university of groningen

faculty of science and engineering

Improving Domain Robustness on Translating Out Domain Corpus

Ethelbert Uzodinma

university of groningen / faculty of science and engineering

University of Groningen

Improving Domain Robustness on Translating Out Domain Corpus

Master's Thesis Proposal

To fulfill the requirements for the degree of
Master of Science in Artificial Intelligence
at University of Groningen under the supervision of
Dr. Jennifer Spenader (Artificial Intelligence, University of Groningen)
and
Dr. Joost Doornkamp (Artificial Intelligence, University of Groningen)

Ethelbert Uzodinma (s3886026)

August 31, 2022

Contents

Abstract

Deep learning has become the latest approach to solving natural language processing tasks such as machine translation, because of its improved performance over translations previously made using statistical techniques. The evidence of this claim can be seen in translations involving a bilingual or multilingual parallel corpus where the source and target text are from within the same genre, known as “in-domain translation”. Despite these achievements, one of the six major challenges of neural machine translations still remains that translations using neural networks still produce poor performance when translating text from a genre different from the genre of the training set. This is referred to as “out-of-domain translation”. This thesis investigates different methods that be used to improve out-of-domain translations such as byte-pair encoding, sub-word regularization, beam size, label smoothing and domain-adaptation were applied to the training and fine-tuning stages in multiple out-of-domain translation experiments. The results showed improvement in fluency and adequacy. Evaluations using BLEU scores and perplexity showed an overall improvement of above 10% in out-of-domain translations.

1 Introduction

Machine translation is the process of using the computer to translate a text from one language to another and maintain the contextual meaning of the source sentence. More technically, It can be regarded as a sequence to sequence task(because it is time and order dependent) in natural language understanding that involves translating a sequence of text in one language to a sequence of text in another language using a mathematical model without changing the message it conveys. It is a very important area of research in the AI community. A typical machine translation workflow follows this general procedure for development as shown below in Fig 1. There are numerous problems that

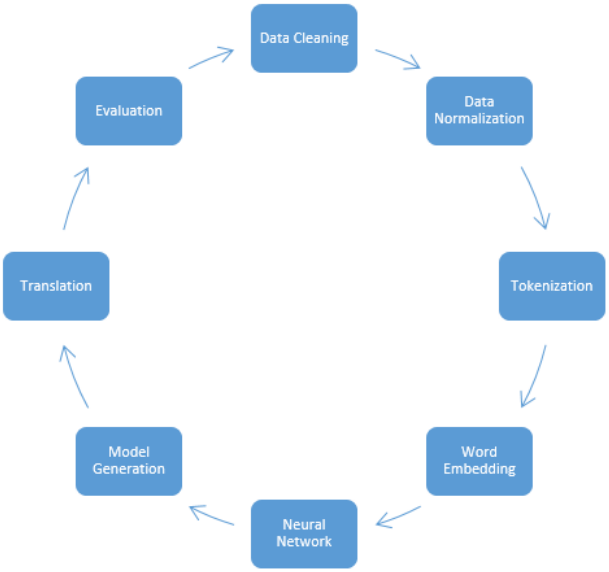


Figure 1: Illustration of general workflow for neural machine translation

have hindered the advancement of natural language understanding in the area of Machine Translation (MT). This problems become more visible with the size of the training data, domain shift, low resource words, sequence length of input, word alignment and beam search(Koehn et al, 2017[1]). The area of focus in this research is to overcome the challenges of MT in the area of domain mismatch. The outcome of this is the inability of MT systems failing in performance and not generalizing well when translating text outside the domain of the training set. When this occurs the MT model can be said not to be robust enough, which leads to the concept of Domain Robustness.

Domain robustness can be described as the ability of a machine translation model to generalize well to unseen data from other domains (Müller et al, 2020[2]). This concept of domain robustness satisfies the main goal of machine translation which is to learn models that generalize well to a probability distribution outside the distribution of the training set. Machine translation can be achieved by both traditional statistical analysis known as statistical machine translation (SMT) or using neural networks in this case known as Neural Machine translation(NMT). This problem in standard NMT systems can be visible using evaluation metrics to rate the quality of translation from an NMT engine. The evaluation metrics include, fluency and adequacy which are both based on human judgements and the BLEU score(Papineni et al, 2002 [3]) and ChrF score(Maja Popović, 2015[4]) which are automated techniques for evaluation.

Adequacy is used to evaluate if the output of the NMT system conveys the same meaning as the input sentence or If part of the message is lost, added, or distorted.

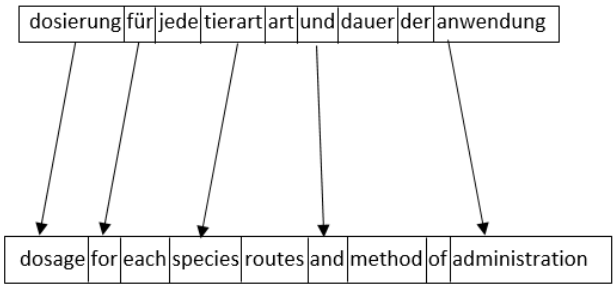


Figure 2: Illustration of a translated German → English text trained on medical text using joeyNMT engine)

Fluency on the other hand evaluates if the output is fluent in the translated language. It also evaluates both grammatical correctness and idiomatic word choices(adapted from Philipp Koehn "Machine translation" lecture slides in September 2020).
BLEU score measures the overlap of single words (unigram) or phrases (n-grams) between the predicted output and the target in terms of precision.
ChrF is another metric for automatic evaluation based on character n-grams. it computes F-score using character n-grams.
A neural machine translation(NMT) system can be said to be robust only after showing good performance both in the in domain and out domain, but unfortunately most MT engines are not able to give good performance for both in and out-domain test sets. The performance of an NMT system can best be evaluated by human or using other automatic means. But getting human annotators who are native speakers are often difficult and time consuming and their judgements can be often subjective, as a result we opt for an automated means.
An example of a machine translated sentence from a medical text from German to English is demonstrated showing human annotated evaluation based on fluency and adequacy below:

	German to English Translation	**Fluency**	**Adequacy**
Source	bei versehentlicher überdosierung sollte eine symptomatische therapie erfolgen		
Reference	In case of accidental overdosing symptomatic therapy should be administered		
joeyNMT	In case of accidental overdose symptomatic treatment should be initiated	5	4

Table 1: Source: Improving Domain Robustness on translating out of domain data: In-domain Medical corpus from the OPUS EMEA dataset

The ranking above on adequacy and fluency is done on a 1-5 scale with 5 been the best [5]. Previous works reported that both Statistical Machine Translation (SMT) and standard NMT (Vaswani et al, 2017[6]) are both affected by weak domain robustness of data from unseen domains. However, both systems are affected in different ways. For out-of domain data, SMT systems have very low fluency but can show good adequacy, while NMT systems are really poor in adequacy but show good quality in terms of fluency(Müller et al, 2020[2]).

Machine translations whose fluency and/or adequacy of translations are very bad, can make the message in the translated text uninterpretable, or mislead readers into making their judgment based on how fluent the sentence or phrase is in a grammatical context. In most scenarios, the translations of this nature are not related to the input. They usually occur during the domain shift or when we

SRC	*dieses abkommen tritt zwölf monate nach dem zeitpunkt der notifizierung ausser kraft*
REF	*the agreement shall cease to apply months after the date of such notification*
HYP	*the choice of this training occurs twelve months after the time of the ultrasonic force is uncertain*

Table 2: Illustration of NMT out-of-domain hallucination on a translated German→ English text trained on medical text but evaluated on a law text. *Source: JoeyNMT translation task in "Improving domain robustness in Out-domain corpus"*

have small training set in NMT system. This very bad translation can be referred to as hallucinations(Chaojun Wang and Sennrich, 2020 [7]). This problem is more serious as the training data and test data are more unrelated. This discrepancy can also be referred to as exposure bias(Razanto et al, 2016 [8]).
Different approaches have been suggested to mitigate this problem, we will investigate an ensemble of this methods which includes regularization techniques, domain adaptation e.t.c to see which of them shows promise or tends to reduce the problem. The effect of the methods will be tested on an NMT engine and evaluated using a number of evaluation metrics used in machine translation. We will also apply other text processing techniques before feeding it to a NMT engine.

1.1 Research Questions

To summarize, this research focuses on investigating the following problems:

Q1. Does subword regularization(BPE dropout) reduce hallucination during in-domain and out-of-domain translations?

Q2. Does label smoothing reduce hallucination on in-domain and out-of-domain translations?

Q3. Does applying domain adaptation technique reduce out-of-domain hallucination?

2 Background Literature

This chapter gives an overview of data-driven machine translation with regards to SMT and NMT and how they rely on parallel corpora to build a translation model. Also how machine translations are evaluated and factors to consider when making the choice on the best evaluation method to use. At the end of the section, an in-depth analysis of the hallucination problem is explained. It is important to note that hallucination occurs mainly during domain shift and causes the translated text to have little or no relationship with the input source text (adequacy).

The last section of this chapter further describes how hallucination occurs in a machine translation task. Also how we can identify and measure hallucination in a translated text using various methods and the need to measure hallucination, will also be explained.

2.1 Overview of Data driven MT

Modern machine translation systems are designed in such a way that given a sample text in one language, the output from the model are very much dependent on a fixed parameter space that was learned from a parallel corpora of the same genre or category. This type of translation is referred to as **in-domain**. In some cases, where there isn't sufficient parallel corpus data for training within a specific genre, where the model receives input text from a source that has a different style or from a genre different from the training data, it is referred to as **out-of-domain** and mostly leads to bad translation. If this translation is so bad that it is not related to the reference translation it is termed a **hallucination**.

SMTs and NMTs differ in many ways when performing a translation task especially with regards to data quality, data quantity, language model and also how they process text from in-domain and out-of-domain corpus. In SMTs, The model is built based on statistical techniques which models the inter-relationship between text elements such as words, phrases and sentences from the monolingual input text. The generated model is applied to another language of the sentence pair. On the other hand, translations involving NMTs require neural networks to generate a model from a bilingual parallel corpus.

Furthermore, in terms of data quality and quantity, SMTs are less strict with regards to bad data during and after training and require less amount of data during training, In cases where a bad data is discovered after training, the bad data can be deleted without retraining from scratch. But this is not the case with NMTs, which require data up to a million bilingual sentence pair for higher quality translation and any bad data after training often leads to retraining from scratch.

In terms of in-domain and out-of domain data, SMTs are more tolerant to a certain amount of data from outside the training domains. In dealing with out-of-domain data, they can still produce quality translations. However, this is not the case with NMTs where the presence of out-of-domain data often leads to lots of hallucinations.

