
How Effective is Synthetic Data and Instruction Fine-tuning for Translation with Markup using LLMs?

Raj Dabre NICT, Japan

Haiyue Song NICT, Japan

Miriam Exel SAP, Germany

Bianka Buschbeck SAP, Germany

Johannes Eschbach-Dymanus SAP, Germany

Hideki Tanaka NICT, Japan

raj.dabre@nict.go.jp

haiyue.song@nict.go.jp

miriam.exel@sap.com

bianka.buschbeck@sap.com

johannes.eschbach-dymanus@sap.com

hideki.tanaka@nict.go.jp

Abstract

Recent works have shown that prompting large language models (LLMs) is effective for translation with markup where LLMs can simultaneously transfer markup tags while ensuring that the content, both inside and outside tag pairs is correctly translated. However, these works assume the existence of high-quality parallel sentences with markup for prompting, which may not always be available. Furthermore, the impact of instruction fine-tuning (IFT) in this setting is unknown. In this paper, we provide a study, the first of its kind, focusing on the effectiveness of synthetically created markup data and IFT for translation with markup using LLMs. We focus on translation from English to five European languages, German, French, Dutch, Finnish and Russian, where we show that regardless of few-shot prompting or IFT, synthetic data created via word alignments, while leading to inferior markup transfer compared to using original data with markups, does not negatively impact the translation quality. Furthermore, IFT mainly impacts the translation quality compared to few-shot prompting and has slightly better markup transfer capabilities than the latter. We hope our work will help practitioners make effective decisions on modeling choices for LLM based translation with markup.

1 Introduction

While a significant majority of machine translation (MT) research has been conducted on translating plain sentences from one language to another, much of the web and proprietary or business documents requiring translation come in structured formats like HTML pages or Microsoft Office files containing markup. Therefore, practical MT systems should be adept not only at translating plain sentences but also sentences with markup (see Figure 1 for an example), where the task is to translate content in the source language while simultaneously ensuring that markup tags wrap the appropriate content in the target language. Until the advent of deep learning, the most commonly used approach for handling markup was the detag-and-project approach (Hanneman and

Dinu, 2020a), which is not end-to-end and is prone to error compounding from individual components such as the MT system, word-aligner and projection algorithms. Therefore, using end-to-end neural networks for translation with markup (Cho et al., 2014) makes a more attractive solution.

Recently, researchers have shown that transformer (Vaswani et al., 2017) based large language models (LLMs) (Brown et al., 2020) can seamlessly translate sentences with markup despite not explicitly being trained to do so (Buschbeck et al., 2022; Dabre et al., 2023). They show that few-shot prompting (Brown et al., 2020) enables LLMs to transfer markup tags when translating from source to target languages. Surprisingly, despite being general purpose, their markup transfer capabilities approach, if not surpass, highly optimized models

English	Click <code><uicontrol></code> Prepayment <code></uicontrol></code> .
German	Klicken Sie <code><uicontrol></code> Vorauszahlung <code></uicontrol></code> .
French	Cliquez <code><uicontrol></code> Prépaiement <code></uicontrol></code>
Japanese	<code><uicontrol></code> 前払 <code></uicontrol></code> をクリックします。

Figure 1: Examples with inline markup, inspired by (Buschbeck et al., 2022).

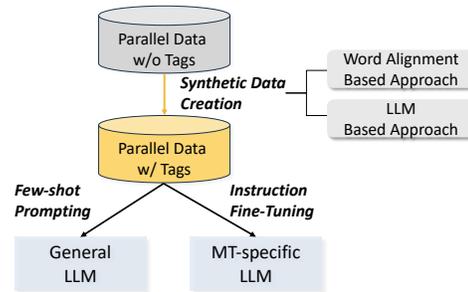


Figure 2: Our framework.

trained specifically for this purpose. However, they assume the existence of high-quality parallel corpora with markup when prompting, and this kind of data may not always be available. Furthermore, while they utilize pre-existing generic instruction fine-tuned (IFT) models, they do not IFT their own MT models, the effectiveness of which remains unknown. In this paper, we fill this gap via a two-pronged exploration on the effectiveness of synthetic data and IFT for translation with markup.

