Likelihood Estimation (MLE)). More specifically, in the training stage, the model is trained to predict the target token based on ground-truth partial translations, whereas in the testing stage the model translates according to the partial translations predicted by itself. The mismatch would result in error accumulated quickly. Accordingly, with the incorrect partial translations feeding into the model, the probability assigned by the model of the hallucination would be gradually higher than the correct translation, and finally, a hallucination translation would be generated by the model. Therefore, we deduce that if exposure bias could be alleviated, the hallucinations would be reduced and thereby the domain robustness would be improved. In our study, we propose to use one of the sentence-level training objectives, Minimum Risk Training (MRT) [2] as the method to mitigate exposure bias by fine-tuning[3] the baseline model that is pre-trained on token-level MLE training objective. We adopt Transformer [38] as the architecture of the model. By comparing the fine-tuning model with baseline, we would derive the conclusion that whether MRT could improve the domain robustness of NMT systems.

We experiment on German-English translation task of a general dataset (IWLST 2014 [5]) to tune the hyperparameters and a domain-specific dataset (from OPUS repository [35]) to test model's domain robustness. There are five domains of parallel data in this domain-specific dataset. We use one of them as the in-domain data to train the model and the data from the rest four domains as the test data. The results of experiments confirm that MRT is effective in improving domain robustness of NMT systems, especially when the baseline is trained by MLE with label smoothing [34]. Our best model exceeds SMT and achieves the state-of-the-art.

The results of quantitative and qualitative analyses support our theoretical proposition about the correlation between exposure bias and hallucinations, and the correlation between hallucinations and domain robustness. Comparing the out-of-domain with in-domain translation, we further speculate the explanation of the phenomenon that hallucinations occur more in out-of-domain than in-domain translations. We believe that exposure bias still exists in in-domain translation, but the problem caused

[2] a kind of sentence-level training objective that inherently avoids exposure bias derived during token-level training objective in typical NMT systems. MRT has been demonstrated effective in NMT systems [32], thus choosing as the method of our study.

[3] a training paradigm in which the model would share the parameters of the pre-trained model (baseline here) and continue the training with new training objective, or dataset etc.

by exposure bias is more likely to be hidden in in-domain than out-of-domain translation. Moreover, based on the theoretical deduction and empirical study, we find that exposure bias would also relate to the phenomenon that increasing beam size does not consistently improve the translation quality of NMT systems ('beam size contradiction'). MRT fine-tuning would alleviate this phenomenon and increase optimal beam size of NMT systems.

The main contributions of this study are:

- We implement MRT training objective[4] based on Nematus toolkit [28][5] and confirm that it yields improvement on a standard dataset over a strong Transformer baseline.

- We demonstrate the effectiveness of MRT in improving domain robustness of the NMT system through experiments, and our best model achieves state-of-the-art on the German-to-English translation task.

- Through quantitative and qualitative analyses, we find the evidence for the hypotheses that exposure bias would breed hallucinations and resulting hallucinations would deteriorate the domain robustness of NMT systems. Meanwhile, we provide an original method to test exposure bias of NMT systems.

- Combined experiments results and theoretical analysis, we speculate that exposure bias would also cause the problem of 'beam size contradiction' and the results of the subsequent experiments support our hypothesis.

The rest of this dissertation is structured as follows. Chapter 2 will firstly introduce the background knowledge about token-level training objective, label smoothing, the derivation of exposure bias. Our speculation that exposure bias would result in hallucinations will also be elaborated. Then, previous studies about exposure bias and domain in NMT will be discussed. Chapter 3 will describe the principle of Minimum Risk Training. Chapter 4 will provide the comparing of domain robustness between baseline and MRT fine-tuning model by experiments. The corresponding datasets, experiment settings will also be described. Chapter 5 will analyse the reason behind the results of the experiments in Chapter 4 to verify our hypothesis. Finally, the summary of the study and possible future work will be concluded in Chapter 6.

[4] implementation released at: https://github.com/zippotju/nematus-MRT

[5] https://github.com/EdinburghNLP/nematus

Chapter 2

Background

This chapter firstly introduces the principles of the standard end-to-end NMT model trained with token-level training objective Maximum Likelihood Estimation (MLE) and its updated training objective, MLE with label smoothing (MLELS). Both training objectives would be used as the baseline of the domain robustness experiments. Next, we explain the derivation of exposure bias in the end-to-end NMT model trained with token-level training objective and our speculation of how exposure bias would relate to the hallucination translation problem. The studies about the solutions to the exposure bias and its resulting practical problem are also discussed briefly. Finally, we discuss the relevant domain studies in NMT. We point out the difference and the relation between our study and previous studies during the discussion.

2.1 Neural Machine Translation

2.1.1 Training

Denote the source sentence $x_1, x_2, x_3, ..., x_{T_x}$ as X and the target sentence $y_1, y_2, y_3, ..., y_{T_y}$ as Y. The standard end-to-end NMT models the translation probability as follow:

$$P(Y|X;\theta) = \prod_{t=1}^{T_y} P(y_t|X, y_{<t};\theta) \tag{2.1}$$

in which $y_{<t}$ is the partial translation $y_1, y_2, ..., y_{t-1}$ and θ represents the parameters of the NMT model.

The probability of $P(y_t|X, y_{<t};\theta)$ can be modelled by a recurrent neural network (RNN) [33], a convolutional neural network [10] or a Transformer [38]. Please refer to [33, 1, 10, 38] for more details. Because Transformer-based architecture achieves

best performance in NMT so far, we choose to use Transformer as the architecture of the model.

Assume we have a training dataset $D = \{(X^{(n)}, Y^{(n)})\}_{n=1}^{N}$. The standard training objective is to maximise the log likelihood (MLE) of the training dataset:

$$\hat{\theta} = \underset{\theta}{argmax}\{L(\theta)\} \tag{2.2}$$

where log likelihood $L(\theta)$ is:

$$L(\theta) = \sum_{n=1}^{N} \log P(X^{(n)}|Y^{(n)}) = \sum_{n=1}^{N} \sum_{t=1}^{T_y^{(n)}} \log P(y_t^{(n)}|X^{(n)}, y_{<t}^{(n)}; \theta) \tag{2.3}$$

Alternatively, we can set the training objective to minimise the negative log likelihood (NLL), then the NLL is the loss function in MLE training objective. Formally written as (NLL for a translation pair):

$$\mathcal{L}(\theta) = -\log P(X|Y) = \sum_{t=1}^{T_y} -\log P(y_t|X, y_{<t}; \theta) \tag{2.4}$$

In practice, we calculate $-\log P(y_t|X, y_{<t}; \theta)$, the NLL at each time step, using multi-class cross entropy loss:

$$loss\ at\ each\ time\ step = -\sum_{c=1}^{|V|} q_{t,u(c)} \log P(u(c)|X, y_{<t}; \theta) \tag{2.5}$$

where $|V|$ denotes the number of vocabulary in the target language. $q_{t,u(c)}$ and $P(u(c)|X, y_{<t}; \theta)$ separately represent the label probability distribution and the model output probability distribution of the c-th token in the vocabulary at time step t. $u(c)$ is the symbol of the c-th token in the target vocabulary. If we set label probability distribution as follow:

$$q_{t,u(c)} = \begin{cases} 1, & u(c) = y_t \\ 0, & \text{otherwise} \end{cases} \tag{2.6}$$

Then, the resulting cross entropy loss is equivalent to the NLL:

$$-\sum_{c=1}^{|V|} q_{t,u(c)} \log P(u(c)|X, y_{<t}; \theta) = -1 \times \log P(y_t|X, y_{<t}; \theta) = -\log P(y_t|X, y_{<t}; \theta) \tag{2.7}$$

2.1.2 Inference

After training, the resulting probability model would be used to generate the translation according to an input sentence X in the source language. Formally, in theory the

output $\hat{Y} = \underset{Y}{argmax} P(Y|X)$. Therefore, we need to generate every possible sentence Y in the target language and compute the probability $P(Y|X)$ for each sentence and pick the best one. However, the exact search is intractable due to huge search space ($|vocab|^N$ translations for output length N). Therefore, alternatively, people adopt an approximative search algorithm, beam search, to approximately find the best translation $\hat{Y} \overset{beam\ search}{\approx} \underset{Y}{argmax} P(Y|X)$, which balances the quality and speed and is the current default search strategy in NMT. Beam search always keeps top-k-best candidate translation at every time step and expands the translations that have not been stopped by '$< EOS >$' end of sentence token until all of the candidates in the top-k-best list are stopped by '$< EOS >$'. The k is a parameter of beam search named beam width. The larger the k is set, the larger space the algorithm will search, and the algorithm would find the sentence with the higher score, but correspondingly increase the time complexity. The specific implementation of the beam search algorithm, please refer to this Tutorial [20].

2.1.3 Label Smoothing

Equation 2.4 is the basic loss function of MLE training objective. This objective forces the model to distinguish between ground-truth token and the rest of tokens by assigning extreme one or zero predictions. This 'hard' objective would make the model too confident in its prediction and hurt the generalisation performance. Therefore, Szegedy et al. [34] introduced label smoothing, which acts as a regulariser to make the model less confident in its prediction. In practice, label smoothing is implemented by modifying label probability distribution $q_{t,u(c)}$ in Equation 2.6 to:

$$q_{t,u(c)} = \begin{cases} 1-\varepsilon, & u(c) = y_t \\ \frac{\varepsilon}{|V|}, & \text{otherwise} \end{cases} \tag{2.8}$$

where ε is a smoothing parameter which uniformly assigns partial probability of ground-truth token to the rest of tokens. Label smoothing is equivalent to adding KL divergence between a uniform distribution $f = \frac{1}{|V|}$ and the model prediction probability distribution $P(y_t|X, y_{<t}; \theta)$ to the NLL [25] in equation 2.4. Therefore, the loss function of MLE with label smoothing is:

$$\mathcal{L}_{MLELS}(\theta) = -\sum_{t=1}^{T_y} \left(\log P(y_t|X, y_{<t}; \theta) - D_{KL}(f||P(y_t|X, y_{<t}; \theta))\right) \tag{2.9}$$

Here, we use $\mathcal{L}_{MLELS}$ to represent the loss function of MLE with label smoothing to distinguish with basic MLE loss function. Label smoothing has been demonstrated

useful in NMT with Transformer-based architecture [38]. Therefore, we conduct both experiments using MLELS and basic MLE as the baseline separately to investigate whether Label smoothing is also effective in out-of-domain translation.

2.2 Exposure bias

Exposure bias refers to a mismatch between the training and testing stages [26]. More concretely, in equation 2.1 the NMT model is trained to predict the next token y_t given the ground-truth previous tokens $y_{<t}$ as the input. However, at the test time, the resulting model is used to generate the entire sequence by predicting one token at a time and by feeding the generated tokens back as the input at the next time step. The problem exists consistently no matter in basic MLE or MLELS training objective. This discrepancy between the training and testing would yield errors that accumulate quickly along the generated sequence, which is named error propagation [41].