2.2 Evaluation of Machine translations

Evaluation in machine translations is a very important component in determining how good a translation model performs. Due to the ambiguity of various languages with regards to translation, evaluation can be a difficult task and also because it depends on an individual's personal understanding of the language. As a result a text can be interpreted in different ways by different people.

During an evaluation task, the translated text is compared with a ground truth in the form of human-produced reference translation. This comparison can be done by both manual and automated

methods. For the manual method, this involves evaluation by a human, on the basis of two criterion, **fluency** and **adequacy** (Koehn et al , 2006[5]). The table 3 below can be used to interpret the the scores as follows:

Fluency		**Adequacy**	
Score	Meaning	Score	Meaning
1	Incomprehensible	1	None
2	Disfluent English	2	Little Information
3	Non-Native English	3	Much Information
4	Good English	4	Most Information
5	Flawless English	5	All information

Table 3: Liket scale for fluency and adequacy Source: An Augmented Machine Translation Evaluation Metric by Lifeng Han [9]

However, this manual method can be expensive and its results may vary greatly. For these difficulties and many more involved in the manual method, the automatic method is used as a substitute.

For the automatic method, we can evaluate the translations using **BLEU** scores (Papineni et al, 2002 [3]), **METEOR** (Banerjee et al, 2005 [10]) score, **Perplexity** scores among others. Some other methods that can be used such as **Word Error Rate(WER)** which is mainly used in speech to text systems, **Levenshtein distance** (Haldar et al, 2011 [11]) which also computes the cost it takes to correct the predicted translation to look like the reference translation and also **cosine similarity** (Rahutomo et al, 2012 [12]) which measures the level of similarity between the source and the predicted translation. For this research, BLEU scores and perplexity are chosen. The descriptions of the metrics and the reason of the choice will be described below:

1. **BLEU score**: BLEU is an acronym for **Bilingual Evaluation Understudy**. it is based on precision of the N-gram language model. It is the most commonly used method for automatic evaluation because of its simplicity. The score can be graded on a 0 to 100 scale. As the BLEU score of the translation gets higher it means that the precision of the translation is high and becomes closer to human translation. Refer to the table given in the table 4 to interpret the scores.

2. **METEOR scores**: Another common evaluation metric is METEOR. It is a acronym for **Metric for Evaluation of Translation with Explicit Ordering**. It improves on the BLEU score by evaluating the translation in relation to the ground truth based on both the precision as well as recall. It is often better at evaluation.

3. **Perplexity:** Perplexity is a measure of how accurate a model is able to predict the next token in the sequence. It can also be a measure of uncertainty. In the case of language models , low perplexity does not really imply good translation, instead it implies that the model is confident or more certain of its prediction. It can then be said that the model's input correlates well with the output.
   But it depends on the text pre-processing, vocabulary size, the context length. As a result it may be difficult to make comparisons across datasets that were pre-processed differently or have a different vocabulary size. The equation is given by:

$$PPL(X) = \exp\left\{-\frac{1}{t}\sum_{i}^{t}\log p\theta(x_i|x_{i-1})\right\}$$

Where $\log p\theta(x_i|x_{i-1})$ is the log-likelihood of the ith token based on previous tokens.

BLEU Score	Interpretation
<10	Almost useless
10-19	Hard to get the gist
20-29	The gist is clear, but has significant grammatical errors
30-40	Understandable to good translations
40-50	High quality translations
50-60	Very high quality, adequate, and fluent translations
60>	Quality often better than human

Table 4: Source: google cloud translate,"Understanding the BLEU score", refer here for more info

2.3 The concept of Hallucination in NMT

Hallucination as explained previously affects the NMT models in a very unusual way. To explain in detail, hallucination in NMT causes the translations to be fluent but not adequate. This implies that hallucinated translations are often grammatically correct but has a different semantic meaning from the source text. This results in misinterpretation and may convey a wrong message.
In production systems, even though this event occurs rarely, the consequence can be so high that it may compel the user to loose confidence in the model.
The question here is why do NMT systems experience hallucinations? There are a few reasons.

Firstly, when the NMT is pushed beyond its limit or safe zone, at this stage the hallucinations will be obvious because the text will be wrong to the reader because he can clearly observe gibberish or phrases abnormally repeated.

Secondly, the presence of hallucination may have to do with the statistical distribution of words in language. In such instances, hallucinations are not very obvious to the reader. Such cases may occur because of typo errors at the input text which have very low frequency of occurrence in the text or from mismatch based on genre between the data used to train the system and the test data to be translated. It is important to note that the frequency distribution of words in every natural language, whether spoken or written seems to obey Zipf's law, which is the law that describes how the words or tokens in a dataset is modelled in a frequency distribution. It was first proposed by George Kingseley Zipf and he states that "the frequency of a token in a text is directly proportional to its rank or position in the sorted list".

Due to this law, words in the dataset with high frequency of occurrence are mostly smaller in area highlighted in blue colour and is followed by a long list of noise, outliers, rare words and out-of-domain vocabulary e.t.c highlighted in yellow. The problem of hallucination lies in this yellow region. This is illustrated in the figure 3 below: As a result, words in a dataset with very low frequency of occurrence is susceptible to memorization by the model, so any minimal noise source, typo error or adversarial attacks applied to the input text is likely to cause hallucination.

To reduce hallucination and improve domain robustness, a lot of methods have been used by previous researchers these includes: Minimum Risk Training[14] (MRT), Byte-Pair encoding (BPE [15]), subword regularization (BPE dropout [16]), defensive distillation[17], reconstruction, Noisy Channel Modelling. In addition to this, I also proposed a few methods that I think may be useful to

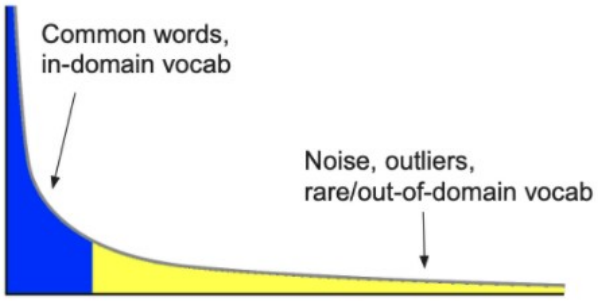


Figure 3: Frequency distribution of words in a text according to Zipf's law. Source: "**A curious case of hallucinations in NMT**" by Dr. Danielle Saunders, Further paper on this by Raunak et al [13]

further reduce hallucination and boost domain robustness of the model these include label smoothing, domain adaptation in combination with byte-pair encoding and sub-word regularization.
Further details of the previous methods used by the researchers will be explained in the literature review.

2.4 Hallucination Detection and Evaluation

In the previous section we explained some metrics for evaluating neural machine translations. To test for the presence of hallucinations, this same metric is adopted from the general metrics for evaluating NMTs and used to determine the amount of hallucination in the translated text.

The hallucination problem happens with both in-domain and out-of-domain translation, the only difference is that it occurs to a very large extent when translating to/from different domains, when the genre of the test corpus is different from the genre of the training set. In such cases, there will be large numbers of unknown words. In the experiment, there are three ways used to track hallucinations. This include:

1. **Using BLEU scores**: In situations where hallucination occurs, the BLEU scores are actually very low, usually below 10. Most times in hallucinations, the prediction from the model has no connection with the source sentence. From the experiment I did with JoeyNMT, I fed a law test set of 2000 German-English sentence pair into the model, The BLEU score of a sample translation when compared with the reference was about 9.67. See illustration in table 2.
   To interpret the BLEU score it is important to refer to table 4 from google cloud translate.

2. **Using fluency and adequacy:** Most of the time when hallucination occurs it usually sounds very natural, and is perceived as very "fluent", but does not preserve most of the message from the source. This evaluation metric is human annotated. To illustrate consider table 5, which shows German to English translation produced with JoeyNMT which lead to hallucination, to interpret the fluency and adequacy scores of the translation, one can refer to the table 3 above:

	German to English Translation	**Fluency**	**Adequacy**
Source	dieses abkommen tritt zwölf monate nach dem zeitpunkt der notifizierung ausser kraft		
Reference	the agreement shall cease to apply months after the date of such notification		
joeyNMT	the choice of this training occurs twelve months after the time of the ultrasonic force is uncertain	4	1

Table 5: trained on Medical corpus from the OPUS EMEA dataset, tested on law from the JRC-Acquis Source: "Improving Domain Robustness on translating out of domain data"

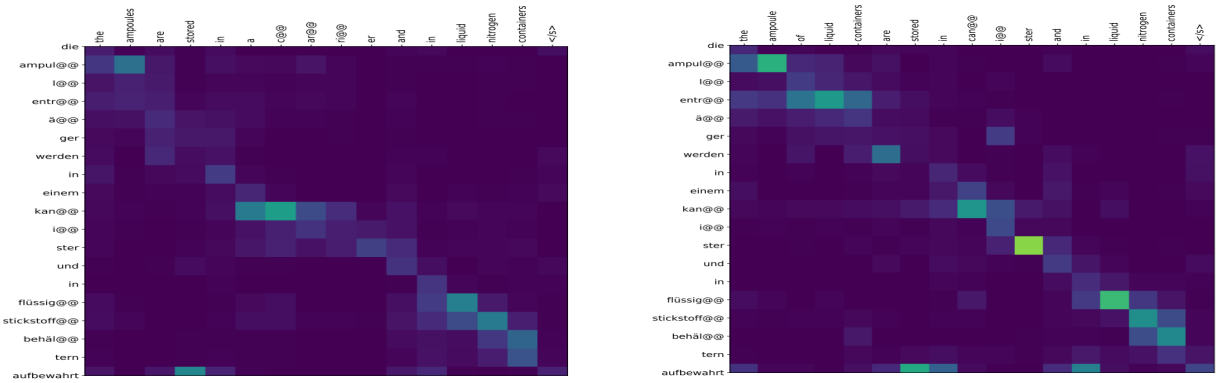
From the example above, recall from table 3 that by applying this human evaluation, translations that are evaluated as having high fluency but very low adequacy are classified as "Hallucination".

3. **Using Attention Matrices**
   The attention matrix map is provided by the attention mechanism of the transformer architecture. In combination with the help of positional encoding, the transformer is able to assign relevance to each token by assigning larger weights to the more relevant token and lower weights to the less relevant weights. The map is colour graded, so that the diagonal shows the amount of correlation between the source and the predicted text. The brighter colours shows stronger correlation between the source and the target. See below in table 6 a law text fed as input to the pretrained model in the medical domain and figure 4 is an illustration of the attention map of the model on the source law text fed into a model pretrained in the medical domain which hallucinates after byte-pair encoding was applied and became more robust when subword regularization (BPE dropout was applied).