We take the case of translation from English to five European languages, German, French, Dutch, Finnish and Russian, and first establish the efficacy of zero- and few-shot prompting on a popular open-source LLM, namely BLOOM (Le Scao et al., 2022). Following this, we explore approaches for synthetically creating parallel data with markup to understand its efficacy for prompting. We further deepen our investigation by performing IFT of BLOOM with both clean and synthetic data and attempt to discern settings in which synthetic data can be useful. We show that regardless of few-shot prompting or IFT, synthetic data created via word alignments leads to slightly inferior markup transfer compared to high-quality human-curated data; however, it does not negatively impact the translation quality. Furthermore, somewhat surprisingly, we find that IFT itself mainly improves the translation quality compared to few-shot prompting and has only slightly better markup transfer capabilities than the latter. We hope our findings will act as guidelines for practitioners to make effective decisions on modeling choices for translation with markup.

2 Related Work

Our work focuses on machine translation with markup, LLMs and synthetic data.

2.1 MT Model Based Approaches

Detag-and-project is a prevalent technique for translating sentences with markup comprising two steps: 1) stripping tags from the source sentence and translating the plain text, and 2) reinserting tags into the translations. Joanis et al. (2013) utilize a Statistical Machine Translation (SMT) model to translate sentences with markup using a set of tag reinsertion rules in the *project* phase. Similarly, researchers compared various strategies for handling markup using SMT techniques and found that involving complex rules achieves the highest tag projection accuracy (Müller, 2017). More recent works use NMT as the translation model and apply a translation management system to handle the document structure (Hanneman and Dinu, 2020b).

End-to-end approach becomes possible with NMT models. They are often enhanced with data augmentation strategies to optimize the large number of parameters. Synthetic data can be created by inserting tags into corresponding fragments in the source and target plain text parallel sentences (Hanneman and Dinu, 2020b). However, aligned phrases are identified through an exhaustive search, which is computationally expensive. To address this, researchers use efficient word alignments for tag augmentation during the *project* phase (Ryu et al., 2022).

2.2 LLM Based Approaches

LLMs such as GPT-3 (Brown et al., 2020), BLOOM (Le Scao et al., 2022), BLOOMZ (Muenighoff et al., 2022), XGLM (Lin et al., 2022)

and Llama-2 (Touvron et al., 2023) with few-shot in-context-learning (Brown et al., 2020) are well known for their ability to tackle diverse tasks owing to having seen vast amounts of data. Due to their flexibility, LLMs can be directly applied to the structured document translation task without further fine-tuning (Dabre et al., 2023). They apply retrieval-augmented (Lewis et al., 2020) few-shot prompting, which assumes the training set contains numerous parallel sentences with markup in hand. However, for most translation directions, there is usually no dataset with markups available. To this end, we propose to generate synthetic data. Furthermore, rather than prompting, we apply IFT, which our experiments show can achieve higher performance.

2.3 Datasets

Datasets are crucial in advancing structured text (usually with markup) translation. Hashimoto et al. (2019a) create a high-quality multilingual dataset comprising structured web pages designed for the documentation domain translation. Likewise, Buschbeck et al. (2022) develop a multilingual and multi-way evaluation dataset for structured document translation, focusing on Asian languages but only providing evaluation sets.

3 Methodology

Figure 2 presents an overview of the methodology followed in this paper: few-shot prompting in Section 3.1, instruction fine-tuning (IFT) in Section 3.2, and our methods for creating synthetic parallel data with markup in Section 3.3.

3.1 Few-shot Prompting

For our experiments, we use the N -shot approach, selecting N translation pairs (S_i, T_i) from an example pool to prompt the LLM. Like Dabre et al. (2023), unless (plain) data without markup is used, we use structure-aware prompting, where we use examples containing markup tags for test sentences with tags, and examples without markup tags for test sentences without tags. The specific template is in

¹If we have the source phrase $s_i \dots s_N$, word alignments $A = \{i : j\}$, then the aligned target words are $L = \cup_{x=i}^N A(x)$ and the aligned target phrase is $t_{\min(L)} \dots t_{\max(L)}$.

²Although it might seem unnatural to consider all tags appearing with the same probability, in practice there is no way to know the tag distribution in a realistic setting so we make no assumptions and rely on uniform sampling of tags.

³Unnatural means that those phrases are unlikely to be surrounded by tags in the real structured data, resulting in the mismatch of the distribution of training data and test data.

Appendix A.1.

