# Chinese-to-English Translation with Transformer Models

**A0211218W, A0211207Y, A0211244X , A0218178X , A0240201H, A0239130N**
Group 51
Mentored by Tan Yong-Jia Naaman
{e0492488,e0492477,e0492514,e0544214,e0774799,e0773728}@u.nus.edu

## Abstract

The increasing need for cross-lingual communication have underscored the importance of high-quality machine translation systems. While the Transformer architecture has revolutionised machine-learning translation systems, there still exists many challenges in the Chinese-to-English translation task. In this work, using the IWSLT 2017 dataset, we identify three common challenges in Chinese-English translation using the Transformer architecture: sentences with Chinese idioms, with polyphonic/homophonic Chinese words, and with unseen tokens. By subclassing the test data, we then explore the effect of modifying our baseline model through hyperparameter tuning, character-level radical preprocessing, and adding document-level context attention to address each problem, and measured its effectiveness via BLEU and BERTScore as evaluation metrics. Experimental results demonstrate some positive results in the above directions to address these challenges, such as model embedding size on all three challenges, character-level radical preprocessing on unseen tokens and document-level context attention for polyphones.

## 1 Introduction

In today's increasingly interconnected world, the need for effective cross-lingual communication has become paramount. Machine translation (MT) systems play a crucial role in bridging linguistic barriers, enabling seamless information exchange and fostering global collaboration. In particular, the Chinese-English language pair can be particularly challenging due to the logographic nature of Chinese characters, as well as vastly different grammatical structure of Chinese and English.

Traditional statistical machine translation (SMT) approaches, which rely on parallel corpora and language models (Lopez, 2008), have made significant strides in addressing this challenge. However, the advent of deep learning and neural network-based architectures has paved the way for more powerful and effective translation models. Our initial exploration into Neural Machine Translation (NMT) yielded us the Recurrent Neural Networks (RNN) (Sherstinsky, 2020), and the Transformer architecture (Vaswani et al., 2017). We found that the Transformer architecture allows us to train models that outperform RNN models in terms of performance and training speed.

Several studies have explored the application of Transformer models for the Chinese-to-English translation task, demonstrating their effectiveness in handling the linguistic complexities involved. The use of dual learning and deliberation networks by Hassan et al. (2018) has achieved human parity when compared to professional human translations. However, the use of Transformer models for Chinese-English translation still presents several challenges.

Firstly, Chinese is a language rich in idioms and metaphors, many of which do not have direct translations in English. The translation of idioms and metaphors can be challenging especially when we are trying to preserve their intended meaning and context.

Secondly, both Chinese and English have many words with multiple meanings, and phonetic values. It may prove especially hard to discern the correct meaning that a Chinese character is trying to convey, and further look for its correct translation in English.

Thirdly, the increasing amount of vocabulary in Chinese and English, especially due to the rise of slang language and new terminology, poses a problem for translation. Transformer models may struggle with translating out-of-vocabulary (OOV) words and phrases, especially for domain-specific or technical terminology.

Our project attempts to investigate the effectiveness of various techniques in targeting these chal-

lenges. We aim to subclass our test data to measure the impact of each method against a baseline Transformer model using BLEU and BERTScore as evaluation metrics.

## 2 Background & Related Work

### 2.1 Transformers for NMT

The development of the Transformer architecture by Vaswani et al. (2017) introduced an effective self-attention mechanism that captures long-range dependencies in sequential data without the need for recurrent or convolutional layers. This resulted in widespread adoption of the Transformer architecture, with readily available models by Wolf et al. (2020) on the HuggingFace platform, which has largely replaced the use of recurrent neural networks (RNNs) in many NLP tasks.

### 2.2 Common Challenges in Chinese-English Translation

As mentioned in the introduction, we identify three main categories of challenges in Chinese-English translation, as well as existing efforts to solve these issues.

#### 2.2.1 Idioms and Metaphors

Idioms and metaphors, especially in Chinese, are considered highly problematic for a wide variety of NLP tasks. Shao et al. (2018) proposed a blacklist method to discourage the NMT from doing literal translations. However, the research on the use of the blacklist-based evaluation is framed as more of a classification problem in identifying literal translations, proving it to be beneficial for quality estimation only in a post-editing environment.