We speculate that the resulting error propagation from exposure bias would be one of the reasons that lead to hallucination translation. Let us take an example to describe our hypothesis. Assume that there is a translation pair with the source sentence $X = A, B, C, D$ and the target sentence $Y = a, b, c, d$. There is also a hallucination translation $Y' = e, f, g, h$ that is unrelated to the source sentence X but fluent and exists in the training set. Figure 2.1 shows the progress of how the model would probably generate hallucination translations. The number in the bracket represents the corresponding probability of each term. At initial, the model would assign a higher probability to a

Reference translation			Hallucination translation		
P(abcd|ABCD) =	P(a|ABCD) ·	(0.32)	P(efgh|ABCD) =	P(e|ABCD) ·	(0.21)
	P(b|ABCD, a) ·	(0.45)		P(f|ABCD, e) ·	(0.49)
	P(c|ABCD, ab) ·	(0.52)		P(g|ABCD, ef) ·	(0.73)
	P(d|ABCD, abc)	(0.65)		P(h|ABCD, efg)	(0.92)
=	0.0487		=	0.0691	

Figure 2.1: Illustrative example of how the NMT model with exposure bias would lead to the hallucination translation

than e based on the source sentence $ABCD$, since e is unrelated to the source sentence. However, when the predicted a and e is fed into the model separately to predict the next token, the error will arise. We speculate the reasons would be based on two facts:

- During training stage the partial translation of the model $y_{<t}$ is always correct, which would result in the fact that the model would probably trust the partial translation regardless of source sentence or pay greater attention to the partial translation than the source sentence.

- Because hallucination $efgh$ exists in the training set, the model was optimised to assign the highest probability on each token given the previous tokens during the training stage.

Therefore, the model would be misled by the incorrect partial translation and have a possibility at some situation to assign a higher probability of $P(f|ABCD,e)$ than $P(b|ABCD,a)$ although ef is unrelated to the source $ABCD$. Then, with the increase of the time step, the model would assign more and more certainty to the hallucination translation and thereby the probability of the hallucination would finally exceed the reference, thus being generated by the model.

Some related works try to solve exposure bias by training the model to generate the next token according to its own predictions. For example, Bengio et al. [2] introduced Scheduled Sampling in neural sequence generation tasks, in which the true previous target token is replaced with a changing probability by a sampled word-level oracle during training. Zhang et al. [42] recently further developed the method by adding sentence-level oracle and noise perturbations on the predicted distribution and further improved the performance. Another direction of attempts is training the model at the sequence level, including using RL-inspired training strategy MIXER [26], structured prediction objective function [32, 9], or with beam search optimisation [39]. Edunov et al. [9] empirically compared several kinds of structured prediction objective functions and beam search optimisation. They found that one of structured prediction objective functions, the expected risk, outperforms other objective functions and beam search optimisation. Therefore, we choose the expected risk or named MRT as the method to conduct the domain robustness experiments. The principle of MRT and how would it solve exposure bias, please refer to Section 3.1.

In fact, there is no consensus of what practical problem would mainly result from exposure bias. Some practitioners ascribe the accuracy drop (the left half part of the translation is more accurate than its right half part in most cases) to exposure bias [17, 43]. Because the model with exposure bias would predict the next token according to the previous translation, incorrect previous translations would degrade subsequent translations and result in the accuracy drop. However, this study [41] found

that although exposure bias would be one of the reasons that lead to the accuracy drop, language branching [3] plays a more important role in causing the accuracy drop. To our knowledge, there exists no study to investigate the relation between exposure bias and hallucination translation problem. We aim to conduct the analysis experiments to investigate it.

2.3 Domain studies in NMT

At present most of the studies in NMT which are relevant to the domain study are concentrated on the domain adaptation. These studies explore the ways to leverage a small amount of parallel data or a large amount of monolingual data in the desired domain to improve the translation of the desired domain and have already been shown very effective for NMT [18, 29, 13]. Different from domain adaptation, domain robustness refers to a property that the NMT system shows good generalisation to unseen domains [19]. Therefore, in the study of domain robustness, we assume that neither parallel nor monolingual data in the desired domain is available. Among a few researches about domain robustness, this research [19] empirically compared serval strategies to strength the domain robustness of the NMT model, including architectural changes (coverage models [37] and reconstruction [36]) which would potentially eliminate the hallucination translation problem, a regularisation technique (subword regularization [16]) with which improvement of domain robustness has been reported, and the defensive distillation which has been demonstrated effective in improving robustness to adversarial examples in image recognition tasks [23]. Their results show that the reconstruction is the most effective approach in improving the domain robustness and an average of 1.8 BLEU improvement on out-of-domain test sets, but still lower than the standard, phrase-based SMT model trained with Moses [14] (11.8 BLEU). Our study follows their methodology [19] that indirectly improve the domain robustness by alleviating hallucination translation. We propose to use MRT to reduce the hallucination translation and indirectly improve the domain robustness.

Chapter 3

Minimum Risk Training

Minimum Risk Training (MRT) is one of the classical structured prediction training objectives that trains the model at the sequence level. In this chapter, we present the MRT strategy that will be used for the domain robustness experiments. In the first section, we will present the principle of MRT and discuss how MRT avoids exposure bias. In the second section, various strategies to generate candidate sentences that are required for MRT would be discussed. The final section sums up the loss functions of serval kinds of training objectives that will be used in the later experiments.

3.1 Principle of MRT

Minimum Risk Training was first proposed by Och [21] to train log-linear models for structured prediction and applied in neural sequence to sequence models by Shen et al. [32]. The basic idea of MRT is to find a set of the model parameter, θ, so that it will minimise the expected loss on the training data. The loss is named *risk* in MRT. Unlike MLE, the loss of MRT (risk) is evaluated on the sequence level. Therefore, MRT would be able to optimise the model toward the evaluation metrics instead of focusing on every single token like in MLE. Optimising toward evaluation metrics is also one of the advantages of MRT in theory, but we mainly focus on its ability to the alleviation of exposure bias.

Formally, given an end-to-end NMT model with the translation probability $P(Y|X;\theta)$ as in Equation 2.1 and a training dataset $D = \{(X^{(n)}, Y^{(n)})\}_{n=1}^{N}$, the expected risk, $R(\theta)$ can be defined as:

$$R(\theta) = \sum_{n=1}^{N} \sum_{Y \in S(X^{(n)})} \frac{P(Y|X^{(n)};\theta)}{\sum_{Y' \in S(X^{(n)})} P(Y'|X^{(n)};\theta)} \Delta(Y, Y^{(n)}) \tag{3.1}$$

Where $S(X^{(n)})$ represents a set of all of the possible candidate translations of source sentence $X^{(n)}$. $\Delta(Y, Y^{(n)})$ denotes the sentence-level loss between the hypothesis Y and reference $Y^{(n)}$. In our work, we use $1 - BLEU(Y, Y^{(n)})$[1] as the loss in MRT. However, since the sample space of $S(X^{(n)})$ is exponentially huge, the expected risk is usually intractable to calculate. We have to sample certain number of candidate translations as a subspace $U(X^{(n)})$ to approximate the full search space. We discuss approaches for generating this subset in Section 3.2.

After generating the subspace $U(X^{(n)})$, the new approximate risk function $\tilde{R}(\theta)$ is:

$$\tilde{R}(\theta) = \sum_{n=1}^{N} \sum_{Y \in U(X^{(n)})} \frac{P(Y|X^{(n)};\theta)^{\alpha}}{\sum_{Y' \in U(X^{(n)})} P(Y'|X^{(n)};\theta)^{\alpha}} \Delta(Y, Y^{(n)}) \tag{3.2}$$

Here besides changing the candidate translation set, a hyperparameter α is also added. This hyperparameter is used to control the sharpness [21] of distribution of the sampled subspace $U(X^{(n)})$ and have been reported having a critical effect on MRT training [32]. After sampling the subspace and calculating the expected risk, the expected risk would be used as the loss function in back-propagation to find the optimal parameters as in MLE.

MRT, same as other classical structured prediction training objectives, can avoid exposure bias. Revising the definition of exposure in Section 2.2, the training and testing mismatch in MLE leads to exposure. From another perspective, during training, the model is only exposed to the training data distribution rather than its own prediction. However, this discrepancy is avoided naturally in MRT. According to the principle of MRT, the model is trained to give the penalty to bad sampled translations and the reward to good sampled translations in order to minimise the risk. Therefore, the training stage involves the inference process or say, the predictions of the model, and the model is trained to discriminate its translation candidates. Accordingly, there is no mismatch between training and testing in MRT, and thereby, exposure bias is avoided.

However, because MRT training is computationally expensive compared with MLE, people usually use MRT to fine-tune the model pre-trained on MLE [32, 9, 6]. Our study follows the same strategy. Hence, the model after fine-tuning with MRT would only gradually alleviate exposure bias instead of complete eliminate it.

[1] BLEU [24] is an automatic evaluation metrics to evaluate the sentence-level loss between model's translation and reference, which is the most commonly used metrics in NMT

3.2 Candidate generation strategies

The MRT training objective is defined over the entire space of possible output translations, which is usually intractable to compute. Therefore, a subset of K candidate sequences $U(X) = Y_1, Y_2, ..., Y_K$ that is generated by the model is used to approximately calculate the expected risk.

3.2.1 Sampling strategies

There are two commonly used sampling strategies in MRT to generate the candidate translations, randomly sampling and beam-search sampling [32, 9]. Beam-search sampling follows the same algorithm of beam search during inference. Based on a source sentence X, the final top-K-best translations generated with the beam search algorithm are used as the K sampled candidates in subset $U(X)$. Please refer to Section 2.1.2 for the introduction of the beam search algorithm.

Algorithm 1 [32]: Randomly sampling strategy

Input: the n-th source sentence in training data $X^{(n)}$, the set of model parameters θ, the limit on the length of a candidate translation l, the limit on the size of sampled space K.

Output: sampled space $U(X^{(n)})$

for $i = 1$ to K **do**
  $Y \leftarrow \emptyset$ // an empty candidate translation
  for $t = 1$ to l **do**
    $y \sim P(y_t|X^{(n)}, y_{<t}; \theta)$ // sample the t-th target token
    $Y \leftarrow Y \cup \{y\}$
    if $y =< EOS >$ **then**
      break //terminate if reach the end of sentence
    end if
  end for
  $U(X^{(n)}) \leftarrow U(X^{(n)}) \cup \{Y\}$
end for

Compared with beam-search sampling, which would generate high probability candidates, randomly sampling would introduce more diverse candidates. Algorithm 1 shows how to build the subset $U(X^{(n)})$ with randomly sampling. The algorithm randomly samples a token y at each time step t according to the current output probability

distribution over $|V|$ tokens in the target vocabulary. Then the sampled token will be fed into the model to compute the output distribution of the next time step. The algorithm will keep sampling until reaching the end of the sentence (line 3-9). Then the algorithm will repeat the process to sample K candidates. Note here to make use of the parallel architectures of GPUs, in actual implementation we would feed a batch of K same source sentences into the model so that the model would generate K candidates in parallel, which would be done very efficiently. Moreover, the sampled subset may exist duplicate candidates, which will be removed to build the final subset. We consider both sampling strategies in the following experiments.

Recent work [9] shows that adding reference translation $Y^{(n)}$ into the subset of $U(X^{(n)})$ would destabilise training. Thus we do not add reference translation while generating the sample space.

3.2.2 Online and offline candidate generation

In the online setting, the candidates will be regenerated whenever we encounter a new input sentence X. Whereas offline setting would only sample once before fine-tuning with MRT based on the pre-trained model. These two methods have been compared on German-English NMT by Edunov et al. [9] and the results showed that online setting outperforms offline counterpart. Therefore, our study only focuses on using online setting to generate candidates. However, because the offline setting is much faster than the online setting, infrequent regenerating would be a good strategy (balancing between the quality and speed) to try in the future work.