This method is more of a visualization technique used to show improvement in translation quality than to identify hallucination between the methods used. However, from the map it can be inferred that the darker the colours along the diagonal the more hallucinated is the translation. While on the otherhand, the brighter the colour the more robust is the model at translating the input.

	German to English law text
Src	die ampullenträger werden in einem kanister und in flüssigstickstoffbehältern aufbewahrt
Ref	ampoule carriers are stored in canister and in liquid nitrogen containers
HYP 1	The ampoules are stored in a carrier and in liquid nitrogen containers
HYP 2	The ampoule of liquid containers are stored in canister and in liquid nitrogen containers

Table 6: Law text translated to after applying byte-pair encoding(HYP1) and sub-word regularization(HYP2)



(a) attention plot after byte-pair encoding

(b) attention plot after subword regularization

Figure 4: Attention matrix map to track domain robustness

In conclusion, of this section, we have presented 3 methods for identifying hallucinations, so now we can assume that these methods can also function as a general evaluation of translation quality. However, due to practical reasons, using fluency and adequacy is not possible.
As a result, only BLEU scores and attention matrix maps were used in this project to identify and evaluate the amount of hallucination in the translated text. In the next section we will have a detailed look into the papers that form the basis of this research.

2.5 Literature Survey on Improving Domain Robustness in NMTs

In this chapter, we will review other papers by other researchers who used various methods to reduce hallucination and tried to improve domain robustness most especially in out-of-domain data. We will review these methods, how they were used, what improvements was seen and lastly, their advantages and disadvantages.

Method 1 : Minimum Risk Training: *Shen et al, 2016[14]*

Minimum Risk Training (MRT) was first proposed by Shen et al, 2016[14] to be used for NMT tasks. The researchers (Chaojun Wang and Rico Sennrich, 2020[7]) applied Minimum Risk Training (MRT) technique during fine-tuning of the NMT model. The aim is to avoid exposure bias which may lead to hallucination in the translation. It also reduces beam search problem.
They performed their research with the IWSLT'14 German to English sentence pairs. The experiments on domain robustness was done on the OPUS dataset (Lison and Tiedemann, 2016), with genres in medical, IT, law, koran and open subtitles. They used Medical domain for training and development; some part of the medical data for in-domain experiment; a percentage of the others genres as test set for out-of-domain experiment. They applied byte-pair encoding(Sennrich et al 2016 [15]), before feeding it to the Nematus NMT toolkit(Sennrich et al, 2017).

Method 2 : Reconstruction: *Tu et al., 2017 [18]*

The researchers (Tu et al., 2017[18]) applied reconstruction technique on the output of the last decoder to reconstruct the input sentence from the decoder states. The problem with these is the tendency of more parameters generated during reconstruction, to mitigate these a training on multilingual data by combining this technique with other mentioned techniques (Johnson et al. 2017). In the research by Müller et al.[2], he used this technique to improve domain robustness in his out-of-domain experiments and reproduced a slight better performance with a +0.4 BLEU score above the baseline for the DE→EN. The drawback here is that a compromise is made between fluency and adequacy, so while there are visible improvements in the adequacy of the out-of-domain texts at the same time we have reduction in the fluency. An illustration of the reconstruction technique in NMT is shown in the Fig. 5.

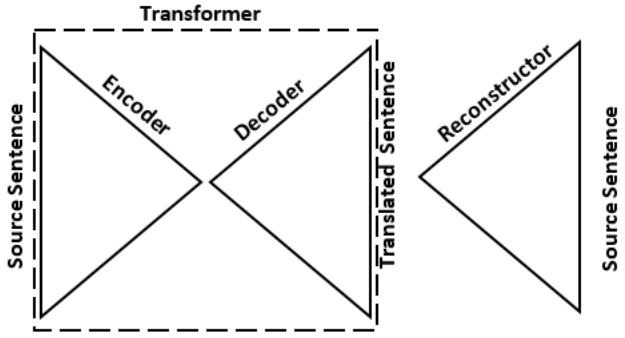


Figure 5: Source: Adapted from Mathias Müller lecture on "Domain Robustness in Neural Machine Translation"

Method 3 : Subword Regularization: Kudo, 2018[19]

This researcher (kudo) applied subword regularization technique for data augmentation instead of subword tokenization.
This technique involves the use of sentence piece tokenization instead of using the byte-pair encoding or using byte-pair encoding with BPE dropout[16] which is another form of subword regularization. (Müller et al [2], 2020) experimented with this technique by carrying out probabilistic sampling during training and k-best translated sentences during testing. This method produced improvements in in-domain and out-of-domain translation with BLEU score gain of +1.2 above the baseline in both cases.

Method 4 : Defensive Distillation: *Papernot et al. 2016 [17], Kim and Rush (2016)[20]*:

The defensive distillation is an adversarial learning technique. it was first used for image recognition (Ba and Caruana, 2014 and Hinton et al., 2015). In the original research by Papernot et al, the idea was only expressed but not tested in NMT, but in the work by Kim and Rush, it was implemented with a beam size of 10. To reproduce the same results, Müller et al[2] used this technique to improve domain robustness, by training the student model on outputs of the machine translation from the teacher model instead of its target label also referred to as ground truth. In his research, he attributed the out-of-domain translation as approximately equal to adversarial examples. The results where good especially in inputs from the out-of-domain test sets with an average gain of +1.4 above the baseline for DE→EN but did not show significant results in the in-domain.

Method 5 : Noisy Channel Reranking: Li and Jurafsky, 2016

The researchers here used noisy channel modelling to address the insufficient adequacy still present in the above methods. Another reason why this method was proposed was because the reconstruction method earlier mentioned can only be used during training. Müller et al.[2] 2020 in his paper used this technique by applying n-best list re-ranking because it's computationally cheaper compared to other methods.
The noisy channel re-ranking was done after the reconstruction method, it gave an improvement of +1.3 BLEU score over the baseline and a slight improvement of out-of-domain translation with an average gain of +0.5 BLEU above reconstruction results.

Using these methods, these researchers in their experiments, achieved the results shown in the table 7 below after applying these techniques on the translated sentences from German to English only on the same OPUS dataset judged based on the percentage of hallucinations in the predictions and BLEU scores.

In conclusion, it can be seen that despite the previous methods above, We still have a lot of hallucinations from translations in the out-domain data, and a few in the in-domain data. I feel that by applying some techniques like: label smoothing, domain adaptation e.t.c in combination with some of the previous methods mentioned we can reduce hallucination and make better quality translations.

Paper	Method	in-domain BLEU	out-domain BLEU	in-domain (% haullucination)	out-domain (% haullucination)
Wang et al, 2020	NMT+MRT using MLE	58.3	9.5	2	35
Müller et al, 2020	NMT + SR	61.4	11.2	1	37
Müller et al, 2020	NMT + D	61.1	13.1	3	33
Müller et al, 2020	NMT + RC	61.5	12.5	1	29
Müller et al, 2020	NMT+ RC+SR	60.3	13.2	-	-
Müller et al, 2020	NMT + RC+ NC	62.8	13.0	-	-
Müller et al, 2020	NMT + RC +SR+ NC	60.8	13.1	-	-
Müller et al, 2020	NMT	61.5	11.7	-	-

Table 7: *Sources: "Domain Robustness in Neural Machine Translation" by Müller et al, 2020[2], "On Exposure Bias, Hallucination and Domain Shift in NMT" by Wang et al, 2020 [7]. Where NC=Noisy Channel Model, RC= Reconstruction, D= Distillation, SR= Subword Regularization, MLE = Maximum likelihood estimation.*

2.6 Neural Machine Translation

The main aim of NMTs is to find the best translation $\hat{y}_k$ among the k-best translations, $y = y_1, y_2, y_3, ..., y_k$ candidates in our case 3-best, that maximizes the conditional translation probability equation given a source x_j from a set of source sentences $x = x_1, x_2, x_3, ..., x_j$ as shown below:

$$\hat{y}_k = \text{argmax} P(y_i|x_j) \tag{1}$$

Where the conditional probability is given by:

$$P(y_i|x_j) = \prod P(y_i|y_j, x_j) \tag{2}$$

Furthermore,

$$P(y_i|x_j) \approx P(x_j|y_i)P(y_i)$$

Where $P(x_j|y_i)$ represents the **adequacy** and $P(y_i)$ represents the **fluency**

2.7 NMT Framework

The JoeyNMT is a NMT toolkit based on Transformer architecture [6], capable of supporting novel neural networks architectures for implementing sequence-to-sequence tasks. It is built on pytorch, and can be used to carry out research and experimentation in tasks like: machine translation, question answering, speech translation and language modeling e.t.c. Previous methods used for neural machine translation of seq2seq tasks include Encoder-Decoder model, RNN, LSTM, GRU, CNN.

But in contrast to this Transformers are mostly preferred and currently regarded as the de-facto standard to use in tasks involving text generation because of its capacity to easily compute long-range contextual dependency in a sentence. This extra advantage is as a result of its attention mechanism (Bahdanau et al, 2015, [21]) which it uses to compute the sparse representations from the source text to the target text. Also, transformer based models are easily parallelized as compared to recurrent networks that are mostly sequential. This makes transformers faster compared to other models. They can also be applied to images in computer vision tasks.

The basic architecture of the transformer is made up of the following components:

- Input : The input consists of a sequence of sentence tokens that have been tokenized and converted into word embedding: which are a sequence of one-hot encodings because computers only understand numbers.

- Output : the output consist of sentences generated from the final decoder, which have been mapped to the sequence of output representations in the target language.

- Positional Encoding: are also vectors representing the position of tokens in the sentence. The transformer uses it to keep track of the position of words in the context they are used. They are generated using math functions of the sine and cosine. The sine is used for even positions of tokens in a sentence while the cosine is used for the odd position of tokens in the sentence. The full description of the equation of positional encoding is described the paper by Vaswani et al [6].

$$P(k,2i) = sin(\frac{k}{S^{2i/d}}) \tag{3}$$

$$P(k,2i+1) = cos(\frac{k}{S^{2i/d}}) \tag{4}$$

  Where

  - k : is the position of the token in the input sequence, $0 \leq k < L/2$,
  - d : is the embedding dimension of the output space,
  - $P(k,j)$: mapping of position k in the input to element (k,j) of the positional matrix,
  - S : scalar value set to 10000 from the paper,
  - i : maps the column indices $0 \leq k < d/2$

- Encoder: The encoder function is to map the word embedding to the input representations. It consists of two main components: the multi-head attention(a combination of many self attention staked in parallel) and the feed-forward neural network.