#### 2.2.2 Polyphones

Polyphonic words (also referred to as homographs) are also widely acknowledged to be challenging in NMT (Liu et al., 2017), and this is especially challenging in Chinese due to the large number of homographs present. Jin et al. (2023) showed that while context-aware NMT does not always help disambiguate certain discourse phenomena, the incorporation of context still helps to resolve many ambiguities such as pronoun resolution, and named-entity consistency. We found that the document-level context attention proposed by Zhang et al. (2018) is an effective implementation of context-aware NMT.

#### 2.2.3 Out-of-Vocabulary Tokens

Another common linguistic challenge faced in NMT is the translation of out-of-vocabulary words and tokens. Most systems traditionally cannot translate such words, but Liu et al. (2018) explores the use of subwords to perform translation, rather than entire words, with English as a source language. While this method is not directly applicable to the Chinese language, Han and Kuang (2018) explored a similar concept, integrating Chinese radicals to target the challenge of out-of-vocabulary words translation, breaking down individual characters into the semantic-carrying radicals to express the words they are constructed in. We will explore the use of this method in improving performance for unseen tokens.

#### 2.2.4 Evaluation Metrics

The evaluation of MT systems is a crucial aspect of assessing their performance, and guiding further improvements. Several automatic evaluation metrics have been proposed to consistently and efficiently measure the quality of translations. The BLEU Score proposed by Papineni et al. (2002) is one of the most widely used metrics for MT evaluation. However, its reliance on exact n-gram matches results in its inability to capture semantic equivalence on its translation results. Some of these limitations are addressed through METEOR by Banerjee and Lavie (2005), which incorporates stemming and synonym matching. Finally, the BERTScore by Zhang et al. (2020) is a recent metric that leverages contextual embeddings from BERT (Devlin et al., 2019) to evaluate the semantic similarity between outputs and reference translations. This metric has shown improved correlation with human judgments and is suitable for evaluating various language pairs like Chinese-English.

Our project aims to build upon previous research by Zhang et al. (2018) and Han and Kuang (2018), alongside with hyperparameter tuning techniques learnt in course lectures, to evaluate the effectiveness of various approaches in addressing the above problems in the Chinese-to-English translation context. We will be using the Chinese-English corpus from the IWSLT 2017 dataset[1] to train, validate and test our model. The evaluation of all models will be done with the BLEU Score and BERTScore.

[1]HuggingFace Repository: https://huggingface.co/datasets/iwslt2017

## 3 Corpus Analysis & Method

### 3.1 Dataset

This project makes use of the IWSLT 2017 dataset. Specifically, we use the Chinese-English subset of the dataset provided for the "Multilingual" task which contains ~241k English-Chinese sentence-level parallel pairs extracted from various TED talks (Cettolo et al., 2017). We make use of the provided Train [231266 rows], Test [8549 rows] and Val [879 rows] set split to perform our model training. It is worth noting that the dataset not just includes sentence-level parallel corpora, but because each set of sentences comes from a particular TED talk, each sentence pair is labelled with a corresponding document id. This will be useful for incorporating document-level context.

The dataset is not without its limitations. As identified by Cettolo et al. (2016), due to the extensive rehearsal of TED talks, the utterances are likely more well-formed than spontaneous speech, making the task less challenging than general dialogue translation. Another limitation of this dataset is that the source language of the TED talks in this case is in English, while the Chinese language sentences are actually created by human translators, setting an upper bound on the performance of a model trained to perform translation in the opposite direction.

Despite this, the dataset still demonstrates a number of challenging cases due to the vast difference between Chinese and English, which we will explore in the **Test Data** section below. While the raw BLEU and BERTScore may not be indicative, we can still observe the relative performance of each experiment to assess performance impact.

### 3.2 Data Preprocessing

We use the spaCy tokenizer to tokenize all sentences. The spaCy tokenizer is capable of handling both Chinese and English, as it employs language-specific rules and models to accurately break down text into meaningful units.

### 3.3 Test Data Subclassing

To investigate the performance impact of each model and preprocessing, we not only compute evaluation metrics of each model on the whole test set, but also subclass the test set into various subclasses representing each of the aforementioned challenges.

