Masked Word Prediction Differences for German-English RTT Models and German Models

Manoush Pajouh
Vassar College
mpajouh@vassar.edu

Abstract

This project investigates how round-trip translation (RTT) affects a model's ability to fill in and predict a random masked word of a German sentence through two separate models: one that is trained and tested on German sentences only, and the other that first masks the sentence in German, and then uses RTT to translate the German sentence into English, predict the masked word of the sentence using a model trained with English sentences, and then translates the sentence back to German. This word is then compared to the true word of the original sentence and to the predicted word of the German model. I hope to determine how accurate these predictions are to investigate how RTT may affect the model to discover whether this technique may be effective for masked word prediction models for lower-resource languages.

1 Introduction

Round-trip translation (RTT) is a popular technique for developing multilingual models, utilizing a high-resource language to support the technological tasks of a lower-resource language. German is a relatively high-resource language; however, it is still lower-resource than English, which is why this project aims to study how RTT affects a model's ability to predict a token of a sentence. Rather than using this approach as an evaluation tool or to discuss translatability (Somers, 2005), I use RTT for masked language modeling. The German language has different sentence structures and word order from English sentences, causing issues with alignment in terms of which words are masked. For example, the last word of every sentence cannot be masked since, in the English translation of the German sentence, the same word may not be at the end of the sentence anymore. To account for these issues, I mask the dataset upfront and then pass the masked dataset into the models, rather than masking the tokens during the evaluation process. (See Section 3.6 Evaluation). By analyzing the types of words predicted and the model's ability to predict the last word of the sentence, I aim to highlight the differences between a standard German model and the RTT model, to ultimately determine if the RTT model proves to be more accurate. To explore this, I implement two models: a BERT masked language model trained on German sentences and a BERT masked language model trained on English sentences. Both of these models are pre-trained, but I fine-tune them to the German and English versions of the subset of the OPUS dataset, respectively. I then mask the German dataset. During testing, the German model takes in a German sentence with the masked token already in place and then predicts the masked token. The English model takes in a German sentence with the mask token already in place, uses a translation model to convert the sentence into English with the mask token effectively untouched by the translation, predicts the masked word in English, replaces the masked token with the prediction, and then translates the entire sentence back to German. Using these two models, I aim to understand how RTT impacts word prediction by examining each model's accuracy of the predictions it generates. From here, I determine this to be an unreasonable approach to masked language modeling, considering the baseline I conduct, and for languages of different sentence structures, such as German and English. However, this is partly because my results are compared to a purely German model, which is pretrained on a significant amount of data. For lower-resource languages where such models are not an option to work off of, using RTT models would not be as accurate or successful as a monolingual model, but would still suffice.

2 Related Work

Much work explores cross-lingual transfer learning, usually via multilingual pretraining. Models

like mBERT, XLM, and XLM-R learn shared representations across languages and enable zero-shot transfer on downstream tasks (Pires et al., 2019; Lample and Conneau, 2019; Conneau et al., 2020; Choenni et al., 2023; Wu and Dredze, 2019). These models are explicitly trained on multilingual data, often with objectives designed to encourage alignment between languages. In contrast, recent work has demonstrated that language models are capable of translation-like behavior even in the absence of an explicit machine translation objective, implying that semantic alignment can arise implicitly from large-scale training (Alves et al., 2023). However, most prior work concerns multilingual architectures, leaving open the question of whether monolingual models are able to take advantage of cross-lingual context when parallel data is available only at inference time. This paper contributes to that question by probing English-to-German semantic transfer in the absence of any sharing across languages at training time. In fact, parallel corpora have seen wide usage in analyzing linguistic representations across neural models aside from traditional machine translation. Previous studies have probed whether language models encode syntactic structure, semantics, and hierarchical relationships using probing tasks, among many others (Linzen et al., 2016; Hewitt and Manning, 2019). Under cross-lingual settings, parallel data allows controlled comparisons of semantic equivalence across languages. Instead of training models directly on parallel text, this work leverages the OPUS corpus of English–German sentence pairs as an evaluation framework, allowing me to isolate how contextual information in one language influences predictions in another language, if at all. Previous pivot translation studies ground my approach in a similar indirect cross-lingual transfer (Imamura et al., 2023; Talwar and Laasri, 2025). For morphologically rich languages such as German, word prediction tasks are considerably more challenging due to inflection, agreement, and compounding, making them a useful test for analyzing semantic versus surface modeling. Predicting a masked word in a sentence constitutes an evaluation of cross-lingual semantic transfer that is focused and interpretable. There is a plethora of prior work on multilingual representation learning and zero-shot transfer in explicitly multilingual models. Previous work has investigated next-word generation in language-specific settings (e.g., for low-resource languages like Yoruba (Aliyu, 2024)). This study builds on such experiments to understand semantic relationships between the German and English languages and begins to determine if RTT models can overcome differences in sentence structure through simple word masking.