3.3 Loss functions in the study

For the convenient of comparison, we conclude loss functions of all of the training objectives that will be used in our study in Figure 3.1. We uniformly define the loss function on a training sample $\{X, Y\}$ for simplicity. There is no much different from these functions defined in previous sections, except expected risk $\tilde{R}(\theta)$ is replaced to $\mathcal{L}_{risk}(\theta)$ to unify the format. $\mathcal{L}_{MLE}(\theta)$ in Equation 3.3 represents the loss function of Maximum Likelihood estimation training objective, negative log-likelihood. $\mathcal{L}_{MLELS}(\theta)$ in Equation 3.4 denotes the loss function of Maximum Likelihood estimation training objective with Label Smoothing. $\mathcal{L}_{risk}(\theta)$ in Equation 3.5 is the loss function in Minimum Risk Training objective, in which $\hat{Y}$ represents the candidate translation of the source sen-

$$\mathcal{L}_{MLE}(\theta) = -\sum_{t=1}^{T_y} \log P\left(y_t \mid X, y_{<t}; \theta\right) \tag{3.3}$$

$$\mathcal{L}_{MLELS}(\theta) = -\sum_{t=1}^{T_y} \left(\log P\left(y_t \mid X, y_{<i}; \theta\right) - D_{KL}\left(f \parallel P\left(y_t \mid X, y_{<t}; \theta\right)\right)\right) \tag{3.4}$$

$$\mathcal{L}_{risk}(\theta) = \sum_{\hat{Y} \in U(X)} \frac{P\left(\hat{Y} \mid X; \theta\right)^{\alpha}}{\sum_{\hat{Y}' \in U(X)} P\left(\hat{Y}' \mid X; \theta\right)^{\alpha}} \Delta\left(\hat{Y}, Y\right) \tag{3.5}$$

Figure 3.1: Sum up of all of the loss functions in our study.

tence X. The NMT model aims to minimise these loss functions to find the optimal set of parameters θ of itself with some numerical solutions, such as gradient descent [27] or Adam [12].

Chapter 4

Experiments

This chapter describes the process and results of the main experiments conducted in our study. The first section mainly introduces the preparation work, relevant settings and plan of the experiments, including datasets that will be used in the experiments, the data preprocessing of the datasets, evaluation metrics and the plans of hyperparameters setting and experiments. The second section reports experiment results and conclusions and analysis deduced from the results.

4.1 Experiments setup

4.1.1 Datasets

4.1.1.1 Domain-specific dataset

For convenient to compare with the previous study [19], we use the same dataset as them, namely domain-specific dataset. This dataset is a German-English corpus which contains five subsets, and each subset has a specific domain. The five domains are *medical, IT, koran, law and subtitles*. The dataset is publicity available from OPUS repository [35][1]. Development and testing set are selected 2000 consecutive sentence pairs from each subset[2].

Because these domains are quite distant, it would be relatively reliable to be used to test the model's domain robustness. In all of the domain robustness experiments, the *medical* domain serves as the training domain, while the remaining four domains are used for testing. Note based on the definition of domain robustness, both the training

[1] http://opus.nlpl.eu/

[2] download dataset: https://files.ifi.uzh.ch/cl/archiv/2019/clcontra/opus_robustness_data.tar.xz

and development set are assumed unavailable for the remaining four domains. Therefore, the hyperparameters search can only be executed with the development set of the *medical* domain.

4.1.1.2 General dataset

- Motivation: From the later experiments, we found that the model fine-tuned with MRT does not show improvement compared with the baseline model on the development set of the *medical* domain. We decide to conduct experiments to search MRT-relevant hyperparameters [3] on a general dataset. Then use the optimised hyperparameters searched on the general dataset to conduct fine-tuning in the domain robustness experiments.

- Description: We choose to use IWSLT'14 [5] German-to-English dataset in the general-domain experiments. IWLST'14 dataset consists of 180 thousand bilingual sentence pairs which derive from manual transcripts of English TED talks into German and is commonly used in the NMT studies [9, 26, 31]. We use the same train/dev/test split as this study [9]. The development set is randomly sampled around 4.5% from the training set, and the testing set is derived by concatenating *tst2010, tst2011, tst2012, dev2010, dev2012* in IWLST'14.

4.1.2 Settings

4.1.2.1 Settings on general dataset

In general dataset, following the data preprocessing in this study [9], we filter the training set by removing (1) the length of the source and target sentences that are less than 1 and longer than 175; (2) the pairs where source length $>$ 1.5 $\times$ target length or target length $>$ 1.5 $\times$ source length. All data is truecased and tokenised with Mose toolkit [14]. Then, the data is segmented into subword symbols using shared Byte-Pair Encoding(BPE) [30] with 30000 merge operations.

For comparison to previous work [9], we report the lowercased, tokenised results evaluated with case-insensitive BLEU, computed with the *multi-bleu.perl* script in Nematus repository on the general dataset.

[3] described detailed in Section 4.1.3

4.1.2.2 Settings on domain-specific dataset

In the domain-specific dataset, we follow the preprocessing in this study [19]. The only two differences from the preprocessing of the general dataset are that (1) training sentences that are longer than 80 instead of 175 are removed ; (2) the BPE merge operations are 32000 rather than 30000.

For comparison to previous work [19], we report the detruecased, detokenised results evaluated with case-sensitive BLEU [24] ,computed with the *multi-bleu-detok.perl* script in Nematus repository on domain robustness experiments. Unless specified, we consistently use the beam size of 12 to do the translation in the following experiments.

4.1.3 Systems

We use Nematus toolkit as the baselines, in which training objectives of MLE and MLE with label smoothing (MLELS) concluded in Section 3.3 based on Transformer architecture have been implemented in the toolkit. We modify Nematus toolkit to include the training objective of MRT based on Transformer architecture.

Because of experiments in the general dataset is only for hyperparameters search, we only use training objective of MLELS as the baseline to pretrain the model, and then the resulting pretrained model with the best development set performance is fine-tuned with MRT. The hyperparameters of the baseline model are almost same as this work [9]. After getting the baseline, we conduct four MRT-relevant hyperparameters search during fine-tuning.

Firstly, two sampling strategies, beam search sampling and randomly sampling, to generate candidate translations are both experimented. Secondly, during fine-tuning the model, we use the fixed learning rate rather than Transformer scheduled learning rate [38] as in pretraining. Therefore, the learning rate is another hyparameter to tune. Thirdly, the batch size which controls the number of training samples of an iteration interacts with learning rate, thus also being tuned. Lastly, we search over the sharpness parameter α in Equation 3.2. The number of sampled candidate translation is also an adjustable hyperparameter. However, because both previous studies [32, 9] reported that with the increase of the candidate set size, the performance consistently increases, we do not tune the candidate set size and fix it to 4 for all of the experiments. The best hyperparameters with beam search sampling strategy and randomly sampling strategy are separately searched and will be separately used in the domain robustness experiments. The detailed information on the baseline's and MRT-relevant hyperparameters

are list in Appendix A.

After finding the optimised MRT-relevant hyperparameters, we start the domain robustness experiments. We use both MLE and MLELS as the baseline. The hyperparameters of the baselines (except 'label smoothing' hyperparameters, the rest of hyperparameters are same over these two baselines) follows the settings in this study [19], except that the warmup of the Transformer learning schedule is adjusted to 6000 rather than 0 after our limit hyperparameter search. After getting baselines, we keep the rest of the hyperparameters of baselines constant and use the MRT-relevant hyperparameters found in the general-dataset experiments to fine-tune these two baselines.

4.2 Results

4.2.1 Results on general dataset

The results on general dataset are shown in Table 4.1. Edunov et al. [9] compared MLELS and MRT training objective on the same dataset with the standard convolutional seq2seq architecture [10]. Compared with their baseline (MLELS), our Transformer baseline improves almost 2.5 BLEU, which would derive from the stronger modelling ability of Transformer architecture than standard convolutional architecture. Recently, Wu et al. [40] introduced a dynamic convolution based on standard convolutional seq2seq architecture and achieved BLEU score of 35.2 on the same dataset, exceeding our Transformer baseline by 0.48, which provides the evidence for that convolutional-based architecture is still competitive on this task.

Systems	BLEU
ConvS2S (MLELS) [9]	32.23
ConvS2S (MRT) [9]	32.84(+**0.61**)
DynamicConv (MLELS) [40]	**35.2**
our Baseline (MLELS)	34.72
our MRT (beam search sampling)	34.97(+**0.25**)
our MRT (randomly sampling)	35.18(+**0.46**)

Table 4.1: Evaluation results of various systems on IWLST'14 German-English tokenized and lowercased test set.

After fine-tuning with MRT, the model exceeds the baseline by 0.25 with beam

search sampling strategy and 0.46 with randomly sampling, which is close to the dynamic convolution model. The results show that MRT fine-tuning is still effective based on strong Transformer architecture. Moreover, by comparing the improvement of fine-tuning in previous study [9] (0.61) with the improvement in our study (0.46), we think the increase of the fine-tuning in our study is in an acceptable range since our baseline is higher than theirs, which might limit the improvement of our MRT fine-tuning. Therefore, we believe that our implementation of MRT and our searched MRT-relevant hyperparameters are both qualified to be used in the domain robustness experiments.

As for the two sampling strategies, although randomly sampling performs better than beam search sampling strategy, we still want to investigate the effects of the sampling strategies on MRT training under out-of-domain dataset. Therefore, we store two groups of hyperparameters, one for beam search sampling, one for randomly sampling, to fine-tune baseline model in the domain robustness experiments.

4.2.2 Results on domain-specific dataset

We conduct the experiments with MLELS as the baseline firstly. We find that the beam search sampling strategies still performs worse than randomly sampling strategy in domain robustness experiment. As shown in Figure 4.1, we evaluate the BLEU score of four out-of-domain test sets (*it, koran, law, subtitles*) per 20 MRT iterations and report the average BLEU over these four test sets. From the figure, we can see that after a period of improvement, the out-of-domain translation quality of the beam search sampling strategy (blue line) starts to decrease and gradually becomes worse than baseline (green line). In contrast, the randomly sampling strategy (orange line) exhibits a relatively better improvement. Its performance keeps rising until 1200 iterations, and the line tends to flatten but still exists gently increase. Therefore, we abandon the beam search sampling strategy and only adopt the randomly sampling method in the following experiments and evaluations.

Table 4.2 shows the evaluation results of our models compared with previous works on the domain-specific dataset. Because we cannot use medical development set to do early stopping of MRT fine-tuning, we uniformly evaluate the model after 2000 updates. The first two systems in the table are the baseline (bidirectional deep Recurrent Neural Network trained with MLELS) and the best method to improve the domain robustness, reconstruction, proposed by Müller et al [19]. The third system is the

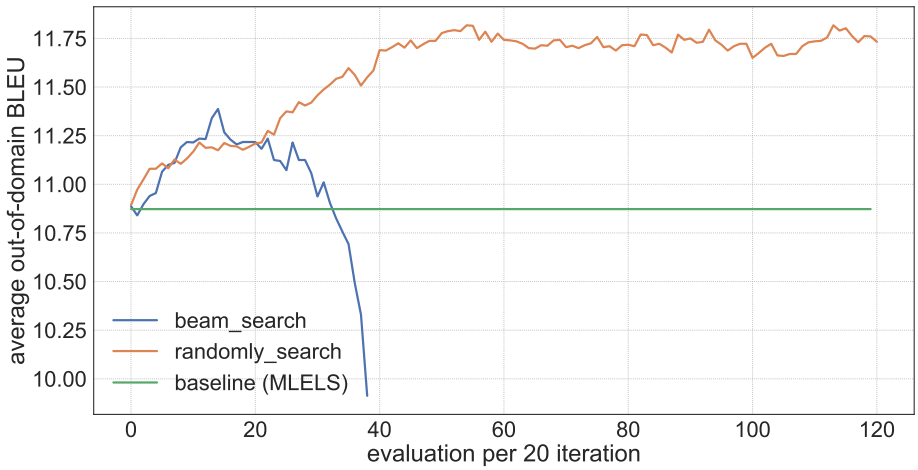

Figure 4.1: Comparison between beam search sampling and randomly sampling strategies of MRT based on MLELS baseline in the domain-specific dataset. The average BLEU score of four out-of-domain test sets are computed and reported in the graph with the increase of the iterations (evaluate per 20 iterations).

phrased-based statistical machine translation system. Our models contain two baselines of MLE and MLELS and their corresponding MRT fine-tuning models.