- Decoder: This component has the same components as the decoder with the addition of the masked-multihead attention layer. The presence of the masking makes the decoder unidirectional as opposed to the encoder block.

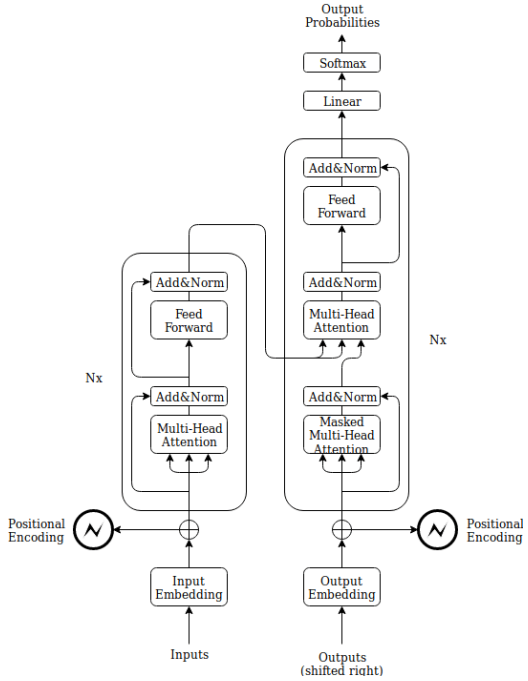


Figure 6: Transformer architecture, Source: Adapted from Vaswani et al [6]

3 Proposed Method

In the previous chapter, various methods used by previous researchers have been described and their performance in machine translation also evaluated. While these aforementioned techniques actually showed remarkable improvement in performance, it still showed numerous challenges. The aim of this chapter is to propose a new approach that leverages on some of the previous techniques but extends them with additional features as well as describe the preprocessing pipeline involved.

The method proposed in this chapter would be a combination of any of the methods described previously in the chapter on the literature review. MRT [14]was not used, because it cannot be combined with the other methods previously mentioned. The methods used will also be benchmarked against the baseline which is the model developed with the byte-pair encoding alone.

Firstly, I have to ensure proper text pre-processing of the source (German) and target data (English), also considering the diacritics of the source data would determine the kind of text prepossessing to be applied. After data analysis, it was discovered that the data contains quite a bit of noise such as duplicate sentences, presence of foreign language text other than German or English, unwanted symbols and sentences of length less than 2 words. This will be removed before training with the NMT. Details of the preprocessing pipeline is explained in chapter 5.

The problem at hand here is that we are dealing with a domain shift situation, because the test set is from another genre or domain other than the domain used to train the NMT model. So there is high rate of unseen or unknown words that cannot be found in the dictionary of vocabularies that was made during training and sub-word tokenization.

Another technique which I think that may help if implemented in the NMT engine is to provide joint vocabulary for the source and target languages otherwise known as joint BPE (Sennrich et al, 2016[15]). This can be implemented during the byte-pair encoding. Rather than learning from two unrelated encodings for the source and target languages and have the model learn separately from them, the model learns from a combined dictionary of the source and target. This method has been proved to be helpful when translating between languages that share common spelling and pronunciation for example German and English, French and English etc. While the previous method of encoding provides assurance that each subwords from both languages has been learned during training, the joint BPE provides consistency in both input language segmentation and output language segmentation (Sennrich et al, 2016[15]).

To improve the performance of the model, I attempted a combination of techniques such as joint byte pair encoding (BPE) and BPE dropout (Provilkov et al, 2020 [16]), during training of the model and domain adaptation during fine-tuning to give better output or generalization. The detailed description of these methods will be further described in the next sub-section.

3.1 Method description

This section further elaborates on the proposal at the beginning of this chapter and explains the details of the main methods applied during training and fine-tuning stages of the machine translation. The four main changes that made a great impact to my translation and helped reduce hallucination especially when working with out-of-domain data are listed and described below:

1. **Byte-pair Encoding (BPE)**: This technique can be regarded as a data compression technique as well as a type of tokenization method called sub-word tokenization. Other methods are either word and character based. Vocabularies in machine translation act as a dictionary from where the model compares to make judgment. I chose BPE as a tokenization technique because it solves the issue of out-of-vocabulary usually associated with word tokenization. It focuses on splitting the rare words into smaller sub-tokens that could become high frequency tokens rather than splitting high frequency tokens. I used a vocabulary size of 32000, which corresponds to the number of merge operations. The choice of 32,000 merge operations was chosen based on the size of the dataset by trial and error. First, I attempted a number from 16,000 then 32,000. The aim is to select the best "number of merges" that gives us the optimal representation of sub-words that can be sufficiently learned by the model while also reducing the likelihood of memorization of the model on representation of tokens that occurred just a few times. In the paper by Sennrich et al [15] , the use of BPE improved the model performance in the WMT 15 dataset involving English to German translations by +1.1 BLEU score above the word dictionary baseline.

2. **BPE dropout**: BPE dropout(Provilkov et al. [16]) technique is a type of sub-word regularization technique applied during training of the BPE. It is a simple approach which preforms multiple segmentation of the same token. It randomly drops some merge operations while making the tokenization.
   In the paper by Provilkov et al [16] he applied BPE dropout to a German to English translation task, in an experiment involving an original and misspelled source text, it yielded an improvement in BLEU score of +1.5 for original source text and an increase in +2.32 for misspelled source text.
   The amount of dropout is regulated by a probability score between 0 and 1, with a score of zero representing the original BPE segmentation and a score of 1 represents a character based segmentation. In my research, I used a score of 0.1 signifying a dropout of 10% of each merge operation.

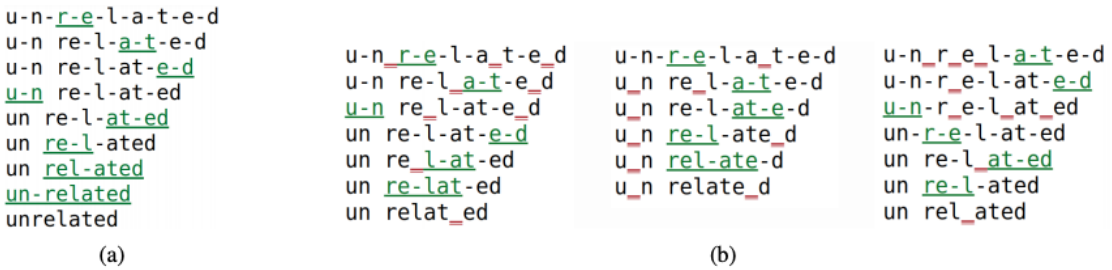


Figure 7: Source: "BPE-Dropout: Simple and Effective Subword Regularization" by Provilkov et al.[16] This illustration describes BPE (a) and BPE dropout (b) process of the word 'unrelated'. The hyphens represents a merge point, each merge is shown in green, the dropout points are shown in red underscores, while each line of text represents an iteration.

3. **Label Smoothing** : Label smoothing (Gao et al, 2020 [22]) is both a regularization technique and a method used to prevent overconfidence in the model. Another reason for applying this technique because I was having issues with generalization. In the paper by Vaswani et al [6], a label smoothing value of 0.1 was used in training the machine translation and improved the performance of the transformer model by increasing uncertainty and reducing the perplexity score. The result of this is that it forces the model to learn more before making a decision on prediction. I also applied this same value for label smoothing during training of my transformer model.

4. **Domain Adaptation**: Domain Adaptation (Chu and Wang, 2018[23]) is a technique usually applied in low resource context or when we have a mismatch between the source genre and target genre to improve the translation in out of domain context. This method is a type of transfer learning technique applied during the fine-tuning stage of machine translation. See figure 8 for further illustration.
   This method transforms the parameter space of the model whose parameters was pre-initialized base on the large in-domain dataset which in this case is the medical domain into the parameter space of the new input dataset from a different genre. This technique has been previously applied in computer vision and machine learning task. I will adopt this technique in this research during the fine-tuning stage of the machine translation workflow.

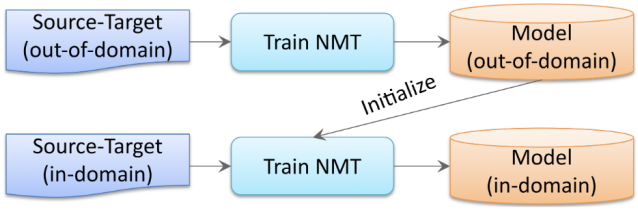


Figure 8: Fine-tuning using domain adaptation. Source: Chu and Wang, 2019 lecture "Domain Adaptation for Neural Machine Translation"

3.2 Text Processing

When analyzing the dataset, we discovered that the corpus contained about 143,000 duplicate parallel text and some noise which includes phone numbers, email address, bullets and other non-ascii characters. A text processing pipeline was developed to remove the excess noise in such a way as to avoid mis-alignment of the source and target by synchronous parallel processing of the corpora.
To illustrate, the method used for cleaning was mainly by the use of regular expressions, and by unicode normalization. The regular expression gets rid of the artifacts such as phone numbers, emails, double spacing, web/hyperlinks while the unicode normalization gets rid of non-ascii-characters like bullets and numbering, symbols, mathematical formulas, roman numerals. Also the unicode normalization decomposes the language specific characters that have accents, diaeresis, umlaut as in the German text.
The importance of unicode normalization is to get rid of bad data that could have a negative impact on the performance of the model by increasing training time. Removing them helps the model to focus only on key features that benefit the model. Furthermore the text has to be normalized to lowercase, which helps ensure consistency during the entire workflow. Also punctuations were removed, but

stop words were retained in the machine translation task because I think it's necessary to add more meaning to the translation and also because of the attention mechanism (Bahdanau et al, 2015[21]). Further description of the text processing pipeline is described below.

1. **Unicode Normalization**: There are many normalization strategies depending on the outcome you are trying to achieve and the language. Unicode normalization, involves the composition and decomposition of the characters into its base character and its (accents or diaeresis or umlaut). Every character of a language is made-up of its Canonical equivalence and its compatibility equivalence. In the medical German text used in the experiment we have characters which include ä, ö, ü and ß . In an example below using the NFKC normalization method (see Fig 9), the German text is decomposed into a compatible form and then reformed into its canonical form for easy filtering of unprintable characters.