3 Methods

My goal is to investigate whether RTT models are accurate in predicting a masked token within a sentence. This section describes my data, models, and experiments. See Section 5 Limitations and Biases for more information and considerations.

3.1 German-English Parallel Corpus

I am using a subset of the OPUS Open Subtitles Corpora, which consists of translated movie subtitles (Lison and Tiedemann, 2016). I am specifically using the subtitles that are translated between German and English, though the corpora contains various languages to choose from for other translation pairings. The models use the first released version ("v1") of the dataset, which contains 70,534 sentences in total. For the sake of limited training and testing time available for the models, I am using roughly 20% of these 70,534 and then filtering out any sentences comprised of a single word to ensure the model has context to predict the masked word (See Section 3.4.1 Processing Data). The OPUS corpus is immensely helpful for providing parallel structures of sentences, which allows for easier comparisons of model predictions and lowering of bias due to content (See Section 5 Limitations and Biases).

3.2 Models

This project utilizes two open-source BERT models available on HuggingFace (Devlin et al., 2018a). One is a BERT model designed for English sentences (Devlin et al., 2018b) and the other for German sentences (Chan et al., 2020). These models are pretrained and used via the same Transformers library from Hugging Face. These models have different model identifiers for particular tasks and have each been fine-tuned for the specific OPUS data. This study also utilizes the open source translation model Helsinki-NLP English to German and German to English models, also available on Hugging Face (Tiedemann and Thottingal, 2020).

3.2.1 German BERT

For German word predictions, I will be utilizing a version of BERT, which is trained on German

sentences (Chan et al., 2020). The tokenizer for this model tokenizes sentences into not just words, but also common prefixes and suffixes.

3.2.2 English BERT

For English word predictions, I have implemented a version of BERT available on HuggingFace, which is trained on English sentences (Devlin et al., 2018b). The tokenizer for this model tokenizes sentences into not just words, but also common prefixes and suffixes.

3.3 Helsinki-NLP German-English

The masked German testing sentences are translated into English using the Helsinki-NLP/opus-mt-de-en pretrained model (Tiedemann and Thottingal, 2020). This model is fine-tuned with the English subtitles from the OPUS corpus (Lison and Tiedemann, 2016).

3.3.1 Helsinki-NLP English-German

The predicted English words are then translated back into German using the Helsinki-NLP/opus-mt-en-de pretrained model (Tiedemann and Thottingal, 2020). This model is fine-tuned with the English subtitles from the OPUS corpus (Lison and Tiedemann, 2016).

3.4 German Masked Language Model

This model handles German data only. It is trained on German sentences, and used in evaluation to predict the masked token within a German sentence.

3.4.1 Processing Data (German)

The data for this model consists of German sentences extracted from the OpenSubtitles corpus. Each line in the dataset file corresponds to a single sentence. The dataset is preprocessed by reading the file and is then split into individual strings. To evaluate model generalization, the dataset is split into training and testing subsets: 80% and 20% respectively. Both subsets are stored as HuggingFace DatasetDict objects, with the column "text" representing each sentence.