Firstly, let us compare the baseline of MLE and MLELS. The results show that MLE with label smoothing shows better performance than MLE in both out-of-domain and in-domain translations (9.93, 57.59 of MLE and 10.81, 59.52 of MLELS on out-of-domain and in-domain translations respectively). We deduce that the enhancement of generalisation ability derived from label smoothing would not only improve the model's translation of in-domain data but also benefit model's domain robustness and thereby improve model's performance on out-of-domain translations. Moreover, as the baseline model for the subsequent MRT fine-tune, MLELS seems to provide a good starting point than MLE which would give the model a greater improvement of out-of-domain translations after fine-tuning (0.36 increase initialised with MLE and 1.02 increase initialised with MLELS).

Secondly, compared with previous works, our baseline (MLELS) outperforms their baseline of RNN on out-of-domain translations by 2.13, which would result from the benefit of Transformer architecture. After fine-tuning, our model further improves by 1.02. The growth is lower than their growth of 1.76, but our best model outperforms SMT by 0.03. Although the in-domain translation degrades slightly after fine-tuning (from 59.52 to 58.68), due to the strong baseline, it is still higher than their proposed

model and SMT. Through further observation, we find that although our method outperforms SMT in total, for those domains that original have lousy performance (e.g. *koran* and *subtitles*) our method does not bring significant improvement and underperforms SMT. On the contrary, for domains that are relatively close to the training domain (e.g. *law* and *it*), our method brings more significant improvement and achieves the best performance in all systems.

Systems	in-domain	out-of-domain				
	medical	it	koran	law	subtitles	average
Nematus RNN (baseline) [19]	57.15	13.47	1.11	18.20	1.93	8.68
Reconstruction [19]	58.42	17.50	1.04	20.37	2.85	10.44(+**1.76**)
Moses PBSMT [19]	58.4	21.4	**1.4**	19.8	**4.7**	11.8
our Baseline (MLE)	57.59	17.84	0.80	18.63	2.44	9.93
our MRT init with MLE	58.13	18.42	0.86	19.29	2.58	10.29(+0.36)
our Baseline (MLELS)	59.52	19.13	1.03	20.06	3.00	10.81
our MRT init with MLELS	58.68	**21.75**	1.34	**20.72**	3.49	**11.83**(+1.02)
		(+2.62)	(+0.31)	(+0.66)	(+0.49)	

Table 4.2: Evaluation results on domain-specific dataset German-English detokenized and detruecased test set. The numbers in the bracket denote the amount of increase compared with their corresponding baselines.

From the above results, we may conclude that MRT fine-tune is effective to improve the model's domain robustness (especially in some domains that are close to the training domain). Label smoothing not only benefits the out-of-domain translation but also provides a good initialisation model to be fine-tuned with MRT to improve the model's domain robustness furthermore. Therefore, using MLELS to pretrain and MRT to fine-tune the model would be a relatively better strategy to improve domain robustness. Although MRT fine-tuning will degrade the in-domain translation, it is useful in some situations. For examples, a large amount of available training data belongs to the domain (such as biography) that will not be involved in practical use. Then, MRT fine-tuning would be a good choice to improve the model's performance on translations of other domains that would be met in practical use.

4.2.3 Extra experiments on domain-specific dataset

The results above demonstrate the effectiveness of MRT on improving domain robustness of NMT model. If the reason of the improvement partially or entirely derives from our hypothesis (exposure bias), we speculate that MRT would mitigate exposure bias and somehow make model fewer trusts on the partial translations or assign more attention on source sentence. Therefore, we think that if the MRT fine-tuning is conducted on the dataset from another domain (such as *koran*) instead of *medical*, whether MRT fine-tuning would exhibit more improvement. Because the style of *koran* data is farther away from the *medical* data exposed in MLE, the partial translations are rarely similar with translations usually exposed during MLE. Therefore, the model would learn to assign more attention to source sentence rather than partial translation and thereby would have a stronger ability to revise exposure bias and improve domain robustness.

Therefore, we execute the experiments, in which two baselines are used to implement reasonable comparison. One baseline is that the model is trained with MLELS (in this experiments we only use MLELS training objective to execute pretraining) on the training set that concatenates the training set from *medical* and *koran* domains. The second baseline is pretrained with MLELS on *medical* training set and fine-tuned also with MLELS on *koran* training set. The only difference between the MRT model and the second baseline is that the MRT model is fine-tuned with MRT instead of MLELS. To maximise the performance, we re-preprocess the dataset in which the training set is the concatenation of the training set from *medical* and *koran* domain.

We again evaluate the test set of *it, law, subtitles* per 20 iterations and compute their average BLEU value in Figure 4.2. The figure helps us determine the general tendency of the performance rather than early stopping. From the graph, we can see that the out-of-domain performance increase gradually and exceed the first baseline after 200 iterations with MRT. The second baseline decreases directly after fine-tuning on *koran* data. This indicates that the improvement of domain robustness through MRT fine-tuning depends on characters of sentence-level training objective itself, rather than the training paradigm. Although MRT does not tend to flatten like in Figure 4.1, we still only train it for 2000 iterations (for comparison with experiments in Section 4.2.2) and report the results due to the time limit.

The specific results are shown in Table 4.3. ‘pre-Baseline2’ represents the Baseline2 after pretraining on *medical* set. Compared with Baseline1, the MRT improves the average BLEU by 1, while the average BLEU is improved by 1.02 with MRT fine-

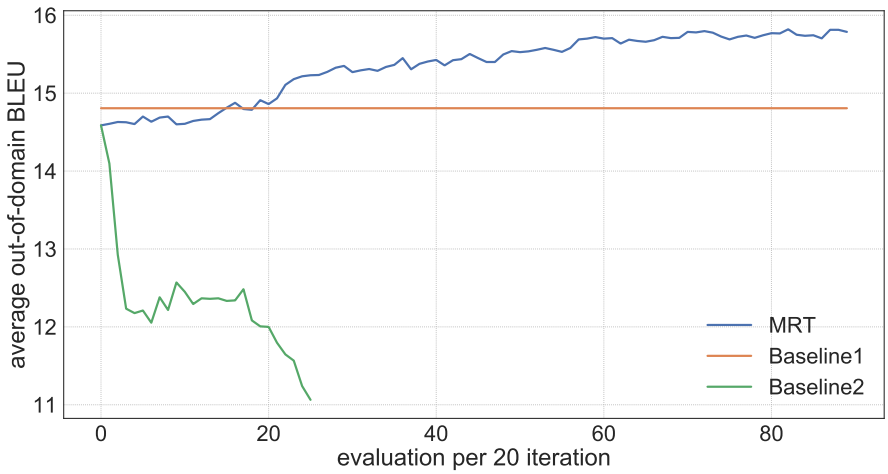

Figure 4.2: The extra experiments in which Baseline1 is trained with MLELS on *medical* and *koran*. Baseline2 is pretrained on *medical* and fine-tuned on *koran* data both with MLELS. MRT is pretrained with MLELS on *medical* data and fine-tuned with MRT on *koran* data. The average BLEU score of three out-of-domain test sets(*it, law, subtitles*) are computed and reported in the graph with the increase of the iterations (evaluate per 20 iterations).

	in-domain	out-of-domain				
Systems	medical	it	koran	law	subtitles	three average
Baseline1 (*medical+koran*)	59.19	20.46	4.07	20.00	3.96	**14.81**
pre-Baseline2 (*medical*)	58.94	20.76	1.02	20.18	2.82	14.59
MRT init with pre-Baseline2	59.12	22.25	1.68	21.38	3.81	**15.81**

Table 4.3: Experiments results in which assuming training sets from two domains are available. Baseline1 is trained with MLE on both datasets. The comparison model is pretrained with MLE on *medical* (pre-Baseline2) and fine-tuned with MRT on *koran*.

tuning on *medical* dataset in Section 4.2.2. The results show that MRT fine-tuning using training data from another domain does not show considerably better ability to improve domain robustness. Although the performance would continue to increase if we continue to fine-tune the model, at least this strategy does not provide an efficient (fast and good) way to improve domain robustness.

Chapter 5

Analysis

Although the results of domain robustness experiments show the effectiveness of our proposed method, it is still not clear whether the reason behind the improvement is consistent with our hypothesis. Therefore, we aim to analyse this quantitative and qualitatively in this chapter by conducting a series of auxiliary experiments.

Figure 5.1 is the logic chain of our hypothesis. Therefore, the first section investigates the hypothesis 1 ('hyp1') that the improvement of domain robustness results from the alleviation of hallucination, and the second aims to explore the hypothesis 2 ('hyp2') that the alleviation of exposure bias leads to the alleviation of hallucination. We think the fact that MRT fine-tuning would alleviate exposure bias is the inherent character of MRT and therefore we treat it as an objective fact rather than our hypothesis. Furthermore, according to the analysis results of the second section, we speculate that exposure bias would relate to 'beam size contradiction' problem. Therefore, we conduct experiments to verify our speculation in Section 5.3.

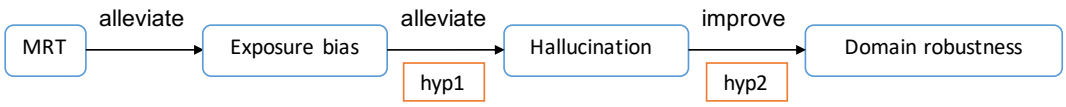

Figure 5.1: Logic chain of our study in which we deduce that MRT fine-tuning would improve the domain robustness according to two hypotheses.

5.1 Hallucination analysis

To verify our hypothesis, we manually evaluate the proportion of hallucination translations of baseline and fine-tuning models on the out-of-domain test sets. We evalu-

ate both MLE and MLELS baseline and their corresponding 2000-updates MRT fine-tuning models (four systems the same as reported in Table 4.2). The proportions of hallucinations on in-domain test sets of two baseline models are also evaluated to confirm the phenomenon found in the previous study [19].