3.2 Instruction Fine-tuning (IFT)

IFT is simply fine-tuning a pre-trained LLM with parallel data to enable it to translate from a source language to a target language without needing to provide demonstrations (or shots). The specific template is in Appendix A.3. As is common practice (Wei et al., 2022), we only consider the loss computed on the completion part of the sequence.

3.3 Synthetic Data Creation

We consider two approaches for synthetic data creation: using word alignment and LLMs.

Word Alignment Based Approach

The overview of the word alignment based approach is shown in Figure 3 and we call the resultant data Alignment-Synthetic-Tagged (AST). This approach involves the following steps:

1. Obtain word alignments for a parallel corpus without markup.
2. Randomly sample a phrase of a *maximum size* from the source sentence.
3. Use the word alignments with the min-max algorithm¹ (Zenkel et al., 2021) to identify the aligned phrase in the target sentence.
4. Uniformly² sample a tag from a pre-defined set.
5. Wrap both the source and target sentence phrases with the sampled markup tag.

Our approach is mainly motivated by the detag-and-project methods (Hanneman and Dinu, 2020b) and the idea of grouping words into phrases in phrase-based SMT (Och, 1999), and the results of data augmentation (Ryu et al., 2022). However, ours is more efficient than the one by Hanneman and Dinu (2020a), which relied on a more computationally expensive approach by exhaustively covering multiple phrase spans and translations via MT to identify high-quality aligned phrases.

LLM Based Approach

Random sampling in the word-alignment-based approach often results in *unnatural* phrases.³ To this end, we propose utilizing LLMs to select *natural* phrases for inserting markup, as shown in Figure 4.

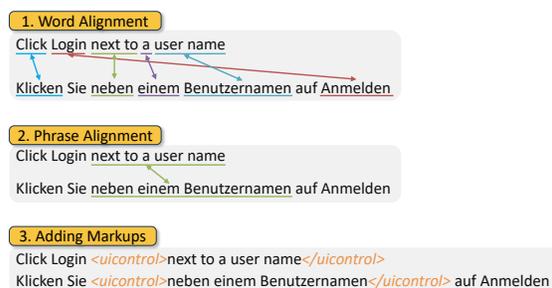


Figure 3: The overview of the word-alignment-based synthetic data creation method. It generates word alignments in the first stage, samples a phrase in the second stage, and inserts a randomly sampled tag pair in the final stage.

We called the resultant data as LLM-Synthetic-Tagged (LST). We prompt the LLMs (BLOOM 7B in our experiments) with few-shot examples, and the model takes source and target sentences without markup as input and outputs source and target sentences with markup. The hand-crafted and fixed 5-shot examples (prompt in Appendix A.2) show how a sentence pair without markup can be transformed into a pair with markup.

4 Experimental Settings

This section describes datasets, implementation details, and various settings for analysis.

4.1 Datasets and Languages

We consider the Salesforce Localization Dataset (Hashimoto et al., 2019b) which spans English and seven languages, out of which we choose five European target languages, namely, German, French, Dutch, Finnish and Russian. The data for each language pair consists of approximately high-quality 100k training, 2k development and 2k testing high-quality sentence pairs of which 26% of the pairs *naturally* contain markup. We use the development set of 2,000 sentence pairs as the test set because the test set is hidden. Furthermore, since LLM IFT is computationally expensive, and our objective is to study the efficacy of synthetic data and IFT, we

⁴The aligned target spans may be longer or shorter, but this is not something that can be controlled.



Figure 4: The overview of the LLM-based synthetic data creation method. We prompt the LLM with few-shot examples, and the model directly generates parallel sentences with tag pairs.

choose a subset of the training data for our experiments. Specifically, we choose the first 2,000 sentence pairs for development (instead of the official development set), and the next 20,000 sentence pairs for few-shot prompting or IFT. We create a version of the 20,000 pairs by removing all markup information and call this *Plain* data, whereas the corresponding version with 26% of the sentences naturally containing (high-quality/gold) markup is called *Clean* data.