3.4.2 Tokenization (German)

I used the google-bert/bert-base-german-cased tokenizer from HuggingFace. The tokenizer converts the raw sentences into token IDs for input to the BERT model. Since the model does not define a padding token by default, the tokenizer's eos_token was used as a padding token to ensure uniform sequence lengths. The tokenization function was applied in batch mode to both the training and the testing datasets using HuggingFace's .map() method. This produced tokenized inputs containing input_ids and attention_mask required for model training.

3.4.3 Training (German)

I then used the google-bert/bert-base-german-cased masked language model, which I will refer to as German BERT. This is a masked language model with multiple self-attention layers, trained to predict the masked token in a sentence. I used the model's AutoModelForMaskedLM implementation from HuggingFace for masked word prediction tasks. I used the DataCollatorForLanguageModeling class for batch training. This data collator pads each batch to the maximum sequence length within the batch and prepares input tensors for masked language modeling. The model was then trained using the HuggingFace Trainer API with the following details: batch size: 2 per device, with gradient accumulation of 4 steps (effective batch size 8), number of epochs: 3, learning rate: 5×10^5, weight decay: 0.01, warmup steps: 100, logging steps: 50, checkpoint saving steps: 500. The Trainer handled batching, gradient computation, and evaluation on the testing set. To monitor performance, the evaluation loss was computed at regular intervals. All processing, tokenization, and model training are implemented in Python 3.10 using the Transformers and Datasets libraries. The tokenizer, model, and collator were initialized from the same pretrained German GPT-2 checkpoint to ensure consistency.

3.5 RTT Prediction Model

This model uses round-trip translation in order to investigate how well an English model can ultimately predict masked tokens for German sentences.

3.5.1 Processing Data (RTT)

The data for this model consists of the English sentences extracted from the OpenSubtitles corpus and is preprocessed in the same manner as the German MLM (See Section 3.4.1 Processing Data)

3.5.2 Tokenization (RTT)

I used the google-bert/bert-base-cased tokenizer from HuggingFace. The tokenizer converts the raw sentences into token IDs for input to the BERT model. Since the model does not define a padding token by default, the tokenizer's eos_token was used as a padding token to ensure uniform sequence lengths. The tokenization function was applied in

batch mode to both the training and the testing datasets using HuggingFace's .map() method. This produced tokenized inputs containing input_ids and attention_mask required for model training.

3.5.3 Training (RTT)

I then used the google-bert/bert-base-german-cased masked language model, which I will refer to as English BERT. This is a masked language model with multiple self-attention layers, trained to predict the masked token in a sentence. I used the model's AutoModelForMaskedLM implementation from HuggingFace for masked word prediction tasks. I used the DataCollatorForLanguageModeling class for batch training. This data collator pads each batch to the maximum sequence length within the batch and prepares input tensors for masked language modeling. The model was then trained using the HuggingFace Trainer API with the following details: batch size: 2 per device, with gradient accumulation of 4 steps (effective batch size 8), number of epochs: 3, learning rate: 5×10÷5, weight decay: 0.01, warmup steps: 100, logging steps: 50, checkpoint saving steps: 500. These are the same details as for the German MLM. All processing, tokenization, and model training are implemented in Python 3.10 using the Transformers and Datasets libraries.

3.6 Evaluation

I constructed an evaluation dataset by randomly masking a single word in each sentence of the German dataset. Sentences are first preprocessed by stripping punctuation and removing samples containing fewer than three tokens. For each remaining sentence, one word is selected uniformly at random and replaced with the token "[MASK]" which is defined by the German BERT. To ensure compatibility with the BERT masked language modeling objective, only words that correspond to a single token under the German BERT tokenizer are considered; sentences in which the selected word is split into multiple subword tokens are discarded. The original word and its corresponding token ID are retained in the German BERT vocabulary to enable exact-match evaluation.(See Figure 1 for a flowchart depicting the pipeline). The model is evaluated by predicting the masked token and computing Top-k accuracy, where a prediction is considered correct if the original token appears among the k most probable tokens predicted at the masked position. This is done for k=1, k=5, and k=10.