We approximately evaluate the proportion by evaluating 100 sampled sentences from the corresponding test sets. Firstly, we randomly sample 100 sentences from the four out-of-domain test sets and 100 sentences from *medical* in-domain test set separately. To avoid bias, we mix the translations of 100 out-of-domain sentences from four systems and 100 in-domain sentences from two baselines into one file so that we do not know each translation belongs to which systems and from which domains. Then we annotate these 600 sentences as hallucination translation or not. Note here we strictly obey the definition of hallucination in our hypothesis to implement the annotation, which is the translation that (1) the content is totally different from reference; (2) but fluent and grammatical correct. We take a hallucination translation of our model as an example. The reference is: '*It's really good. Really good.*', but model's translation is: '*Immune system disorders.*'. Finally, we restore the annotated translations to the corresponding models and domains according to the information of the belonging of the translations stored in another file and compute the proportion of hallucinations for each system on out-of-domain and in-domain sentences. The results are shown in Table 5.1

Systems	Out-of-domain PROP (BLEU)	In-domain PROP (BLEU)
Baseline (MLE)	33% (9.93)	2% (57.59)
MRT init with MLE	29% (10.29)	-
Baseline (MLELS)	30% (10.81)	1% (59.52)
MRT init with MLELS	24% (11.83)	-

Table 5.1: Manually evaluated proportion of hallucinations and automatic evaluated BLUE score on the four out-of-domain test sets and in-domain test set of four systems.

By Comparing the hallucinations proportion of out-of-domain to in-domain translations of the same system, we find that the phenomenon (found in previous study [19]) that hallucinations occur more frequently in out-of-domain translations than in in-domain translations is consistent with our results. Considering that the proportions of hallucinations are quite low in baselines, we believe that the phenomenon (more hallucinations in out-of-domain than in-domain translations) would be consistent af-

ter fine-tuning. Hence we do not evaluate the in-domain hallucination proportion of fine-tuned models. Next, let us focus on out-of-domain translations. By comparing the baselines with the fine-tuning models, we observe that the proportion of hallucination reduces with the MRT fine-tuning. Consistent with the improvement of BLEU score on out-of-domain translations, the MRT fine-tuning based on MLELS baseline also leads to more reduction of hallucinations than the fine-tuning based on MLE (decreasing by 20% based on MLELS and 12.1% based on MLE). Therefore, the results would indicate that the improvement of domain robustness partially derives from the reduction of hallucinations. Moreover, we observe that although the BLEU score of MLELS baseline is higher than MRT initialised with MLE baseline, its proportion of hallucinations is not lower. We deduce that the label smoothing would probably not improve the out-of-domain translations by reducing hallucinations, but through other perspectives which would be interested in investigating in the future.

[Reference] *She **found all** of us.*
[baseline] *If their symptoms are the same as yours.*
[MRT] *It has not been **all found**.*
[Reference] *By no means**!** for it would be the **Fire** of Hell!*
[baseline] *General CLASSIFICATION FOR SUPPLY.*
[MRT] *16**!** Is a **flammole**.*

Table 5.2: Case study of some typical change after MRT fine-tuning on out-of-domain translations.

We excerpt some typical change of hallucination translations after fine-tuning in Table 5.2. We can see that although the model still fails to translate the correct content after fine-tuning, its translations are closer to the content of reference than baseline, such as predicate 'found' is correctly expressed in the first group and object 'fire' is expressed with a close word 'flammole' in the second group.

5.2 Uncertainty analysis

Above experiments support the hypothesis that the increase of BLEU score in out-of-domain translations is correlated with the alleviation of the hallucination. Next, we aim to explore whether the alleviation of hallucinations is related to the alleviation of

exposure bias.

As we discussed in Section 2.2, the model with exposure bias would potentially assign too much trust on partial history and hence be misled by the partial history. This results in the fact that translations that often exposed during training would finally be predicted by the model since the probability assigned to these translations would increase faster and faster due to the effect of error propagation. The 'often exposed' translations are equivalent to in-domain sentences in our domain robustness experiments. Hence if the model translates these sentences during out-of-domain translation, these translations are exactly the hallucination translations. Therefore, if MRT fine-tuning as we speculated alleviates hallucination, we should observe that with the fine-tuning going on, the probability assigned to hallucination translations would reduce at each time step. Accordingly, the possibility to generate hallucination decreases and thus, the out-of-domain translations improved.

Inspired by this study [22], in the experiments, we use four German-English parallel datasets. The first two datasets are from the domain-specific dataset which are: (1) *medical* test sets; (2) the collection of four out-of-domain test sets. Let us name them MEDICAL_TRUE and OUT-OF-DOMAIN_TRUE separately. Above two datasets are then reassembled to produce the rest two datasets by: (1) keeping the source sentences of two datasets constant; (2) replacing each target sentence to a randomly sampled target sentence from *medical* test set (the sampled 'fake' target sentence is constrained to have the same length as the sentence that is replaced). Let us name them MEDICAL_FAKE and OUT-OF-DOMAIN_FAKE separately. Given a sentence pair $X = x_1, x_2, ..., x_{T_x}$ and $Y = y_1, y_2, ..., y_{T_y}$, we record the probabilities that the model assigns to the output token at each time step given the source sentence X, which are $P(y_1, |X; \theta), P(y_2, |X, y_{<2}; \theta), ..., P(y_{T_y}, |X, y_{<T_y} \theta)$ in Equation 2.1. Then we compute the average per-token probability of each time step over all of the sentences of each four datasets described above and draw the results in the line chart.

We use resulting baselines and its corresponding MRT fine-tuning models in Section 4.2.2 to execute the following experiments. We first conduct the experiments on a group of fine-tuning models that are based on the MLELS baseline. In Figure 5.2, the results of these models performing on two datasets, OUT-OF-DOMAIN_FAKE and OUT-OF-DOMAIN_TRUE, are drawn in two subplots. These two subplots share the same y-axis and legends. Each line represents the per-token probability with the increase of time steps of the corresponding model. The numbers in the legends denote the number of iterations of MRT fine-tuning. The numbers in the bracket represent the

difference of the out-of-domain translation BLEU compared with the previous model (fewer iterations) in the line chart.

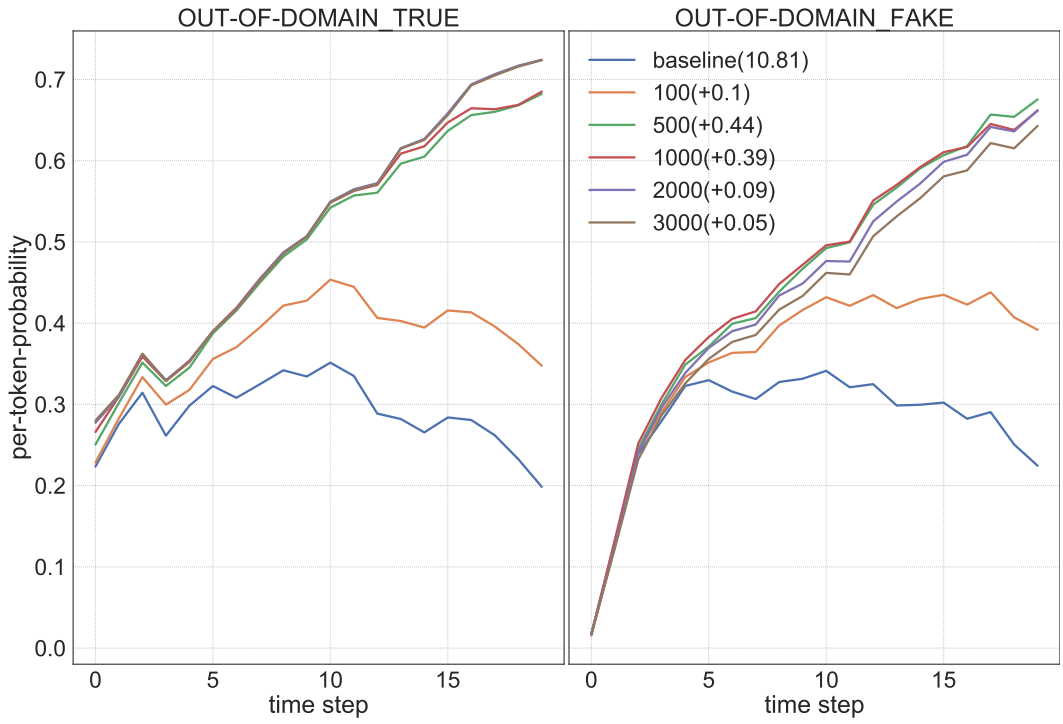

Figure 5.2: Per-token probability with the increase of time step on out-of-domain true and fake target sentences given the correct source sentences. The models include MLELS baseline and its MRT fine-tuning models. The numbers in the legend denote the updates number of fine-tuning and the numbers in the bracket denote the corresponding average BLEU increase on out-of-domain test sets compared with the previous model in the graph.

Because label smoothing would suppress the certainty of the model (principle, please refer to Section 2.1.3), the baseline keeps assigning low probability even for the ground-truth sentence (left subplot blue line). After MRT fine-tuning, without limit of label smoothing the model starts to increase its certainty with the increase of time steps on both OUT-OF-DOMAIN_FAKE and OUT-OF-DOMAIN_TRUE datasets until 500 updates. However, after 500 updates, the model shows the difference between 'fake' and 'true' target sentences. For 'fake' target sentences, the model starts to reduce the assigned probability (see lines from green, red, purple to brown lines in the right subplot), whereas for 'true' target sentences, the model continues to increase the assigned

certainty and tends to flatten after 2000 updates.

If we analyse after 500 updates, we find that the phenomenon would be consistent with our hypothesis. For 'fake' target sentences (right subplot), all of the models assign a very low probability to the first token. However, after feeding serval tokens, the model is misled by the partial translation as we speculated in the hypothesis and starts to increase the certainty assigned to the next tokens. The more fine-tuning the model conducts, the more alleviation of exposure bias and hence the model would not be misled easily as before and thereby reduce the probability assigned to the next tokens of the 'fake' sentences. Moreover, after drawing the per-token-probability lines of the same model performing on two datasets (lines of same colour in Figure 5.2) in the same graph, we find that the gap between the lines of 'true' and 'fake' target sentences becomes larger and larger with the MRT fine-tuning going on ('true' exceeds 'fake' gradually). Concrete plots are in Appendix B (Figure B.1). Therefore, the increase of clearance between 'true' and 'fake' per-token-probability lines would potentially decrease the possibility for the model to generate a hallucination translation and accordingly improve the domain robustness. Therefore, the experiment results support our hypothesis between exposure bias and hallucination and we can at least deduce that the 0.53 increase of BLEU score from 500 updates to 3000 updates is correlated with the alleviation of exposure bias.

As for the reason of the rise from the baseline to the 500 fine-tuning models, we would not give a precise answer because this period of change is a little complicated. We speculate that it would partially derive from the effect of exposure bias since we observe that the interval between 'true' and 'fake' sentences has slightly increased from baseline to 500 updates. It might also result from the increase of the initial probability assigned to the 'true' target sentences, as shown in the left subplot of Figure 5.2 (from blue, orange to green lines). On the condition that the probability assigned to the first token to the 'fake' sentences does not change (shown in right subplot of Figure 5.2), the increase of probability assigned to the first token in 'true' sentence would decrease the possibility that the hallucinations become candidate translations in beam search. Therefore, the translated hallucinations reduce and out-of-domain translation quality increases.

Figure 5.3 shows the results of the same group of models on the MEDICAL_FAKE and MEDICAL_TRUE datasets. By comparing with per-token-probability lines on two out-of-domain datasets, we would deduce the explanations of two phenomenons.