#### 3.3.1 Idioms and Metaphors

We filter all the test data where the Chinese translation contains an idiom. We obtain a list of Chinese idioms from Github pwxcoo (2019), extracted from the Xinhua Dictionary API. We will refer to this subset in our results as **[idioms]**.

**Test Set Example:**
*Chinese:* 如果一年以前你问我我可不会像现在这么[理直气壮] (Literal character-by-character translation: "logic straight air/energy strong", idiomatic meaning "in the right, self-confident")
*English:* Now if you'd have asked me that a year ago, I wouldn't have been able to tell you that [with any certainty].

#### 3.3.2 Polyphones

We filter all the test data where the Chinese translation contains a polyphone. We use a list of polyphonic Chinese words from Github (hjzin, 2019) to perform such filtering. We will refer to this set in our results as **[polyphones]**. Below, we attach an example of the same Chinese character [重] with two different meanings and pronunciations.

**Test Set Example 1:**
*Chinese:* 这个实验在很多不同的人身上[重(chóng)复]过，而且用了很多不同的图片，结果几乎总是一样。
*English:* This was [repeated] on lots of different individuals with lots of different images, always with a similar result

**Test Set Example 2:**
*Chinese:* 我将要说到的，非常[重(zhòng)要]。
*English:* And this is very [important], what I'm going to say.

#### 3.3.3 Out-of-Vocabulary Tokens

We filter all the test data where the Chinese translation contains an unseen token in the training set. We will refer to this set in our results as **[unseen tokens]**.

#### 3.3.4 Final Test Data Split

With this test data split, we can compare the performance improvement on each subclass, compared to

| Subset of test data | No. Rows |
|---|---|
| idioms | 391 |
| polyphones | 3867 |
| unseen tokens | 1903 |
| all | 8549 |

Table 1: Size of each subset of our Test Data.

the overall improvement, to evaluate if each model experiment helps with the task.

### 3.4 Model Architecture and Exploration

For comparison, we first train a baseline Transformer with an embedding size of 128, 4 attention heads and 4 encoder and decoder layers. These hyperparameters were suggested by Verma and Kolhatkar (2023) to be the best hyperparameters for their Transformer-based translation model. Though we acknowledge that these are likely not the best hyperparameters for our model, given our much smaller dataset and limited computational resources, it provides us with a starting point to generate reasonable results and compare further model improvements against. For the purpose of reproducibility, our source code is uploaded at `https://github.com/TTraveller7/cs4248-project`.

#### 3.4.1 Hyperparameter Tuning

To address the challenge of understanding words with multiple meanings, we experimented with the method of increasing the number of heads in the Transformer model from 4 to 8. The goal is for the larger number of attention heads to capture more relationships between pairs of words in the vocabulary, hopefully learning to differentiate the meaning of words in different contexts.

To address the challenge of translating idioms and metaphors while preserving their rich cultural context and complex semantics, often which cannot be inferred from their comprising words, we experimented with the method of increasing the embedding size of the Transformer model from 128 to 256. The goal is for the larger number of parameters to capture these rich features of idioms and metaphors, hopefully capturing a more accurate understanding of the Chinese source sentence and producing a better translation.

#### 3.4.2 Radical Preprocessing

To address the challenge of unseen (out of vocabulary) words translation, we adopted a method called radical preproecessing (Han and Kuang, 2019) to split Chinese character into different radicals. Chinese words are typically composed of multiple characters, and these characters can be further broken down into smaller units called radicals, which usually carry the semantic meaning of the word. In our method, we use the word along with radical as multiple inputs of our model. The goal is for the model to learn more useful features by integrating information from these different levels of the Chinese writing system and capture more valuable linguistic cues and patterns, leading to improved translation performance on unseen tokens.

The input embedding $x_i$ consists of two parts: word embedding $w_i$ and radical embedding $r_i$ as below:

$$x_i = [w_i; r_i]$$

where ';' is concatenate operation.

For word $w_i$, it can be split into multiple characters and further split into radicals $r_i = (r_{i1}, r_{i2}, ..., r_{ik})$. Thus, the radical embedding of word $w_i$ can be computed as:

$$r_i = \sum_{k=1}^{m} r_{ik}$$

where $k$ is the number of radicals constructing the word $w_i$.