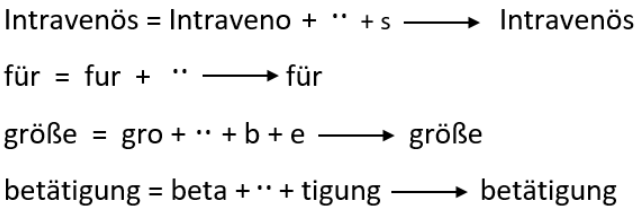


Figure 9: NFKC unicode normalization

2. **Punctuation Removal**: The presence of punctuation marks can introduce noise into the neural machine translation system during training and so had to be removed or replaced with an empty space.

3. **Lower-casing**: used to standardize the text and maintain consistency across characters that will be fed to the NMT.

4. **Tokenization or Segmentation**: This stage in very important in the processing pipeline because it helps to reduce computational cost and training time by limiting the number of vocabularies, given more instances of each word which really improves the quality of the translation. This can come in the form of word tokenization or subword tokenization like byte-pair encoding(BPE).

5. **Filtering**: filtering removes parallel text from the source and target that contains fewer than 3 words. This technique is used to select quality sentence pairs that will contribute significantly to the training process.

4 Experimental Setup

This chapter describes the mode of experiments across the domains to test for the presence of hallucination, the model hyper-parameters, the resources and tools; the source of the dataset, text processing pipeline and the method for evaluating general machine translations as it relates to the project.

The neural network architecture used in the experimentation of this project is the Transformer model (Vaswani et al[6]), This architecture is based on encoder and decoder model used for translation of text from one source language to the target language. This architecture is built into several open source toolkits used to achieve this project. The major tools used in this project include:

- JoeyNMT: A minimalistic neural machine translation toolkit for research based on pytorch. It can also be used to implement RNN and transformer as well as other NMT architecture models(Julia Kreutzer et al, 2019 [24])
- Experiments were done remotely on Peregrine: Accessed using SSH.
  - NVIDIA V100 GPU
  - Python 3.74 + CUDA
  - Number of CPU cores per node: 12 cores. Only 1 node was assigned for the experiments on the virtual environment.
  - RAM 120 GB
- OPUS dataset: a growing collection of translated texts from the web, containing parallel data from various domain, with bitext alignment. See **OPUS documentation** For this research we are using domains from Medical, Law, Koran, IT and open subtitles.
- Other Machine learning/deep learning packages are listed in the requirements document

The NMT toolkits listed above will be implemented using the Transformer architecture and trained on the remote supercomputer with GPU settings.

4.1 Dataset

The dataset used in this current research came from the OPUS and contains five domains: medical, IT, koran, law and open subtitles. From these domains, a subset of the medical domain was used as the training set, while a subset of all the domains including the medical domain were used for the development and test set.
In order to improve domain robustness in machine translation previous researchers applied different methods during fine-tuning of the model using the same OPUS dataset. They performed experiment on the parallel corpora translating the source in German $\rightarrow$ English. This was done to be able to experiment on the domain robustness of the NMT system. The table 8 below shows further details about the dataset to be used in the experiment.
From the dataset, about 950,000 samples where selected for the experiment with the EMEA as training set and subsample of 2000 was used for the development and test set, because about 143,000 of the 1,183,000 where overlaps between the source and target. The out-of-domain datasets are law, IT, OpenSubtitle, koran and from each I randomly selected a sample of 2000 each from them.

DE-EN			DE-RM		
Domains	Corpora	Size	Domains	Corpora	Size
Medical	EMEA	1.1m	law	Allegra, Press Release	100k
IT	GNOME, KDE, PHP, Ubuntu, OpenOffice	380k	Blogs	Convivenza	20k
Koran	Tanzil	540k			
law	JRC-Acquis	720k			
subtitles	OpenSubtitles2018	22.5m			

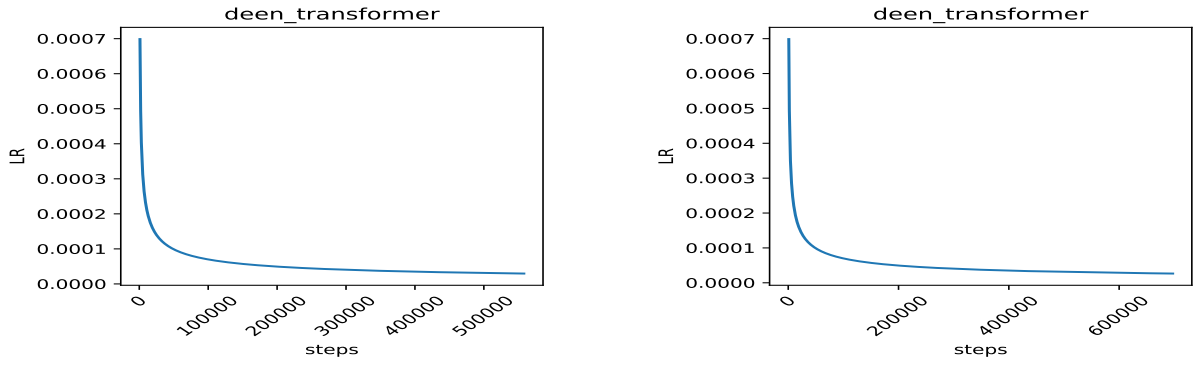
Table 8: *Source: Domain Robustness in Neural Machine Translation by Müller et al, 2020[2]*

4.2 Tools and Technologies

- Sacremoses: python based tool that supports Moses tokenization, truecasing.
- JoeyNMT: A neural machine translation toolkit

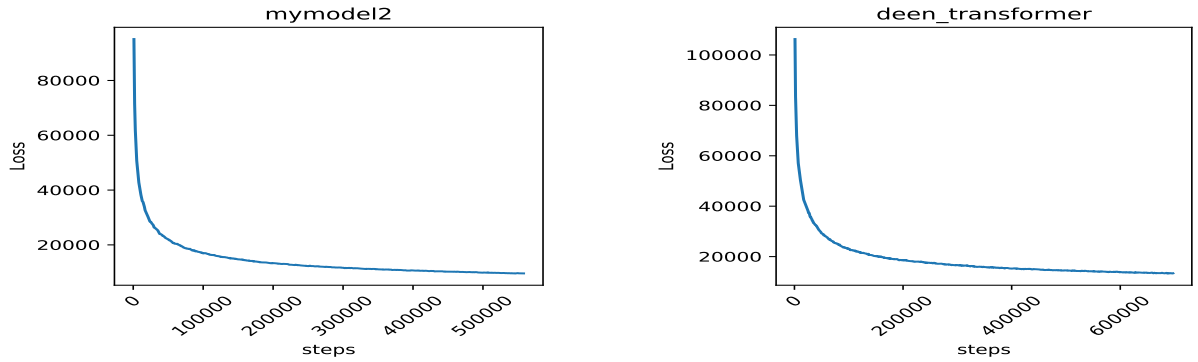
4.3 Model

The model is trained using the transformer architecture on the various datasets from the five domains. The IT domain was augmented with 5 different corpora namely: KDE, GNOME, UBUNTU, Open Office, PHP texts, because it does not contain much dataset enough to train the model.
The learning rate was selected by trial and error. Starting with a very small rate and discovering it was too slow, then I divided by a factor of 10 to reduce the training time.
The model was first trained on the large dataset which is the medical domain then a test set of 2000 German-English sentence pairs was randomly selected from the other domains which are the law, koran, IT, Open subtitles. After the model was generated in the medical domain, it was used for translation, and transfer learning was applied on the pretrained model to transform the model that was previously preinitialized to map the smaller low resource corpus. This technique is known as domain adaptation (refer to chapter 4 for the full description). In the illustration in Fig 10 of 557000 iterations for the model with byte-pair encoding while the model with sub-word regularization was trained at 660000 iterations with a step of 1000 each.



(a) learning rate decay with byte-pair encoding (b) learning rate decay with subword regularization

Figure 10: learning rate of model after 45 epochs



(a) Cross entropy loss with byte-pair encoding

(b) Cross entropy loss with subword regularization

Figure 11: training loss of model at 45 epochs

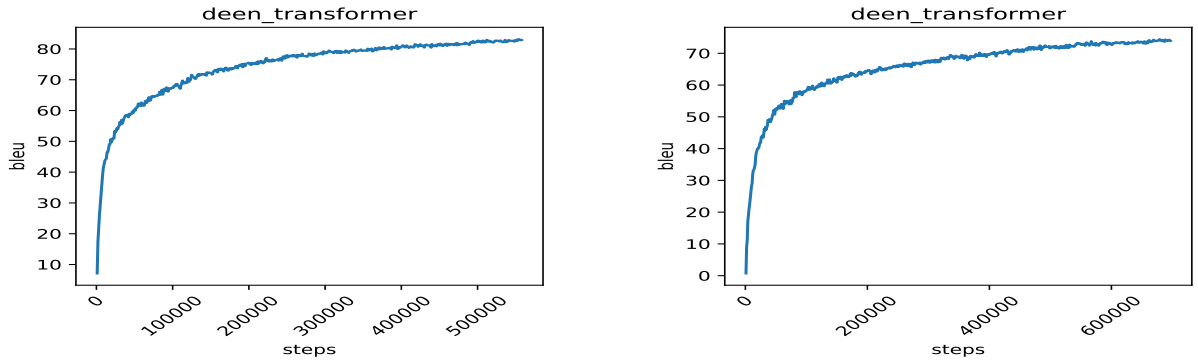
The Figure 10 shows the learning rate decay over 45 epochs. From figure 10a, the plot shows a smooth decay from the initial learning rate of 7E-4 to below 1E-4 using the adaptive learning rate (Adam), to make a smooth decline when the learning rate becomes stable at the global minimum below 1E-4 after in each step a batch size of 4086 tokens is processed. A learning rate factor of 0.5 and a decrease factor of 0.7 is used to reduce the learning curve smoothly to the global minimum. The difference between figure 10a which uses BPE and figure 10b which uses sub-word regularization is that the figure 10b takes a bit more time to converge to the global minimum hence we have 666000 steps.

The Figure 11 shows the decay of the cross entropy loss during training. Early stopping was used as a regularization technique during the training to stop the model from overfitting. The cross entropy loss during training decayed to a value of 4689.60 and 13,288.5 respectively for training with BPE and BPE with subword regularization.

4.4 Model Evaluation

Evaluation of machine translations are usually done from the test set after training the model to evaluate its quality and accuracy. This is mostly by automated means because of the difficulty involved in using human annotators. The evaluation metric used for this project is BLEU score and perplexity.