Firstly, hallucination occurs more frequently in out-of-domain translations than in

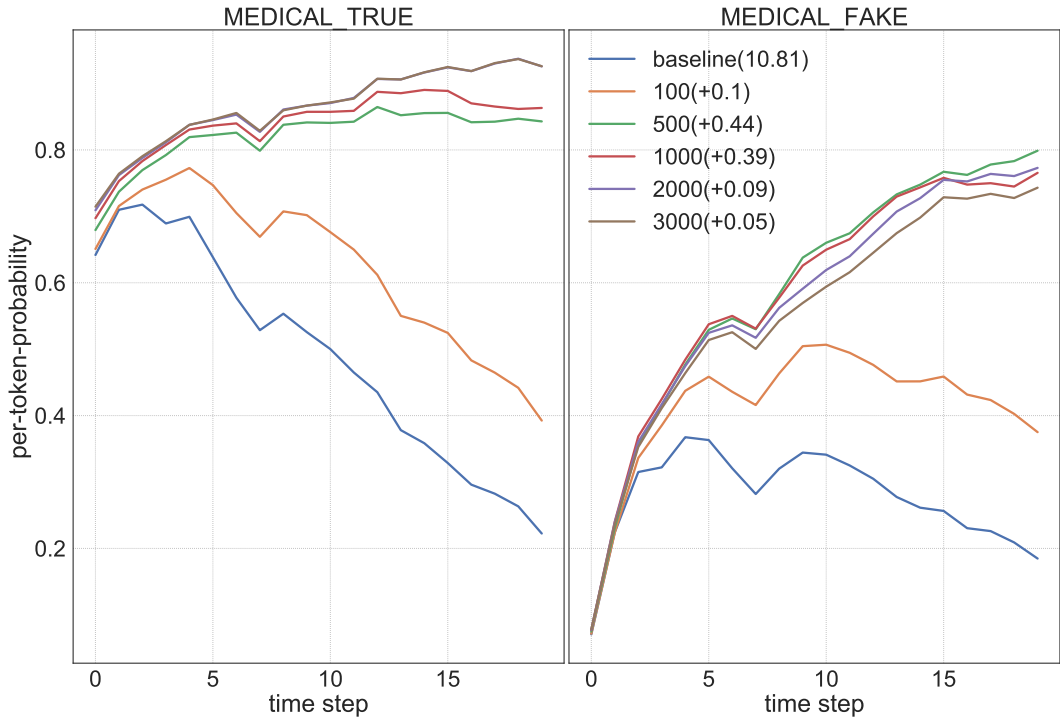

Figure 5.3: Per-token probability with the increase of time step on *medical* true and fake target sentences given the correct source sentences. The models include MLELS baseline and its MRT fine-tuning models.

in-domain translations. From the graph we can see that for all of the models the initial probability assigned to the 'true' sentences are much larger than the 'fake' counterpart (around 0.7 to true and 0.05 to fake) compared with out-of-domain datasets (around 0.25 to true and 0.05 to fake in Figure 5.2). Furthermore, after plotting per-token-probability line of the same model in the same graph, we find that for all of the models from baseline to 3000 updates, the possibilities of the 'true' sentences are always higher than 'fake' counterpart at every time step (graphs in Appendix B, Figure B.2). Therefore, the possibility to generate hallucinations would be greatly reduced compared with the out-of-domain condition (in which not only the probability assigned to the first token between 'fake' and 'true' target sentences is closer but also for models before 500 updates the per-token-probability lines on 'fake' dataset would exceed 'true' dataset). Hence, hallucinations appear much fewer in in-domain than out-of-domain translations (confirmed by the results of the manual evaluation in Table 5.1). However, fewer hallucinations would not mean exposure bias eliminated (probability

assigned by the model to hallucinations still increases in Figure 5.2 (error propagation)), but the problem caused by exposure bias would be hidden by limited beam size of the beam search algorithm. The reason would be that the initial probabilities of the hallucinations are much lower than the correct translations. Therefore hallucinations would have a minuscule probability to be selected in the beam search with limited beam width as the candidates at the first few tokens. Accordingly, error propagation would be avoided, and hallucinations would reduce considerably.

The reasons described above can also explain the other phenomenon, which is why the in-domain translation does not improve after fine-tuning. From the figure, we can see that the MRT fine-tuning does the same function as in out-of-domain translation from the perspective of uncertainty. The probability lines from updates 500 to 3000 are suppressed as in out-of-domain figure (Figure 5.2). However, due to the high probability assigned to the reference translations, a large number of hallucinations is hidden by limited beam size or eliminated by the model. Therefore, the benefit derived from the alleviation of exposure bias would not be shown significantly. Accordingly, the overall statistical result (BLEU) would not increase but decrease because of some unknown reasons.

Next, we experiment on the MLE baseline and its corresponding MRT fine-tuning models. Figure 5.4 shows the results on two out-of-domain datasets. We enlarge the final region of each line because the difference between each line is too narrow. We observe that MRT fine-tuning based on MLE baseline would not suppress the probability assigned to 'fake' translation and improve domain robustness continuously as MLELS-based fine-tuning. It starts to degrade out-of-domain translation (reflected from BLEU score) and re-increase 'fake'-probability after 1000 updates. After drawing lines of the same model in the same graph, we observe that the gap between 'true' and 'fake' lines does not change too much with the MRT fine-tuning (plots also in Appendix B, Figure B.3).

Compared with fine-tuning based on MLELS, the fine-tuning based on MLE exhibits a smaller change of probability assigned to 'fake' translation and less improvement on out-of-domain translation than MLELS counterpart. We analyse the reason would be that the relatively high uncertainty provided by MLELS baseline (like shown in Figure 5.2 bleu line) would make the sampled candidate translations (subspace $U(X^{(n)})$ in Equation 3.2) have higher diversity than candidate translations sampled with MLE baseline. As analysed by Choshen et al. [6], the limitation of the diversity of the sample space would degrade the effect of RL-like training (MRT belongs

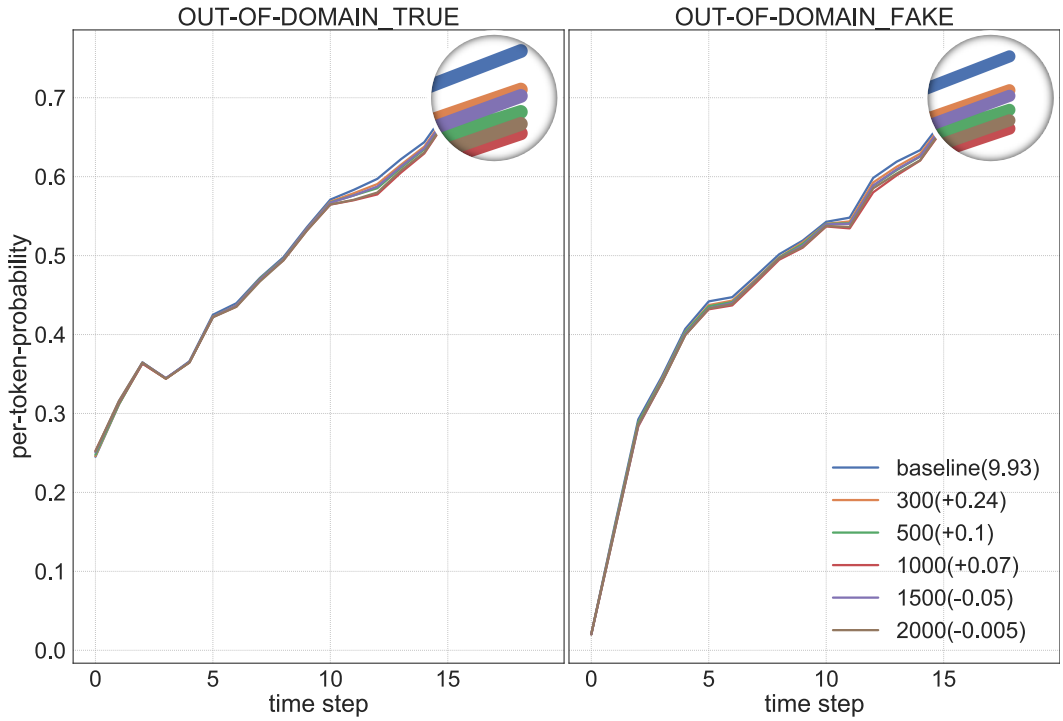

Figure 5.4: Per-token probability with the increase of time step on out-of-domain true and fake target sentences given the correct source sentences. The models include MLE baseline and its MRT fine-tuning models.

to it). Therefore, MLELS baseline would be more beneficial to the subsequent MRT fine-tuning, and thereby the resulting model would assign more uncertainty to 'fake' translations and hence breed more improvement of domain robustness.

Above experiments provide the evidence for the correlation between exposure bias and hallucinations on out-of-domain translations. Moreover, we theoretically deduce the explanation of the phenomenon that hallucinations occur more in out-of-domain than in in-domain translations based on the analysis results. We speculate that the problem caused by exposure bias still exists in in-domain translation, but is more likely to be hidden during in-domain translation than out-of-domain translation.

5.3 Beam size analysis

As described in Section 2.1.2, beam search is an efficient search algorithm to approximately search translation $\hat{Y}$ with the maximum probability given the source sentence X

by exploring a subset of the space of possible translations ($\hat{Y} \overset{beam\ search}{\approx} \underset{Y}{argmax}\, P(Y|X)$). Therefore, increasing the beam size (beam width) allows us to explore a larger set of space of possible translations and hence find the translation with a higher model score. However, Koehn and Knowles [15] pointed out that increasing beam size does not consistently improve translation quality. With the increase of the beam size, the translation quality generally increases first and then decreases with the increase of the beam size after reaching the optimal beam size.

The reasons that lead to this problem would be various. According to the experiments results from uncertainty analysis in Section 5.2, we speculate that one of the reasons that result in this phenomenon on out-of-domain translations would be exposure bias. Our hypothesis is as follow. Firstly, in the case of small beam size, because the probability of "fake" (hallucination) translation is very low in the first few time steps, these "fake" translations would be eliminated from the candidate translations of beam search. Under this situation, the positive effect [1] derived from the increase of search space of beam search would be dominant, and thereby, the translation quality would increase with the increase of beam size. Then if we continuously increase the size of beam search, the possibility, that the 'fake' translations would be kept in the candidate translations until the cumulative probability exceed the 'true' translations, would increase. Therefore, the larger the beam size is, the more 'fake' translations would be generated. When this negative effect exceeds the positive effect derived from the increase of beam size, the overall translation quality will decrease, presented as the decrease of BLEU score. Therefore, with the MRT fine-tuning and the alleviation of exposure bias, the probability assigned to the 'fake' translations will decrease, then under the same beam size, the possibility of generating 'fake' translation by the model would reduce and thereby the problem would be mitigated. The model's optimal beam size would be potentially improved.

To verify our speculation, we experiment on both MLE and MLELS baseline and their MRT fine-tuning after 1000 and 3000 updates respectively (because the fine-tuning models at these updates perform better than 2000 updates as shown in uncertainty analysis experiments in Section 5.2). The models are evaluated with the beam size of 1, 4, 12, 20, 30, 50 separately on the out-of-domain test set. The average BLEU is reported in Figure 5.5

We observe that for all of the models, the performance would increase when the

[1] positive effect refers to that with the increase of beam size, beam search algorithm finds the better translation with a higher model score.