#### 3.4.3 Document-Level Context Attention

To address the OOV and rare word problem, we implemented a variation of the basic transformer model with document-level context. We refactor the model following closely with the structure proposed by Zhang et al. (2018): For each sentence to translate, we encode its preceding sentences in the document into context encoding. Next, we incorporate the context encoding into both the encoder and decoder with multi-head attention Vaswani et al. (2017).

The document-level context method is applicable to our dataset, as the dataset consists the transcripts of of a series of TED talks (Cettolo et al., 2017), and each talk can be seen as a document. Let $X = x^{(1)}, ..., x^{(k)}, ..., x^{(n)}$ be a TED talk with n sentences. For the $k$-th sentence $x^{(k)}$, we could use the sentences on the left side of $k$-th sentence $X_{<k} = x^{(1)}, ..., x^{(k-1)}$, as its document-level context. Zhang et al. (2018) also suggested that using too many preceding sentences as the document-level context may not bring improvement and increases the computational workload. Therefore,

we maintained a context window, limiting each document-level context to at most 10 sentences.

## 4 Experiments

### 4.1 Baseline

The results of testing on our baseline Transformer model are as follows.

| Subset of test data | BERTScore | BLEU |
|---|---|---|
| idioms | 0.8697 | 0.0732 |
| polyphones | 0.8773 | 0.0934 |
| unseen tokens | 0.8664 | 0.0810 |
| all | 0.8887 | 0.1049 |

Table 2: BERTScore and BLEU score of translations generated from baseline Transformer model.

### 4.2 Hyperparameter Tuning

The results of testing on our Transformer model with increased number of attention heads are as follows.

| Subset of test data | BERTScore | BLEU |
|---|---|---|
| idioms | 0.8712 | 0.0801 |
| polyphones | 0.8790 | 0.0983 |
| unseen tokens | 0.8672 | 0.0837 |
| all | 0.8905 | 0.1113 |

Table 3: BERTScore and BLEU score of translations generated from Transformer model with more attention heads.

The results of testing on our Transformer model with increased embedding size are as follows.

| Subset of test data | BERTScore | BLEU |
|---|---|---|
| idioms | 0.8753 | 0.0839 |
| polyphones | 0.8836 | 0.1032 |
| unseen tokens | 0.8728 | 0.0913 |
| all | 0.8940 | 0.1150 |

Table 4: BERTScore and BLEU score of translations generated from Transformer model with increased embedding size.

### 4.3 Radical Preprocessing

The results of testing on our Transformer model with radical preprocessing are as follows.

| Subset of test data | BERTScore | BLEU |
|---|---|---|
| idioms | 0.8750 | 0.0793 |
| polyphones | 0.8837 | 0.1028 |
| unseen tokens | 0.8733 | 0.0932 |
| all | 0.8949 | 0.1154 |

Table 5: BERTScore and BLEU score of translations generated from Transformer model with radical preprocessing.

### 4.4 Document-level Context Attention

The results of testing on our Transformer model with document-level context are as follows.

| Subset of test data | BERTScore | BLEU |
|---|---|---|
| idioms | 0.8735 | 0.0840 |
| polyphones | 0.8822 | 0.1042 |
| unseen tokens | 0.8696 | 0.0882 |
| all | 0.8931 | 0.1162 |

Table 6: BERTScore and BLEU score of translations generated from Transformer model with document-level context.

## 5 Discussion

### 5.1 Tuning Hyperparameters

The percentage changes in BERTScores and BLEU on translations generated by the models with increased number of heads and embedding size, against the baseline model, were calculated as shown in the tables below.

| Subset of test data | Change in BERTScore (%) | Change in BLEU (%) |
|---|---|---|
| idioms | 0.1725 | 9.4718 |
| **polyphones** | **0.1938** | **5.2113** |
| unseen tokens | 0.0923 | 3.3726 |
| all | 0.2025 | 6.059 |

Table 7: Changes (%) in BERTScore and BLEU score of translations generated from model with more attention heads against baseline model.

| Subset of test data | Change in BERTScore (%) | Change in BLEU (%) |
|---|---|---|
| **idioms** | **0.6439** | **14.6489** |
| polyphones | 0.7181 | 10.4223 |
| unseen tokens | 0.7387 | 12.6451 |
| all | 0.5964 | 9.6148 |

Table 8: Changes (%) in BERTScore and BLEU score of translations generated from model with increased embedding size against baseline model.