To evaluate English masked language modeling, masked German sentences are first translated into English using the neural machine translation model (Tiedemann and Thottingal, 2020). To preserve the masked position during translation, the "[MASK]" token is temporarily replaced with the placeholder string "ZZZMASKZZZ" prior to translation and restored afterward. The placeholder can be altered, but be sure to use a token the translation model will preserve. Any translated sentences in which the masked token is not preserved are excluded from evaluation (See Section 5 Limitations and Biases). In doing so, I avoid translating the wrong word entirely due to alignment issues. For each English masked sentence, the English BERT model is used to predict the Top-k candidate tokens for the masked position. Each candidate token is then put back into the English sentence to fully reconstruct the sentence. This is then translated back into German using the Helsinki-NLP-mt-en-de model (Tiedemann and Thottingal, 2020). The prediction is counted as correct if the original German masked word appears in the back-translated sentence. I use Top-k accuracy to measure the proportion of sentences for which at least one of the Top-k English predictions successfully recovers the original German word after round-trip translation. This is done for k=1, k=5, and k=10.

4 Results

Figure 2 reports the Top-k masked token prediction accuracy for both the German MLM and the English round-trip translation (RTT) model. The German MLM had an accuracy rate of 42.68%, a top-5 accuracy rate of 66.63% and a top-10 accuracy rate of 74.16%. These results showcase that the German BERT model can predict masked tokens well, and substantially improves as the number of accepted predictions increases. The RTT model had an accuracy rate of 14.29%, a top-5 accuracy rate of 28.57%, and a top-10 accuracy rate of 42.86%. The German MLM is by far more successful consistently. The RTT model's accuracy scores improve as k increases, but the overall scores are much poorer than those of the German MLM. This gap in performance highlights the difficulty and noise when introducing cross-lingual transfer and back-translation. This is mainly because errors accumulate across the translation and prediction stages. This suggests that correct predictions are often among lower-ranked options but fail to appear as the single most-probable choice.

5 Limitations and Biases

This section discusses the limitations of these models, the biases present in the data and models, and any issues with the experiment overall.

5.1 Corpus

This model is limited by the nature of translated corpora as a whole, meaning that there is bias and inconsistency between the two versions of the dataset – English and German – that the models have been trained on. Furthermore, the dataset is limited to movie subtitles, so it will not be neither extremely formal nor informal. Just as it may not represent natural human language and slang entirely accurately, it will not represent formal vocabulary, such as medical or legal. As with any translated set, but especially for subtitles which prioritize capturing tone and emotion, more or less context may be necessary within each translation. Because of this, translations will not be direct, word-for-word translations.

Additionally, for faster training and testing times, I use a subset of the dataset: this subset is limited to the first 20% of the sentences as they appear rather in random order. I also filter out sentences that contain only one word to give the model context within the sentence. This filtering introduces some bias, since the sentences are not randomly shuffled on my part, despite the sentences not being in any particular order.

5.2 BERT Models

Any biases present in the models incorporated into this study are naturally going to have the same bias for this case study.

5.3 Translation Models

In the RTT model, by translating sentences with the masked token and then with the filled-in prediction, there is a chance that the translation model is changing the sentence rather than reverting it to the exact German sentence. This is especially a concern since BERT models treat prefix and suffix tokens as individual word tokens. Furthermore, I filter out sentences that, once the mask has been applied and the sentence is translated back to German, no longer have the mask in the sentence due to translation errors. This could be because the sentence no longer needs the mask to make sense (i.e. "Ich habe den Film gesehen" → "Ich habe den [MASK] gesehen" → "Ich habe den gesehen").