- **BLEU score:** BLEU score here as described in the previous chapters is used to compare the similarity of predicted translation to the ground truth. Further information on how to interpret the scores is shown in table 4 above. This score is calculated from the n-grams. During training, the BLEU score improved for each epoch and finally stops at about 82% and 73% respectively after byte-pair encoding and the addition of sub-word regularization, as a result of early stopping on the BLEU score.

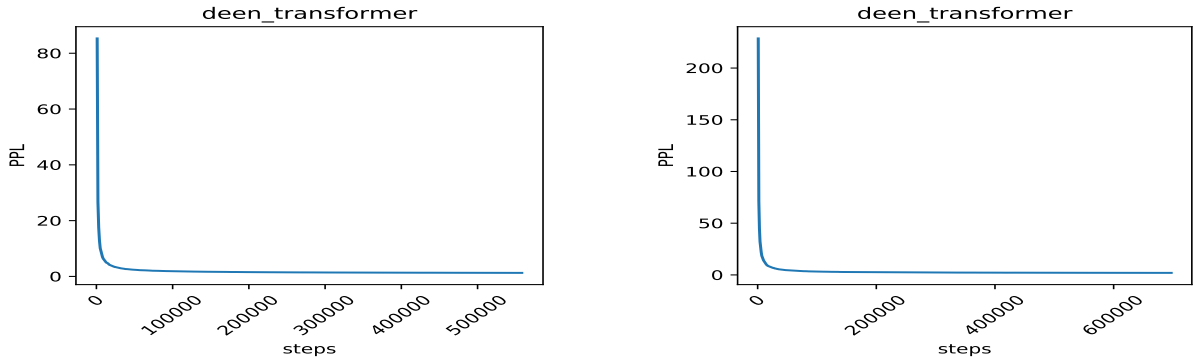


(a) BLEU score after byte-pair encoding

(b) BLEU score after subword regularization

Figure 12: BLEU score of training set

- **Perplexity**: This evaluation metric is used to benchmark the ability of the model to generalize in a machine translation task. The theoretical importance of this metric is explained in chapter 2. However, this same metric can have impact on the hallucination. The smaller the perplexity score the better the generalization performance.
  After applying byte-pair encoding the validation set produced a perplexity score of 1.9787, while after BPE with sub-word regularization, we got a score of 1.2716. As a result of this, better generalization with out-of-domain text leading to a reduction in hallucination .



(a) Perplexity score after byte-pair encoding

(b) Perplexity score after subword regularization

Figure 13: Perplexity score of training set

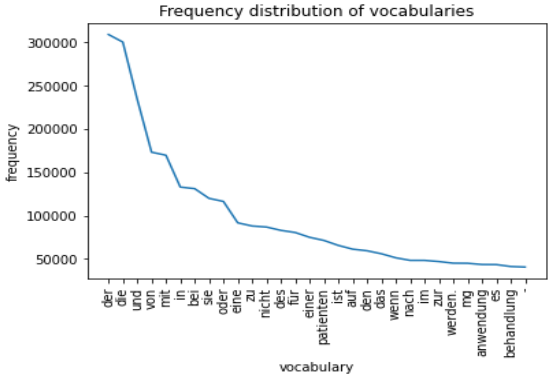
5 Results

After applying the techniques listed above across the 5 domains, we have the results of in-domain and out-domain translations with some improvements in BLEU score as well as fluency and adequacy. Domain adaptation was applied only to the out-of-domain corpus. The out domain translations were tested on 2000 German-English sentence pairs.

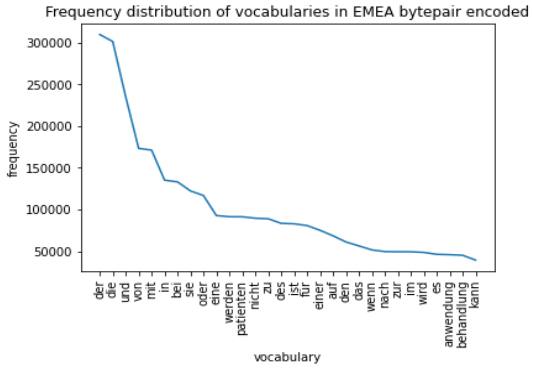
BLEU score was used as the only evaluation metrics while analyzing the results for hallucination because fluency and adequacy involved human annotation as such may not be possible to use for evaluation. On the other hand, attention matrix can only be used at the sentence level to visualize the amount of correlation between the source text and the predicted text, so it may not be feasible to visualize the entire 2000 test set.

5.1 Task 1: In-domain Medical data results

To produce the results in the medical in-domain, a test set of 2000 sentence pairs was taken out of the medical corpus, and used to evaluate the performance of the model based on the BLEU score evaluation metric.

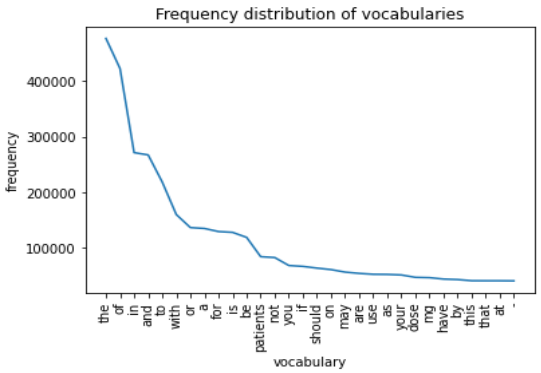


(a) 97614 unique German vocabularies before byte pair encoding

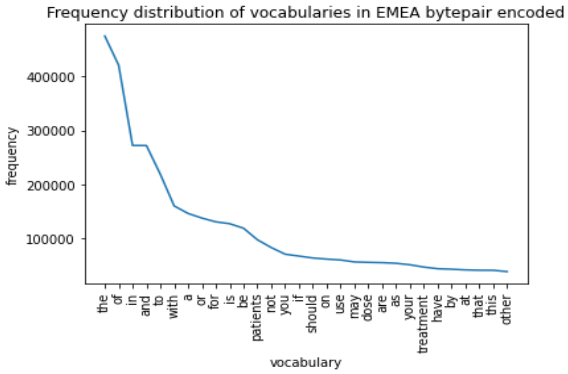


(b) 32000 unique German vocabularies after byte-pair encoding

Figure 14: Power Frequency distribution of German tokens in the source Medical text



(a) English vocabularies before byte-pair encoding



(b) 32000 English vocabularies after byte-pair encoding

Figure 15: Power Frequency distribution of English tokens in the target Medical text

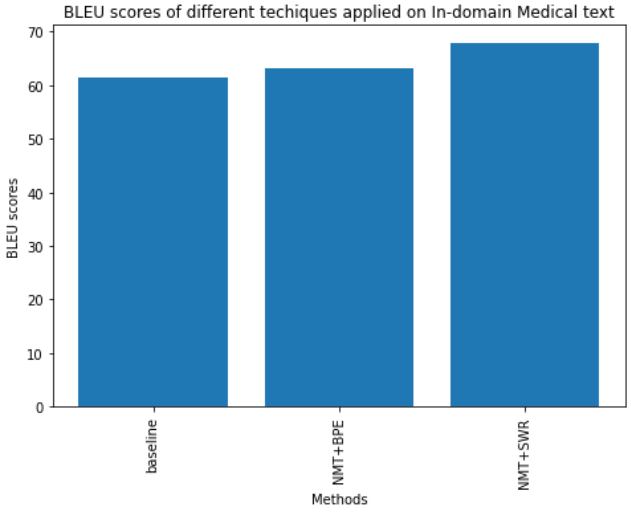
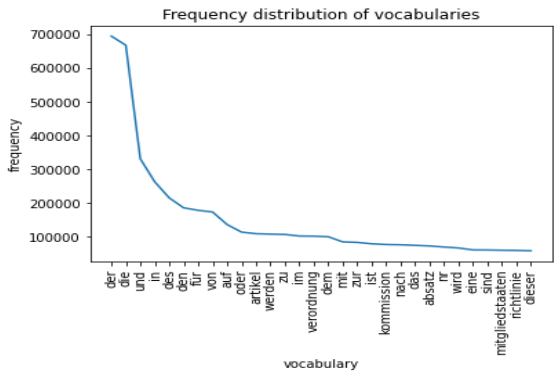


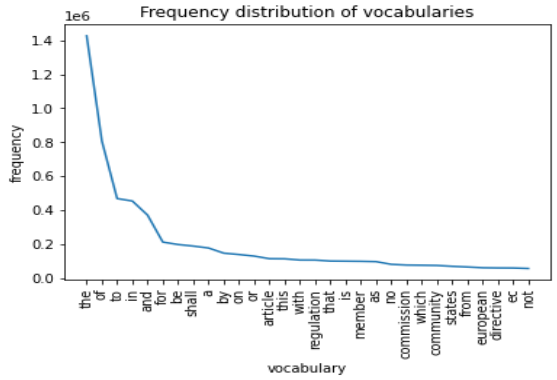
Figure 16: BLEU score for in-domain experiments

5.2 Task 2: Out of domain Law results

After the law text has been cleaned, it had 59,803 unique vocabularies and BPEed to 32,000 vocabularies, a sample law test set of 2000 sentence pairs was fed to the pretrained model of the medical domain. This still showed hallucinations with a low BLEU score. Then label smoothening and domain adaptation was introduced to the byte pair encoded pretrained model which boosted the BLEU score.