Generally, both increasing the number of attention heads and embedding size do indeed improve the models' performance in translating Chinese to English sentences, as seen by the positive change in BERTScores and BLEU when tested on the entire test dataset. An increase in BERTScore indicates that the generated translation is better at capturing the meaning of the original text, as it is closer in semantics to the reference translation. An increase in BLEU indicates that the wordings of the generated translation is more similar to the reference translation, with greater ngram overlaps and smaller difference in length.

Increasing the embedding size of the Transformer model increases the number of parameters, allowing the model to capture more features in the data. In the context of NLP, increasing the embedding size enables bigger embeddings that capture more details about a sentence, including but not limited to finer grained semantic meanings, broader context, and structural information like syntax, all of which are key to how a sentence can be understood and produced. As such, it is as expected that the model generates more sophisticated translations with a richer understanding of the source sentence and target language.

Each attention head in the Transformer learns one relationship between each pair of words in the vocabulary. However, language is complicated and many words can have multiple meanings, play multiple part of speech roles, etc, and hence have multiple relationships to other words in the vocabulary. For example, the Chinese word [好] can be a verb ("to like") or an adjective ("good") in a sentence and learning just one relationship using one attention head is not enough. Multiple attention heads will be needed to capture each relationship for a more robust understanding of the nuances in a word's role in a sentence. Hence, it is also expected that increasing the number of attention heads in the model generates better translations.

It is interesting to note that increasing the embedding size results in a bigger improvement in model performance as compared to increasing the number of heads. From here, we can perhaps conclude that the limiting factor of our baseline model is that its original features were too simplified to give a good representation of the English and Chinese language. Even with better capability to learn different relationships between words, as provided by increased number of attention heads, such features were already lost in the compression of words into its overly-simplified vector representation.

#### 5.1.1 Does increasing number of attention of heads improve translation of sentences containing polyphones?

In our methodology, we hypothesised that increasing the number of attention heads would improve the translation of polyphone rich sentences, due to the models' increased ability to understand the contextual meaning of words. Though we observe a positive change of 0.1938% in BERTScore and 5.2113% in BLEU using a model with more heads, this is lower than the average score of 0.2025% and 6.059% respectively when tested with the entire test dataset. Hence, we cannot attribute the improvement in scores to the models' enhanced ability to understand words with multiple meanings, as opposed to just a general improvement in the models' translation abilities.

As explained earlier, we believe that the performance of the model with increased heads is limited by its smaller embedding size. An even lower improvement in translation quality of sentences containing polyphones likely suggests that these polyphones are even more complex to represent as vectors, compared to the average word, and hence suffer more from the limitations of a small embedding size. It also suggests that converting a word into a vector of size 128 likely loses the particular features that captures information about its diverse range of meanings.

#### 5.1.2 Does increasing embedding size improve translation of sentences containing idioms?

In our methodology, we hypothesised that increasing the embedding size would improve the translation of idioms, due to the models' increased ability to capture more features about its nuanced semantics and cultural context. This is supported by our observation, as we see a large improvement of

0.6439% and 14.6489% in both BERTScore and BLEU respectively, greater than the average improvement of 0.5964% and 9.6148% when tested with the entire test set.

### 5.2 Radical Preprocessing

The percentage changes in BERTScores and BLEU on translations generated by the models with radical preprocessing, against the baseline model, were calculated as shown in the tables below.

| Subset of test data | Change in BERTScore (%) | Change in BLEU (%) |
|---|---|---|
| idioms | 0.6439 | 8.3333 |
| polyphones | 0.7295 | 10.0642 |
| **unseen tokens** | **0.7964** | **15.0617** |
| all | 0.6976 | 10.0095 |

Table 9: Changes (%) in BERTScore and BLEU score of translations generated from Transformer model with radical preprocessing against baseline model.