5.4 Evaluation

Due to the translation issues in which some sentences may not contain the masking token at all, I have filtered out these sentences in the evaluation stage. This, however, means that the German and English versions of the models are not being tested on exactly the same sentences in terms (since the RTT model translates them and some data may be noisy), though they are consistent in content, and not the same quantity of sentences (since there is more filtering done). This may be a contributing factor as to why the accuracy rates increase at a proportional rate.

6 Conclusion

This study compares monolingual masked language modeling performance with a cross-lingual round-trip translation (RTT) evaluation framework. The results demonstrate that while the German MLM performs reliably in a monolingual setting, an RTT model is less successful with its predictions. This degradation illuminates the challenges of translation ambiguity, morphological variation between German and English, and the semantic issues introduced by translating predicted tokens across languages. While the accuracy results are lower, the RTT results are still meaningful, particularly at higher k values. This means the model frequently finds plausible candidates even if the exact original word is not weighted the highest. These findings suggest that RTT-based evaluation provides a realistic assessment of cross-lingual robustness than monolingual evaluation alone, but requires more work to ensure alignment between sentences and proper evaluation tactics. Future work may improve RTT performance by normalizing data to account for morphological issues or measuring word predictions using cosine similarities to evaluate the monolingual and RTT model predictions alongside each other. Additionally, batching translation steps and exploring alternative placeholder strategies may further stabilize RTT evaluation.

7 Tables and figures

References

Ednah Olubunmi Aliyu. 2024. A deep learning approach for yoruba language next word generation. *2024 IEEE 5th International Conference on Electro-Computing Technologies for Humanity (NIGERCON)*, pages 1–4.