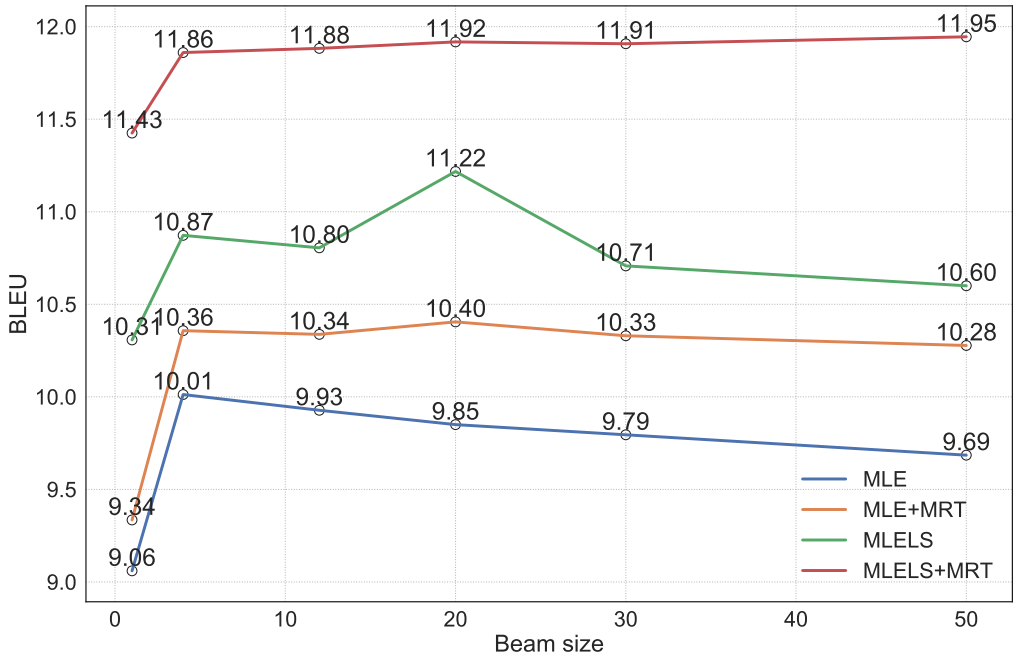

Figure 5.5: Translation quality with varying beam size (including 1, 4, 12, 20, 30, 50) on out-of-domain test sets. The models involve baseline models trained with MLE and MLELS and their corresponding MRT-fine tuning after 1000 and 3000 updates separately.

beam size is less than 4. However, after the beam size of 4, different models show various tendencies. By comparing MLELS and its fine-tuning, we find that the model after fine-tuning (red line) increases its optimal beam size from 20 of the baseline (green line) to 50, which is consistent with our hypothesis. Similarly, the optimal beam size is also extended after fine-tuning based on the MLE baseline (blue and orange lines). Moreover, compared MLE and MLELS, we find that the model trained with label smoothing also show a positive effect on increasing optimal beam size. From the results above, we may conclude that exposure bias would be one of the reasons that lead to this 'beam size contradiction' on out-of-domain translations. After MRT fine-tuning, with the alleviation of exposure bias, the model's optimal beam size would be extended and thus further improving the translation quality of the model.

Chapter 6

Conclusions

6.1 Summary

The objective of the project is to investigate the effect of the training objective of Minimum Risk Training on the domain robustness of Neural Machine Translation systems. More specifically, whether the MRT training objective would improve the performance on out-of-domain translations of NMT systems. Our claim is that the inherent problem of Maximum Likelihood Estimation training objective, exposure bias, would result in the hallucination translation on out-of-domain translations and thereby deteriorate the translation quality of the NMT system on out-of-domain translation. We believe that after fine-tuning with MRT training objective, exposure bias of the NMT model would be alleviated and thereby improve the domain robustness of the NMT systems.

To verify our hypothesis, we compare the performance of out-of-domain translations of the NMT model that is trained with MLE training objective with the model is fine-tuned with the MRT training objective. To compare the effect of label smoothing on domain robustness, we also include the model that is trained using MLE with label smoothing training objective as the baseline. The results of the experiment indicate that the MRT is effective to improve the domain robustness of the NMT systems. The MRT fine-tuning based on the MLELS baseline shows greater improvement than MLE baseline on out-of-domain translations and also achieves the state-of-the-art. Moreover, the results also show that the label smoothing technique benefits not only in-domain translations but also out-of-domain translations.

Our quantitative and qualitative analyses confirm that the reason behind the improvement of domain robustness after MRT fine-tuning is consistent with our hypothesis. Firstly, by manual evaluation of translations generated by baselines and fine-tuning

models on out-of-domain test sets, we find that with the MRT fine-tuning, the proportion of the hallucinations decreases and accordingly BLEU score on out-of-domain translation increases. Therefore, the first analysis demonstrated the correlation between hallucinations and domain robustness. Secondly, by comparing per-token probability assigned to the reference translations and hallucination translations, we find that with MRT fine-tuning, the probability assigned to the second half tokens of hallucinations would be decreased. The experiment phenomenon is consistent with our theoretical deduction about why exposure bias would lead to hallucinations. Hence the second analysis supports the hypothesis of the correlation between exposure bias and hallucinations. Therefore, by the combination of the above two analyses, we may conclude that the partial reason that leads to the improvement of the domain robustness after MRT fine-tuning is consistent with our hypothesis.

Moreover, according to the experiments results of the uncertainty analysis, we firstly theoretically speculate the reason why hallucinations occur more frequently in out-of-domain translations than the in-domain counterpart. We hypothesise that because the probability assigned by the model to in-domain reference translation is much higher than out-of-domain reference translation, most of the hallucination of in-domain translations would be hidden by limited beam size or eliminated by the model. Therefore, hallucinations appear fewer in in-domain translation. Secondly, we theoretically deduce that the phenomenon that the increase of beam size would degrade the translation quality would partially result from exposure bias. Then the subsequent experiments confirm for our speculation. We find that after the alleviation of exposure bias by MRT fine-tuning, the 'beam size contradiction' phenomenon would be mitigated, and the optimal beam size would be extended.

Despite these positive results, there still exist some problems needed to be solved. Firstly, as the results shown in Table 4.2, unlike SMT, the MRT fine-tuning is more effective in improving the translation quality of those relatively good domains in baseline model but less improvement of those domains that perform relatively bad in the baseline model. Next, the uncertainty analyses of NMT model still need more explorations and more in-depth, systematical analyses.

6.2 Future work

As shown in uncertainty analysis, the per-token probability assigned to hallucinations is still too high even after MRT fine-tuning. The phenomenon of the experiments

would indicate that NMT systems are still poor at distinguishing between correct and incorrect partial translations so that the model could not give the incorrect translations punishment. We think the reason would mainly derive from the current token-level training paradigm. Although sentence-level training paradigm like MRT would avoid this problem, its low training speed and unstable character make us have to fine-tune the model that is pre-trained on the token-level training objective. However, our experiments showed that MRT fine-tuning could only alleviate exposure bias rather than resolve it completely. Although this problem would not affect translation quality too much in in-domain translation, under out-of-domain translation, the problem would be easier exposed and breed greater impact on translation quality as we analysed in Section 5.2. Therefore, we need to explore some other methods to mitigate exposure bias further.

Furthermore, we deduce that another inherent problem of typical NMT systems, label bias [4], would also lead to hallucinations and deteriorate domain robustness of the model. The label bias means that the NMT model can only normalise the output at the token-level (at each time step) rather than at the sentence-level. Therefore, the current local normalised model would limit the NMT system to revise its previous decisions according to future inputs. Therefore, the model is more susceptible to error propagation and thus more likely to produce hallucinations. Goyal et al. [11] proposed a way to avoid label bias by implementing global normalisation in NMT architecture. Therefore, it would be a valuable direction in the future to investigate the effect of label bias on domain robustness of NMT systems based on their proposed method.

Bibliography

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

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

[3] Thomas Berg. *Structure in language: A dynamic perspective*. Routledge, 2011.

[4] Léon Bottou. *UNE APPROCHE THEORIQUE DE L'APPRENTISSAGE CONNEXIONNISTE ET APPLICATIONS A LA RECONNAISSANCE DE LA PAROLE*. PhD thesis, 1991.

[5] Mauro Cettolo, Jan Niehues, Sebastian Stüker, Luisa Bentivogli, and Marcello Federico. Report on the 11th iwslt evaluation campaign, iwslt 2014. In *Proceedings of the International Workshop on Spoken Language Translation, Hanoi, Vietnam*, page 57, 2014.

[6] Leshem Choshen, Lior Fox, Zohar Aizenbud, and Omri Abend. On the weaknesses of reinforcement learning for neural machine translation. *arXiv preprint arXiv:1907.01752*, 2019.

[7] Chenhui Chu and Rui Wang. A survey of domain adaptation for neural machine translation. *arXiv preprint arXiv:1806.00258*, 2018.

[8] Anna Currey, Antonio Valerio Miceli Barone, and Kenneth Heafield. Copied monolingual data improves low-resource neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 148–156, 2017.

[9] Sergey Edunov, Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. Classical structured prediction losses for sequence to sequence learning. *arXiv preprint arXiv:1711.04956*, 2017.

[10] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1243–1252. JMLR. org, 2017.

[11] Kartik Goyal, Chris Dyer, and Taylor Berg-Kirkpatrick. An empirical investigation of global and local normalization for recurrent neural sequence models using a continuous relaxation to beam search. *arXiv preprint arXiv:1904.06834*, 2019.

[12] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[13] Catherine Kobus, Josep Crego, and Jean Senellart. Domain control for neural machine translation. *arXiv preprint arXiv:1612.06140*, 2016.

[14] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*, pages 177–180, 2007.

[15] Philipp Koehn and Rebecca Knowles. Six challenges for neural machine translation. *arXiv preprint arXiv:1706.03872*, 2017.

[16] Taku Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates. *arXiv preprint arXiv:1804.10959*, 2018.

[17] Lemao Liu, Masao Utiyama, Andrew Finch, and Eiichiro Sumita. Agreement on target-bidirectional neural machine translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 411–416, 2016.

[18] Minh-Thang Luong and Christopher D Manning. Stanford neural machine translation systems for spoken language domains. In *Proceedings of the International Workshop on Spoken Language Translation*, pages 76–79, 2015.

[19] Mathias Müller, Annette Rios, Philip Williams, and Rico Sennrich. Domain Robustness in Neural Machine Translation. submitted, 2019.

[20] Graham Neubig. Neural machine translation and sequence-to-sequence models: A tutorial. *arXiv preprint arXiv:1703.01619*, 2017.

[21] Franz Josef Och. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 160–167. Association for Computational Linguistics, 2003.

[22] Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. Analyzing uncertainty in neural machine translation. *arXiv preprint arXiv:1803.00047*, 2018.

[23] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 582–597. IEEE, 2016.

[24] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

[25] Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*, 2017.

[26] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*, 2015.

[27] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[28] Rico Sennrich, Orhan Firat, Kyunghyun Cho, Alexandra Birch, Barry Haddow, Julian Hitschler, Marcin Junczys-Dowmunt, Samuel Läubli, Antonio Valerio Miceli Barone, Jozef Mokry, and Maria Nadejde. Nematus: a toolkit for neural machine translation. In *Proceedings of the Software Demonstrations of the*

15th Conference of the European Chapter of the Association for Computational Linguistics, pages 65–68, Valencia, Spain, April 2017. Association for Computational Linguistics.

[29] Rico Sennrich, Barry Haddow, and Alexandra Birch. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*, 2015.

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

[31] Rico Sennrich and Biao Zhang. Revisiting low-resource neural machine translation: A case study. *arXiv preprint arXiv:1905.11901*, 2019.

[32] Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. Minimum risk training for neural machine translation. *arXiv preprint arXiv:1512.02433*, 2015.

[33] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[34] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[35] Jörg Tiedemann. Parallel data, tools and interfaces in opus. In *Lrec*, volume 2012, pages 2214–2218, 2012.