Using radical-level segmentation in Chinese text processing would help in handling unseen tokens. Unseen tokens refer to words or characters that do not appear in the training dataset. Since Chinese radicals carry essential meanings of the characters they are composed of, segmenting text at the radical level can assist the model in understanding the meaning and structure of unseen tokens. By learning representations of radicals during training, the model can decompose unseen tokens into radicals and infer their meanings based on the meanings of the radicals. This approach enhances the model's ability to handle unseen tokens, thereby improving the robustness and generalization capability of machine translation systems.

#### 5.2.1 Does radical preprocessing improve translation of sentences containing unseen words?

In our methodology, we hypothesised that incorporating the radical preprocessing would improve the translaton of unseen tokens, due to the ability of radicals to capture more semantic information about Chinese characters. Our experiments validated this hypothesis, as we observed a large change of 0.7964% in BERTScore and 15.0617% in BLEU for the translation of unseen words after integrating the radical preprocessing, which achieves the highest improvement in BLEU score and BERTScore for translation of unseen words. This suggests that the radical-level inputs indeed helped the model better understand the underlying structure and semantic of Chinese characters, resulting in more accurate translation of unseen words.

### 5.3 Document-Level Context Attention

The percentage changes in BERTScores and BLEU on translations generated by the models with document-level context attention, against the baseline model, were calculated as shown in the tables below.

| Subset of test data | Change in BERTScore (%) | Change in BLEU (%) |
|---|---|---|
| idioms | 0.4369 | 14.7540 |
| **polyphones** | **0.5585** | **11.5631** |
| unseen tokens | 0.3693 | 8.8889 |
| all | 0.4951 | 10.7721 |

Table 10: Changes (%) in BERTScore and BLEU score of translations generated from Transformer model with document-level context attention against baseline model.

The model with document-level context attention achieves the highest improvement in the corpus BLEU score compared to the other two variations. However, it achieves the lowest improvement in BERTScore. This may indicate that the context attention usually generates a translation more similar to the reference, but fails to capture the semantics of the original text as much as the other variations.

The root cause of this phenomenon is probably overfitting. The model has an additional context encoder as well as additional context attention layers in both encoders and decoders. These additional substructures and layers utilize siginificantly more parameters compared to the other variations.

#### 5.3.1 Does document-level context attention improve translation of sentences containing polyphones?

In our methodology, we hypothesized that using document-level context attention can help to resolve the ambiguity of polyphones. Our experiments validated the hypothesis, as we observed a 0.5585% increase in BERTScore and 11.5631% increase in BLEU score, and the improvements are higher than the improvements on the entire data set.

However, our model with context attention

achieves the highest BLEU score improvement among all three variations, but also achieves the lowest BERTScore improvement. This indicates that the phenomenon of getting higher BLEU score improvement but lower BERTScore improvements also occurs with the polyphones.

#### 5.3.2 Does document-level context attention improve translation of sentences containing unseen tokens?

In our methodology, we hypothesized that using document-level context attention can help to resolve the problem of unseen words, as we assume that a model with context could infer the meaning of the unseen words from their context. However, this hypothesis is not proven by our experiment outcome. The model with context achieves a 0.3693% increase in BERTScore and 8.8889% increase in BLEU score over the unseen words, and both of the improvements are less significant than the improvements made by the other variations over the same subset.

## 6 Conclusion

In this project, we investigate the performance impact of hyperparameter tuning, incorporating document-level context attention, as well as character-level radical preprocessing relative to a baseline Transformer model on the Chinese-to-English translation task. In particular, we evaluate performance on three challenging subcategories: sentences with Chinese idioms, with polyphonic/homophonic Chinese words, and with unseen tokens. Some of our experiments showed positive results, such as increasing embedding size in the embedding task helping with all three subclasses, radical preprocessing helping with the unseen token subclass, and document-level context attention on the polyphone task.

Due to the limited size of compute available and the size limitations of our dataset, we were unable to investigate if the various disproven hypotheses were due to limitations in the size of model or dataset. Due to time constraints, we also did not experiment with other forms of tokenization besides spaCy's default behaviour, especially for Chinese, which can impact the performance of our model – experiments here, applied with character-level radical preprocessing can be a direction for future exploration to improve performance in the unseen token task.