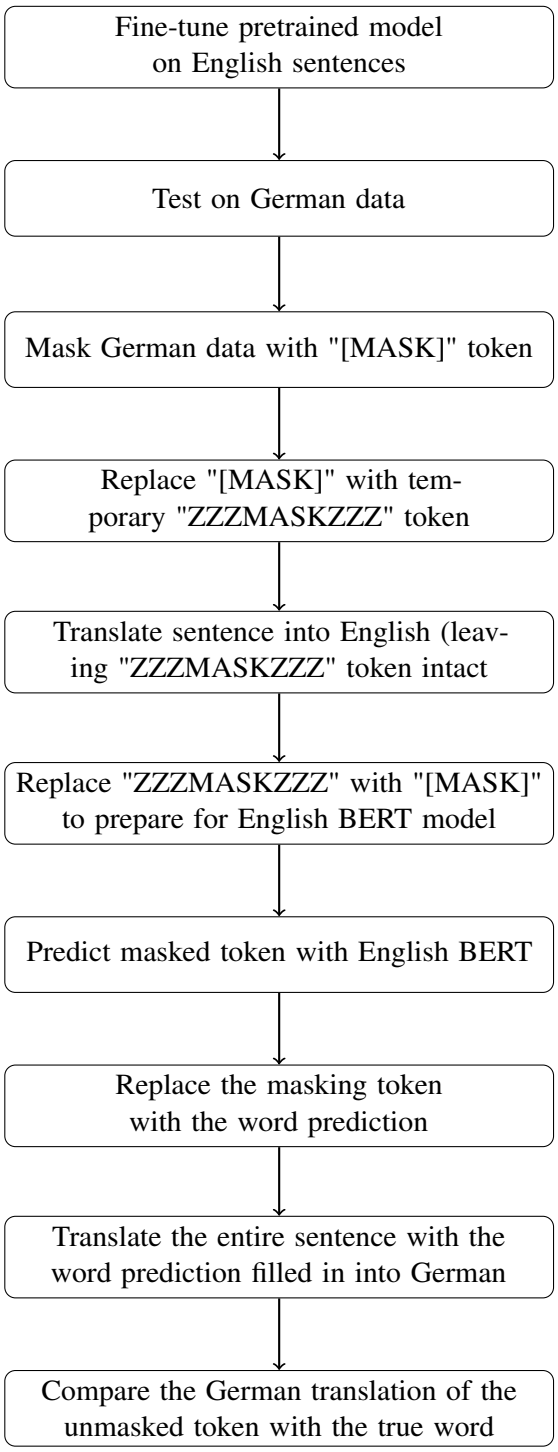

Figure 1: Diagram representing the masked word prediction pipeline for the RTT model

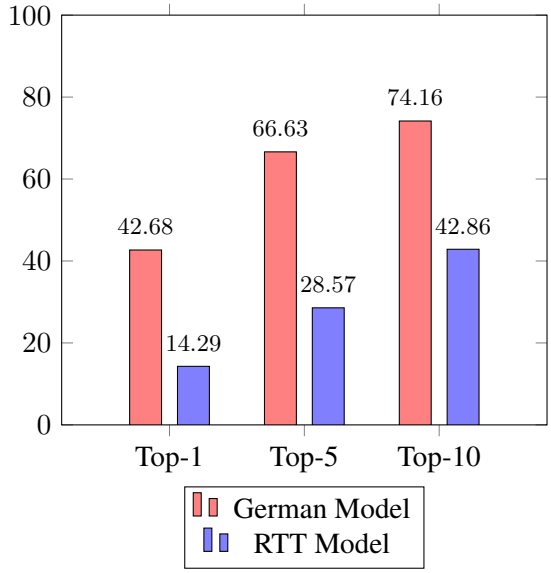

Figure 2: Accuracy Scores for both German and RTT models using 20% of the overall dataset. Both models were tested with 20% of that subset (4% of the overall data), and the English model is likely to have more filtering done to its sentences. See Limitations and Biases for more information.

Duarte Alves, Nuno Guerreiro, João Alves, José Pombal, Ricardo Rei, José de Souza, Pierre Colombo, and Andre Martins. 2023. Steering large language models for machine translation with finetuning and in-context learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11127–11148, Singapore. Association for Computational Linguistics.

Brandon Chan, Timo Möller, Malte Pietsche, and Tanay Soni. 2020. bert-base-german-cased (Revision 43cce13).

Rochelle Choenni, Dan Garrette, and Ekaterina Shutova. 2023. Cross-lingual transfer with language-specific subnetworks for low-resource dependency parsing. *Computational Linguistics*, 49(3):613–641.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018a. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018b. BERT: pre-training of

deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of NAACL-HLT*, pages 4129–4138.

Kenji Imamura, Masao Utiyama, and Eiichiro Sumita. 2023. Pivot translation for zero-resource language pairs based on a multilingual pretrained model. In *Proceedings of Machine Translation Summit XIX, Vol. 1: Research Track*, pages 348–359, Macau SAR, China. Asia-Pacific Association for Machine Translation.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *Preprint*, arXiv:1901.07291.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the ability of lstms to learn syntax-sensitive dependencies. In *Transactions of the Association for Computational Linguistics*, volume 4, pages 521–535.

Pierre Lison and Jörg Tiedemann. 2016. OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 923–929, Portorož, Slovenia. European Language Resources Association (ELRA).

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Harold L. Somers. 2005. Round-trip translation: What is it good for? In *Australasian Language Technology Association Workshop*.

Abhimanyu Talwar and Julien Laasri. 2025. Pivot language for low-resource machine translation. *ArXiv*, abs/2505.14553.

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT — Building open translation services for the World. In *Proceedings of the 22nd Annual Conferenec of the European Association for Machine Translation (EAMT)*, Lisbon, Portugal.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.

Acknowledgments

This paper was written using the template created by Association for Computational Linguistics[1].

[1]https://github.com/acl-org/acl-style-files/