[36] Zhaopeng Tu, Yang Liu, Lifeng Shang, Xiaohua Liu, and Hang Li. Neural machine translation with reconstruction. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[37] Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. Modeling coverage for neural machine translation. *arXiv preprint arXiv:1601.04811*, 2016.

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[39] Sam Wiseman and Alexander M Rush. Sequence-to-sequence learning as beam-search optimization. *arXiv preprint arXiv:1606.02960*, 2016.

[40] Felix Wu, Angela Fan, Alexei Baevski, Yann N Dauphin, and Michael Auli. Pay less attention with lightweight and dynamic convolutions. *arXiv preprint arXiv:1901.10430*, 2019.

[41] Lijun Wu, Xu Tan, Di He, Fei Tian, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. Beyond error propagation in neural machine translation: Characteristics of language also matter. *arXiv preprint arXiv:1809.00120*, 2018.

[42] Wen Zhang, Yang Feng, Fandong Meng, Di You, and Qun Liu. Bridging the gap between training and inference for neural machine translation. *arXiv preprint arXiv:1906.02448*, 2019.

[43] Xiangwen Zhang, Jinsong Su, Yue Qin, Yang Liu, Rongrong Ji, and Hongji Wang. Asynchronous bidirectional decoding for neural machine translation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Appendix A

Hyperparameters

[1]0 for MLE and MRT training objectives; 0.1 of MLELS training objective.

	General dataset	Domain-specific dataset
General hyperparameters		
embedding layer size	512	
hidden state size	512	
tie encoder decoder embeddings	yes	
tie decoder embeddings	yes	
loss function	per-token-cross-entropy(MRT)	
label smoothing[1]	-	-
optimizer	adam	
learning schedule	transformer(constant)	
warmup steps	4000	6000
gradient clipping threshold	1	0
maximum sequence length	100	
token batch size	4096	
beam size(during validation)	5	4
beam size(during testing)	12	
length normalization alpha	0.6	1
encoder depth	6	
decoder depth	6	
feed forward num hidden	1024	2048
number of attention heads	4	8
embedding dropout	0.3	0.1
residual dropout	0.3	0.1
relu dropout	0.3	0.1
attention weights dropout	0.3	0.1
	beam search sampling	randomly sampling
MRT-revelant hyperparameters		
learning rate	0.00003	0.00001
batch size	8192 (tokens)	10 (sentences)
sharpness alpha	0.005	0.005

Table A.1: Configurations of NMT systems used to pre-train and fine-tune over two datasets. Note in general hyperparameters, the items in brackets denote the options that will be used in MRT fine-tuning.

Appendix B

Plots of uncertainty analysis

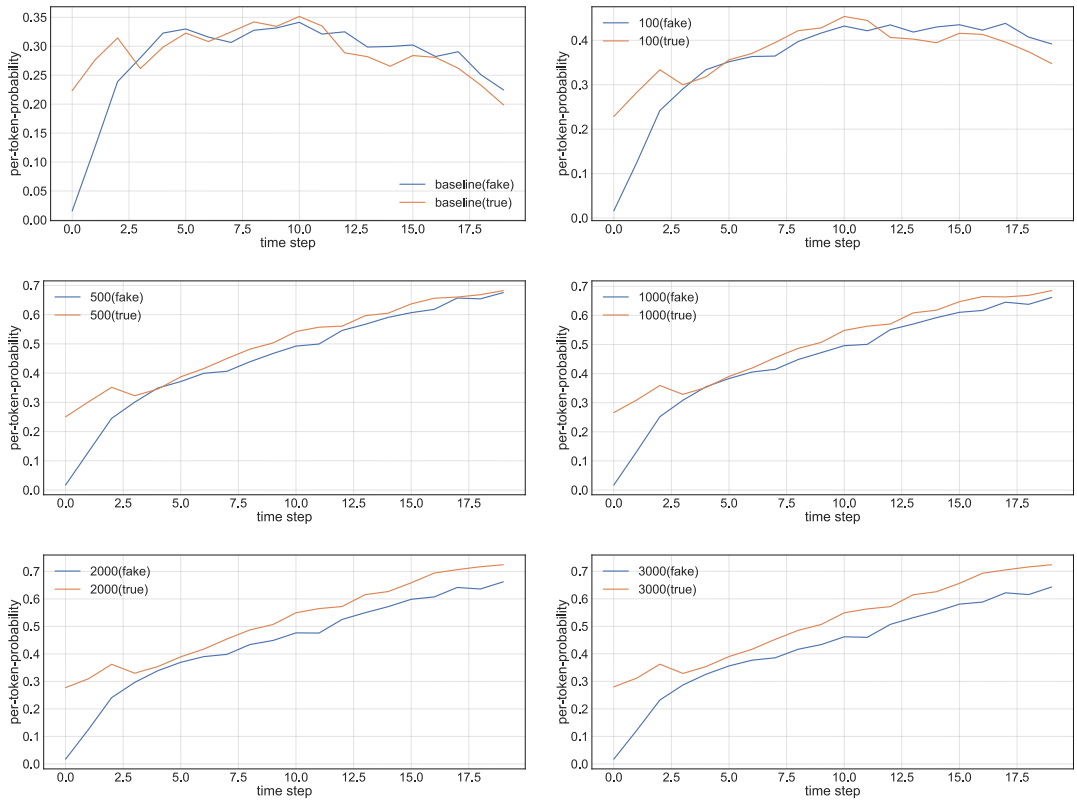

Figure B.1: Per-token probability with the increase of time step on out-of-domain true and fake target sentences given the correct source sentences. The models involve MLELS baseline and its MRT fine-tuning models.

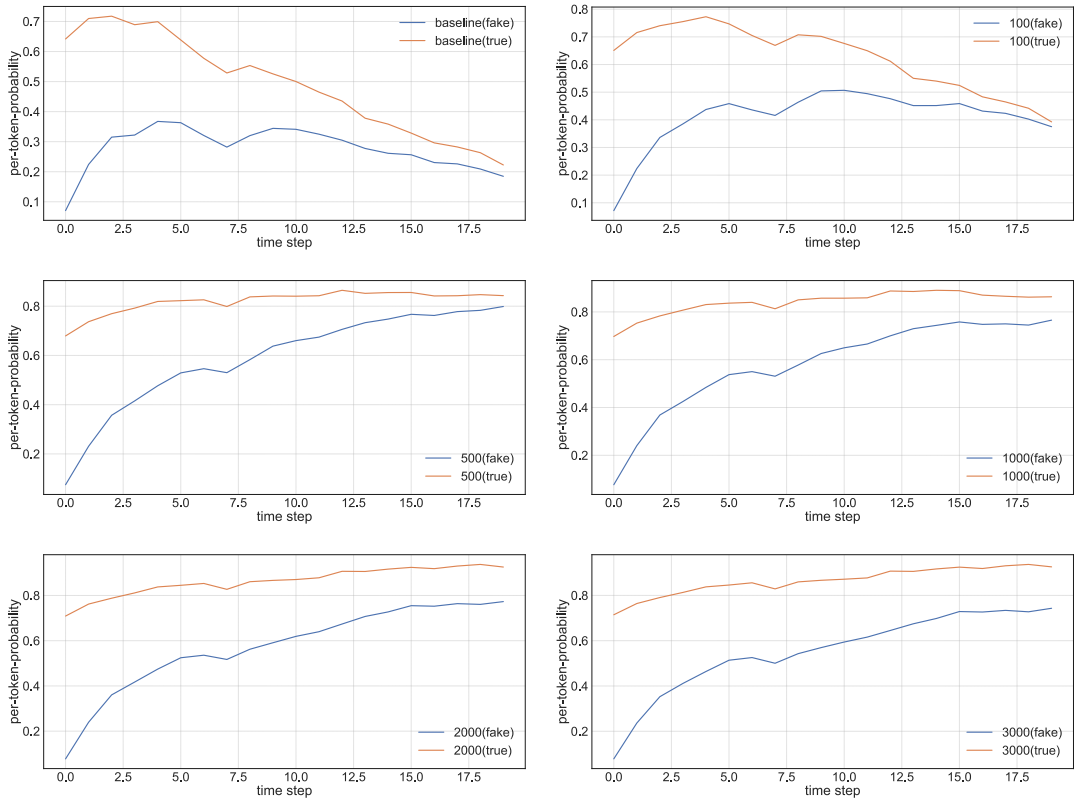

Figure B.2: Per-token probability with the increase of time step on *medial* true and fake target sentences given the correct source sentences. The models involve MLELS baseline and its MRT fine-tuning models.

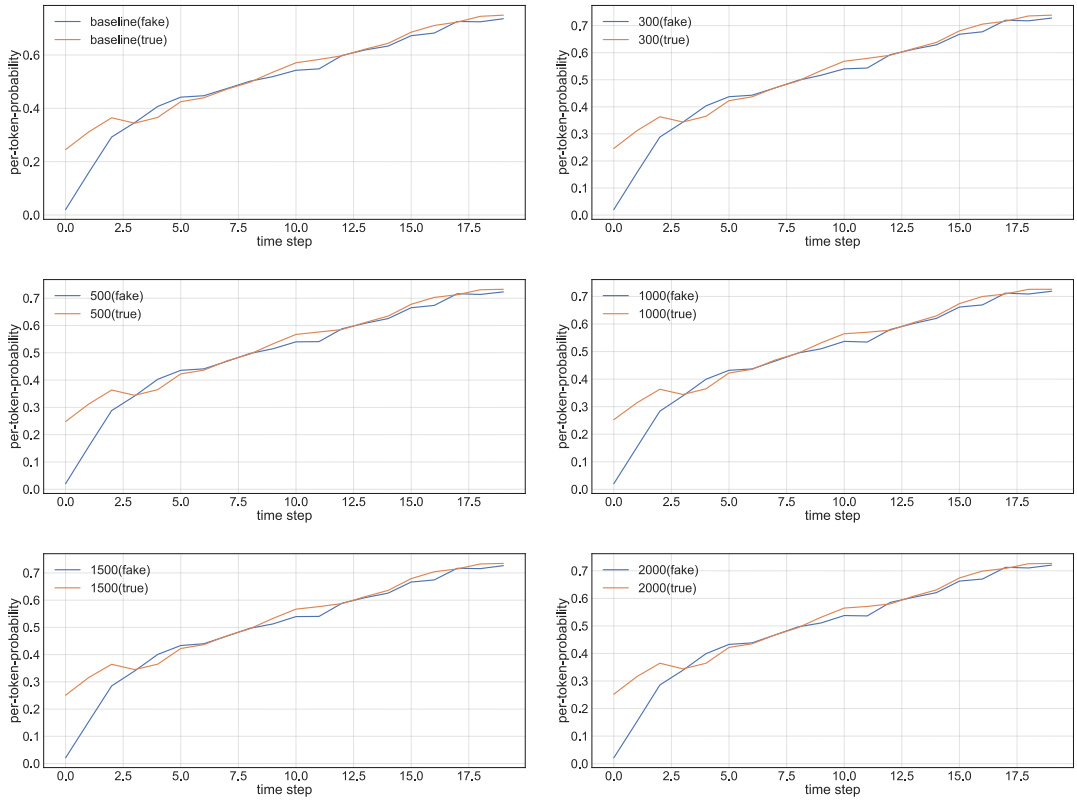

Figure B.3: Per-token probability with the increase of time step on out-of-domain true and fake target sentences given the correct source sentences. The models involve MLE baseline and its MRT fine-tuning models.