## References


Satanjeev Banerjee and Alon Lavie. 2005. 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*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Mauro Cettolo, Marcello Federico, Luisa Bentivogli, Jan Niehues, Sebastian Stüker, Katsuitho Sudoh, Koichiro Yoshino, and Christian Federmann. 2017. Overview of the iwslt 2017 evaluation campaign. In *Proceedings of the 14th International Workshop on Spoken Language Translation*, pages 2–14.

Mauro Cettolo, Jan Niehues, Sebastian Stüker, Luisa Bentivogli, Rolando Cattoni, and Marcello Federico. 2016. The iwslt 2016 evaluation campaign. In *Proceedings of the 13th International Conference on Spoken Language Translation*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Lifeng Han and Shaohui Kuang. 2018. Incorporating chinese radicals into neural machine translation: Deeper than character level. *arXiv preprint arXiv:1805.01565*.

Lifeng Han and Shaohui Kuang. 2019. Incorporating chinese radicals into neural machine translation: Deeper than character level.

Hany Hassan, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, Christian Federmann, Xuedong Huang, Marcin Junczys-Dowmunt, William Lewis, Mu Li, Shujie Liu, Tie-Yan Liu, Renqian Luo, Arul Menezes, Tao Qin, Frank Seide, Xu Tan, Fei Tian, Lijun Wu, Shuangzhi Wu, Yingce Xia, Dongdong Zhang, Zhirui Zhang, and Ming Zhou. 2018. Achieving human parity on automatic chinese to english news translation.

hjzin. 2019. Polyphonedisambiguation. https://github.com/hjzin/PolyphoneDisambiguation/.

Linghao Jin, Jacqueline He, Jonathan May, and Xuezhe Ma. 2023. Challenges in context-aware neural machine translation.

Frederick Liu, Han Lu, and Graham Neubig. 2017. Handling homographs in neural machine translation. *arXiv preprint arXiv:1708.06510*.

Nelson F Liu, Jonathan May, Michael Pust, and Kevin Knight. 2018. Augmenting statistical machine translation with subword translation of out-of-vocabulary words. *arXiv preprint arXiv:1808.05700*.

Adam Lopez. 2008. Statistical machine translation. *ACM Comput. Surv.*, 40(3).

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

pwxcoo. 2019. chinese-xinhua. https://github.com/pwxcoo/chinese-xinhua/.

Yutong Shao, Rico Sennrich, Bonnie Webber, and Federico Fancellu. 2018. Evaluating machine translation performance on chinese idioms with a blacklist method.

Alex Sherstinsky. 2020. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena*, 404:132306.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Luv Verma and Ketaki N. Kolhatkar. 2023. Optimizing transformer-based machine translation model for single gpu training: a hyperparameter ablation study.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing.

Jiacheng Zhang, Huanbo Luan, Maosong Sun, Feifei Zhai, Jingfang Xu, Min Zhang, and Yang Liu. 2018. Improving the transformer translation model with document-level context. *arXiv preprint arXiv:1810.03581*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert.


## Acknowledgements


We would like to thank our project mentor Yong-Jia Naaman Tan, and lecturer Min-Yen Kan for his support and guidance.


## Statement of Independent Work

1A. Declaration of Original Work. By entering our Student IDs below, we certify that we completed our assignment independently of all others (except where sanctioned during in-class sessions), obeying the class policy outlined in the introductory

lecture. In particular, we are allowed to discuss the problems and solutions in this assignment, but have waited at least 30 minutes by doing other activities unrelated to class before attempting to complete or modify our answers as per the class policy.

We have documented our use of AI tools (if applicable) in a following table, as suggested in the NUS AI Tools policy[2]. This particular document did not use any AI Tools to proofcheck and was constructed and edited purely by manual work.

If the production of your report used AI Tools (inclusive of Generative AI), do keep detailed logs of how you used AI Tools, as your project requires the accountability of an audit trail of your interaction(s) with such tools (prompts, output).

Signed,
A0211218W, e0492488
A0211207Y, e0492477
A0211244X, e0492514
A0218178X, e0544214
A0240201H, e0774799
A0239130N, e0773728

[2] https://libguides.nus.edu.sg/new2nus/acadintegrity, tab "AI Tools: Guidelines on Use in Academic Work"