(a) German law vocabularies before byte-pair encoding



(b) English law vocabularies before byte-pair encoding

Figure 17: Power Frequency distribution of law source and target text

	German to English translation on law text	BLEU
Src	die vertragsparteien können hierzu in dem gemischten ausschuß konsultationen durchführen	
Ref	to this end the contracting parties may consult each other within the joint committee	
HYP 1	the prostacyclin can be used in the mixed committee acyclin	3.4
HYP 2	the penicillins could interfere with the mutagenic committee	3.6
HYP 3	the chairman may undertake to do so in the joint committee	12.6

Table 9: Sentence level BLEU scores after BPE (HYP1), sub-word regularization (HYP2), domain adaptation (HYP3) applied to a sample text in law domain

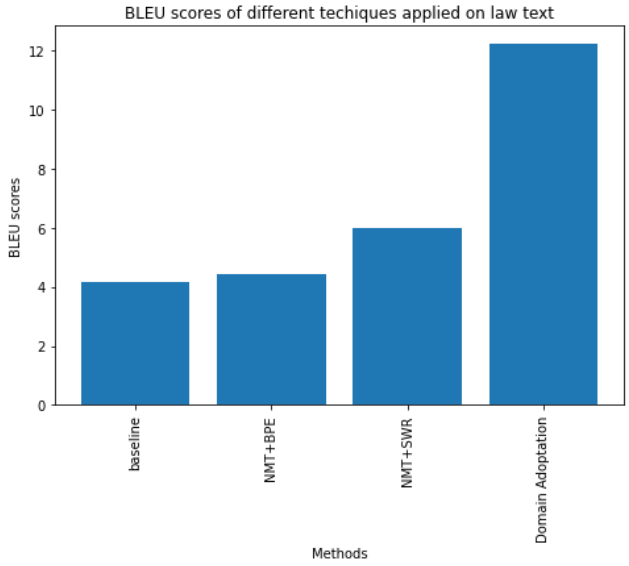
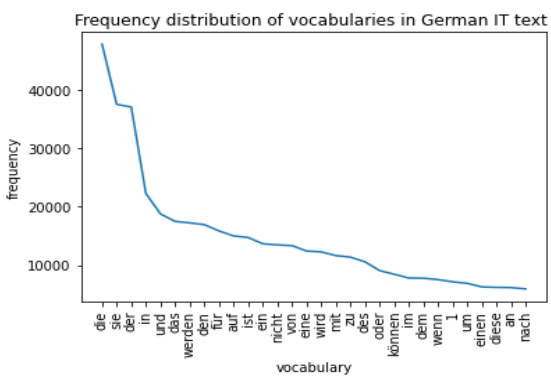


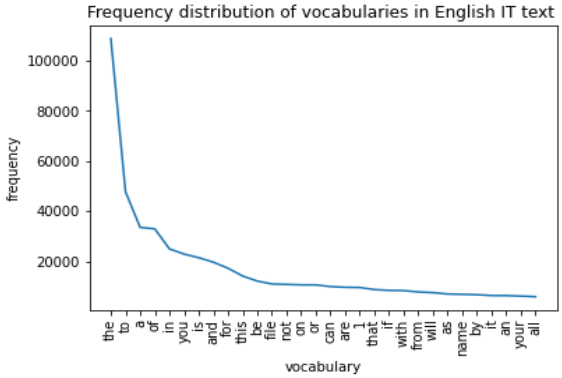
Figure 18: Corpus level BLEU score for out-domain law experiments

5.3 Task 3: Out of Domain IT

The IT domain has a small corpus, as a result data augmentation was applied to boost the token frequency of IT domain. This is an attempt to move the token frequency away from the yellow region illustrated in Figure 3 into the blue region. After applying byte-pair encoding, sub-word regularization and domain adaptation respectively to the test set, It was discovered that the domain adaptation showed highest improvement in the IT domain. The corpus level BLEU score did not show much improvement over the BPEed baseline when sub-word regularization was applied to the BPE.



(a) IT German vocabularies before byte-pair encoding



(b) IT English vocabularies before byte-pair encoding

Figure 19: Frequency distribution of IT domain text before BPE

	German to English IT translation	**BLEU**
Src	ist ohne namen und beschreibung	
Ref	has no name or description	
HYP1	if not all the names and addresses	0.0
HYP2	what lamictal looks like and contents of the pack	0.0
HYP3	is not a name and description	9.65

Table 10: Sentence level BLEU scores after BPE (HYP1), sub-word regularization (HYP2), domain adaptation (HYP3) applied to a sample text in IT domain

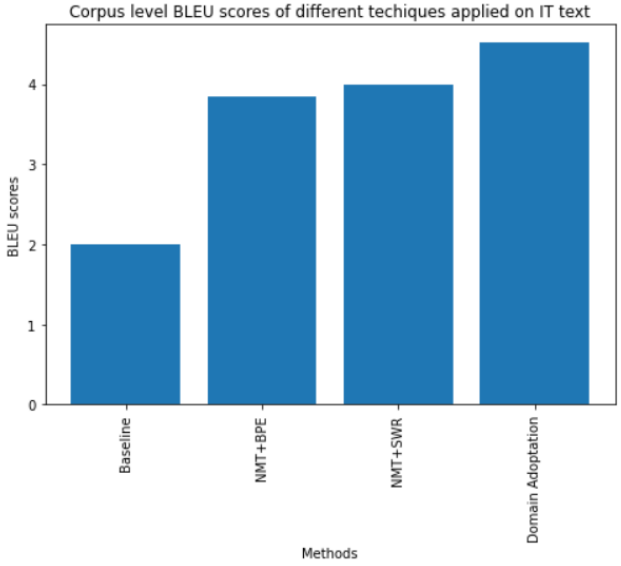
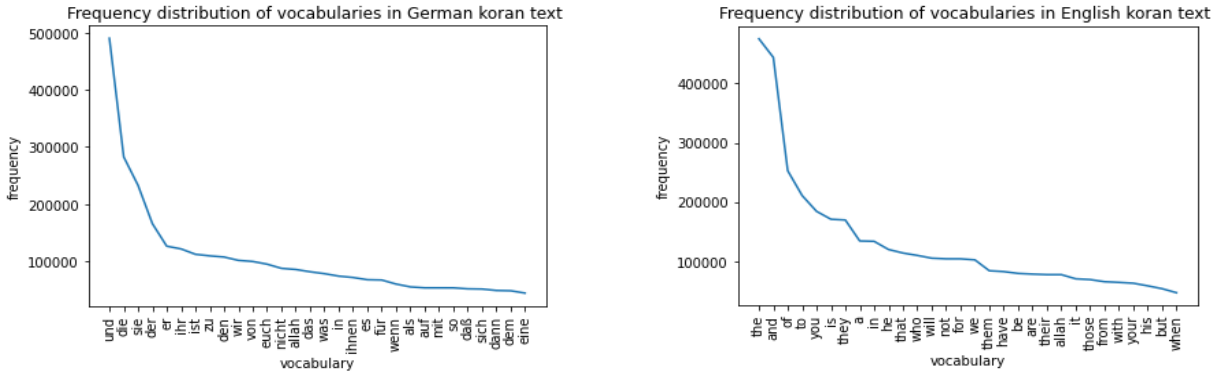


Figure 20: Corpus level BLEU score for out-domain IT experiments

5.4 Task 4: Out of Domain Koran

This corpus is more unrelated to the medical corpus used to train the model than the other domains. Before byte pair encoding we have about 35,095 German Koran text and after BPE we have 32000. The results in the Figure 22 shows the test BLEU scores after applying BPE, subword regularization and domain adaptation respectively. A sample text from the Koran is used to visualize the impact of domain adaptation which gave the best results as shown in table 11. The subword regularization did not show better improvement at the corpus-level in the test set. The test set is made up of 2000 sentence pair.



(a) Koran German vocabularies before byte-pair encoding (b) Koran English vocabularies before byte-pair encoding

Figure 21: Frequency distribution of Koran domain text before BPE

	German to English translation on law text	BLEU
Src	Ihre Beschützer sind nur die Gottesfürchtigen, jedoch die meisten von ihnen wissen es nicht.	
Ref	Its guardians could be only those who are pious and devout. But most of them do not know.	
HYP 1	however most patients will be able to benefit from their knowledge	1.96
HYP 2	cystine levels are only the most likely to cause you to run low of calories	2.66
HYP 3	but most of them do not know	12.6

Table 11: Sentence level BLEU scores after BPE (HYP1), sub-word regularization (HYP2), domain adaptation (HYP3) applied to a sample text in Koran domain

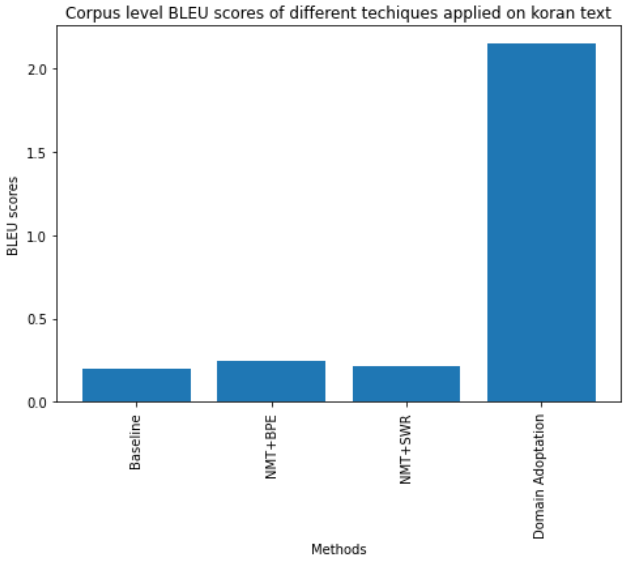


Figure 22: Corpus level BLEU score for out-domain Koran experiments

5.5 Task 5: Out of Domain Open Subtitles

This domain is the most unrelated to the medical domain used to train the dataset. This is because the corpus level BLEU score was the lowest among the other out-of-domain experiments. Despite the attempt to adopt the parameters of the medical domain using domain adaptation, we still had some unknown words that could not be recognized by the new model. Figure 23 is used to visualize the corpus level BLEU scores after applying the techniques.

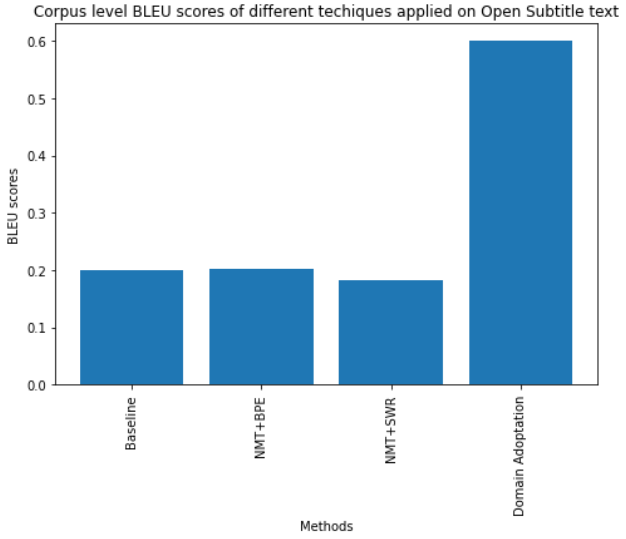


Figure 23: Corpus level BLEU score for out-domain Open Subtitles experiments

5.6 Summary of Hallucination content

So the purpose of this summary is to visualize the amount of hallucinations found after applying the following techniques which include BPE, sub-word regularization, domain adaptation. This statistics was generated from a test set of 2000 German text from all the domains, which was translated , and then the predictions compared with the reference. The threshold of the hallucination was set to 1.0 BLEU scores, as stated in the paper by Lee et al. 2019 [25].
Table 12 shows the number of hallucinations found in each domain , while Table 13 shows the percentage of hallucinations respectively after applying a combination of BPE, sub-word regularization, domain adaptation.

Domain	NMT + BPE		NMT +BPE +SWR		NMT +BPE + DA	
	Number of Hallucinations	Total Sentence lines	Number of Hallucinations	Total Sentence lines	Number of Hallucinations	Total Sentence lines
law	820	2000	761	2000	450	2000
IT	1193	2000	1123	2000	862	2000
Open Subtitles	1220	2000	1185	2000	1054	2000
Koran	1214	2000	1090	2000	1073	2000

Table 12: Number of hallucinations across the domains at a threshold of 1.0 BLEU, *Where BPE=byte-pair encoding, SWR= sub-word regularization, DA = domain adaptation*

Domain	% hallucinations NMT + BPE	% Hallucinations NMT + BPE +SWR	% Hallucinations NMT + BPE + DA
Law	41	38.1	22.5
IT	59.65	56.15	43.1
Open Subtitles	61	59.25	53.7
Koran	60.7	59.3	52.7
Average	55.6	51.9	41.3

Table 13: Percentage hallucinations from the translations of 2000 German test set from each domain. Also average amount hallucinations across the out-of-domains is shown in percentage

6 Conclusion

From the table 12 and table 13, it can be inferred also that the law text is a bit closely related to the medical domain than the other domains, as a result its able to generalize better when its sample text is fed into the model. Also it can be assumed that hallucination decreases with increase in BLEU scores across the domains. See the combined plot in appendix 24. The open subtitle text is the most unrelated as can be seen in the number of hallucinations. This makes the koran and open subtitles the most difficult to adapt using domain adaptation. Also I observed that BPE and BPE with sub-word regularization did not show much improvement in the koran and open subtitles, maybe it has to do with how unrelated the text is to the the domain used in training the pre-trained model which is the medical domain.

6.1 Summary of Main Research questions

1. Does byte-pair encoding with subword regularization(BPE dropout) reduce hallucination in-domain and out-of-domain translations? Answer: subword regularization did not give significant results for in-domain corpus but produces significant results in out of domain. This is because when tested for in-domain we gained only a BLEU score of $+0.5$. while on average it showed significant improvement of above $+2.0$ BLEU in out-of-domain for law and IT domains.

2. Label Smoothing reduce hallucination on in-domain and out-of-domain translations? Answer: works well for both in-domain and out-of-domain data. When applied as a regularization method, it gives an improvement in the BLEU scores in both cases.

3. Does applying domain adaptation technique reduce out-of-domain hallucination? Answer: absolutely yes, this can be assumed from the reduction in percentage hallucination as seen in the table above.

6.2 Discussion

1. Hallucinations effects are more likely in neural machine translation than in statistical machine translations. There are many other reasons for hallucination in NMT besides domain mismatch. According, to Yan et al.[26]. He also believes that hallucination can be caused by faulty encoder layer and embedding layer in the Transformer.

2. Hallucinations reduce the quality of translations and sentences that exhibit hallucinations are most times informative. This is because the words in those sentences have very low frequency of occurrence, but should not be discarded because we need enough data to train the model.

3. We can reduce hallucinations using sub-word regularization, label smoothing, domain adaptation.

6.3 Future Work

1. Does the use of model averaging affect the quality of translation?

2. Does the number of encoders and decoders of the Transformer improve the quality of translation and reduce hallucination?

3. Does the number of hidden layers/embedding size improve the quality of translations and reduce hallucinations?

7 Scientific Relevance for AI/HMC

Machine translation is a work in progress, on-going research is been done to find better ways to improve the quality of translated text both in terms of fluency and adequacy. The aim is to ensure that the contextual meaning of a text in one language matches with the same meaning in another foreign language. Also to make computers or robots independent during the translation process and to remove human intervention in the process.
Therefore, this research with out-domain-text will help improve translations from been generic to domain specific. It can be a gateway to improve the use of translations in various applications like chatbots trained on financial domain to be deployed effectively in financial institutions, as an online customer support capable of communicating in different languages.
Furthermore, In the medical industry there is also a growing trend in intelligent chatbot doctors, capable of communicating with a patient and giving medical advise without the use of an interpreter.
Also, the research will also be of benefit to wearable hearing devices for speech-to-speech translation, which may help communication between multilingual speakers.
Most importantly, The development of an AI-based digital encyclopedia that is knowledgeable in every domain and capable of communicating in multiple languages e.t.c
These are some of the few applications in AI that can benefit from my research contribution.

Acknowledgments

I am grateful to God for everything. I also wish to thank in a special way my supervisor Dr. Jennifer Spenader for her morale and technical support during the research, Also my second supervisor Dr. Kasaei Hamidreza.
Others researchers worthy of thanks are Professor Rico Sennrich from the University of Zurich also for technical support, Mayumi Ohta a Phd researcher at the Heidelberg University in Germany for her advice on using the JoeyNMT toolkit.

Bibliography

[1] P. Koehn and R. Knowles, "Six challenges for neural machine translation," in *Proceedings of the First Workshop on Neural Machine Translation*, (Vancouver), pp. 28–39, Association for Computational Linguistics, Aug. 2017.

[2] M. Müller, A. Rios, and R. Sennrich, "Domain robustness in neural machine translation," in *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, (Virtual), pp. 151–164, Association for Machine Translation in the Americas, Oct. 2020.

[3] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, (Philadelphia, Pennsylvania, USA), pp. 311–318, Association for Computational Linguistics, July 2002.

[4] M. Popović, "chrF: character n-gram F-score for automatic MT evaluation," in *Proceedings of the Tenth Workshop on Statistical Machine Translation*, (Lisbon, Portugal), pp. 392–395, Association for Computational Linguistics, Sept. 2015.

[5] P. Koehn and C. Monz, "Manual and automatic evaluation of machine translation between european languages," pp. 102–121, 06 2006.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017.

[7] C. Wang and R. Sennrich, "On exposure bias, hallucination and domain shift in neural machine translation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, (Online), pp. 3544–3552, Association for Computational Linguistics, July 2020.

[8] M. Ranzato, S. Chopra, M. Auli, and W. Zaremba, "Sequence level training with recurrent neural networks," *CoRR*, vol. abs/1511.06732, 2016.

[9] L. Han, "Lepor: An augmented machine translation evaluation metric," 07 2014.

[10] S. Banerjee and A. Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," in *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, (Ann Arbor, Michigan), pp. 65–72, Association for Computational Linguistics, June 2005.

[11] R. Haldar and D. Mukhopadhyay, "Levenshtein distance technique in dictionary lookup methods: An improved approach," *Computing Research Repository - CORR*, 01 2011.

[12] F. Rahutomo, T. Kitasuka, and M. Aritsugi, "Semantic cosine similarity," 10 2012.

[13] V. Raunak, A. Menezes, and M. Junczys-Dowmunt, "The curious case of hallucinations in neural machine translation," *CoRR*, vol. abs/2104.06683, 2021.

[14] S. Shen, Y. Cheng, Z. He, W. He, H. Wu, M. Sun, and Y. Liu, "Minimum risk training for neural machine translation," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, (Berlin, Germany), pp. 1683–1692, Association for Computational Linguistics, Aug. 2016.

[15] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," *ArXiv*, vol. abs/1508.07909, 2016.

[16] I. Provilkov, D. Emelianenko, and E. Voita, "BPE-dropout: Simple and effective subword regularization," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, (Online), pp. 1882–1892, Association for Computational Linguistics, July 2020.

[17] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a defense to adversarial perturbations against deep neural networks," in *2016 IEEE Symposium on Security and Privacy (SP)*, pp. 582–597, 2016.

[18] Z. Tu, Y. Liu, L. Shang, X. Liu, and H. Li, "Neural machine translation with reconstruction," *CoRR*, vol. abs/1611.01874, 2016.

[19] T. Kudo, "Subword regularization: Improving neural network translation models with multiple subword candidates," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, (Melbourne, Australia), pp. 66–75, Association for Computational Linguistics, July 2018.

[20] Y. Kim and A. Rush, "Sequence-level knowledge distillation," 06 2016.

[21] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014.

[22] Y. Gao, W. Wang, C. Herold, Z. Yang, and H. Ney, "Towards a better understanding of label smoothing in neural machine translation," in *AACL*, 2020.

[23] C. Chu and R. Wang, "A survey of domain adaptation for neural machine translation," in *Proceedings of the 27th International Conference on Computational Linguistics*, (Santa Fe, New Mexico, USA), pp. 1304–1319, Association for Computational Linguistics, Aug. 2018.

[24] J. Kreutzer, J. Bastings, and S. Riezler, "Joey nmt: A minimalist nmt toolkit for novices," 2019.

[25] K. Lee, O. Firat, A. Agarwal, C. Fannjiang, and D. Sussillo, "Hallucinations in neural machine translation," 2019.

[26] J. Yan, F. Meng, and J. Zhou, "Probing causes of hallucinations in neural machine translations," 06 2022.

Appendices

A Experimental Configurations and Hyper-parameter Optimization

Number of Attention heads:	8
Number of layers:	6
hidden size:	512
embedding dimension:	512
learning rate:	0.0003
learning rate factor:	0.5
learning rate warmp:	1000
loss function:	cross entropy
Optimizer:	Adam
Maximum sent. length:	100
Max output length:	60
evaluation metric:	BLEU
BPE dropout:	0.1
label smoothing:	0.1 or 0.0
encoder dropout:	0.3
decoder dropout:	0.2
beam size:	5
batch size:	4096
evaluation batch size:	3026
early stopping metric:	loss
No. of epochs:	45
output activation fxn	Softmax
feed-forward size	2048

B Combined BLEU scores across domains

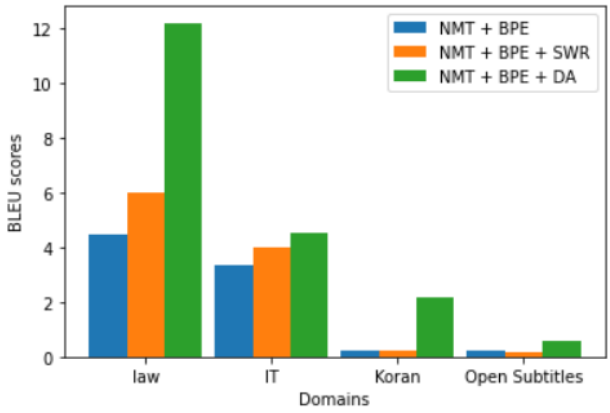


Figure 24: summary of BLEU scores across domains