Synthetic Data Settings

When creating synthetic sentence pairs with markup using word alignment (Alignment-Synthetic-Tagged or *AST*), we experiment with maximum source (English) spans of size 4 and 6 tokens, where we randomly choose one source phrase whose token length is less or equal to this number.⁴ For synthetic data created with LLMs (LLM-Synthetic-Tagged or *LST*), we cannot control maximum spans and leave it to the model to wrap phrases with markup tags as it sees fit. We prompt the model with 5 manually constructed shots, which are fixed for each language pair. For the decoding algorithm, we applied greedy search with a temperature of 0. As for the percentage of sentence pairs with markup tags in the training data, we experiment with 1%, 2%, 5%, 15% and 26% of examples with synthetic markup where 26% is analogous to the amount of *naturally*

occurring markup in *Clean* data. For LST, due to reasons explained in Section 5.3, we were only able to generate a maximum of 24% and 14% tagged data with synthetic markup only for English-German and English-Russian, respectively. For these pairs, henceforth 26% actually implies 24% and 14% respectively. Unless explicitly mentioned, we use the data containing 26% pairs with markup for AST and LST when experimenting with prompting and IFT.

4.2 Implementation, Training, and Evaluation

We implement the code for creating synthetic data and prompting in Python. For word alignment, we used FastAlign⁵ (Dyer et al., 2013) with default settings for forward, reverse, and we symmetrize alignments with grow-diag-final-and. We use open-instruct⁶ (Wang et al., 2023) for IFT. We use the 7.1 billion parameter variant⁷ of the BLOOM model (Le Scao et al., 2022). We choose this model over more recent ones like Llama (Touvron et al., 2023) since the latter is not explicitly suited for non-English generation.⁸ Due to our low-resource setting, we use LoRA (Hu et al., 2021) for fine-tuning, with a rank of 4 and an alpha of 8, and a LoRA dropout of 0.05. We use a total batch size of 32 with gradient accumulation. We train for a maximum of 4 epochs, evaluate every epoch, and choose the checkpoint corresponding to the lowest loss.⁹ Our experiments are performed on 40GB A100 GPUs. For decoding the test sets, we perform greedy decoding.

For evaluation, while Hashimoto et al. (2019b) propose XML-BLEU, we consider XML-chrF as a measure of overall translation quality, including both, *content as well as markup transfer quality*. They use multi-bleu,¹⁰ however, since Post (2018) have shown that using multi-bleu is not reliable, we switch to sacrebleu¹¹ and following recent trends, chrF scores (Popović, 2015) to report XML-chrF. Additionally, just as Hashimoto et al. (2019b) do, we report XML-Structure-Match, henceforth XML-Match, as a measure of *purely* the markup transfer

capabilities, with details explained in Appendix B.

4.3 Prompting Settings

For few-shot prompting on the test set, we use 0-, 1- and 4-shot prompting when the base BLOOM model is used. After performing IFT, we only use 0-shot prompting. The 1- and 4-shot prompting examples are chosen randomly. We perform three runs and report the mean scores.

5 Results

We structure the results in two major sections: the first focusing on synthetic data and using it for prompting, and the second focusing on IFT along with synthetic data.

5.1 Synthetic Data for Prompting

Table 1 gives the results for 0-, 1- and 4-shot prompting with plain, clean and synthetic data. Perhaps the most surprising result is that 0-shot prompting has very high XML-Match indicating that the markup structure is almost always correctly transferred from source to target language. However, the XML-chrF scores are rather low, except for English to French, indicating that while the LLM can transfer markup, it cannot translate content well. Increasing the number of shots has a marked improvement on the XML-chrF scores. On the other hand, the XML-Match scores do not vary much regardless of the data used for prompting.

Although Dabre et al. (2023) used different metrics for evaluation, their *tag* metric is analogous to XML-Match and they always reported very low scores for the same. Note that they focused on Japanese, Chinese and Korean, which are **a**. Not well-supported in BLOOM and **b**, are linguistically distant from English. On the other hand, we focus on European languages which are better supported in BLOOM and are linguistically closer to English. This results in the following finding:

⁵https://github.com/clab/fast_align

⁶<https://github.com/allenai/open-instruct/>

⁷<https://huggingface.co/bigscience/bloom-7b1>

⁸While these models are known to be able to generate in non-English models, our main goal is not obtaining SOTA results but to study how LLMs behave in the context of synthetic data in conjunction with IFT. Therefore, we rely on BLOOM in our experiments.

⁹This is different from typical MT experiments where early stopping is done on the downstream metric itself. Since this is expensive for LLMs, we rely on loss, which can be computed non-autoregressively.

¹⁰<https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-bleu.perl>

¹¹<https://github.com/mjpost/sacrebleu>

Prompting Data Type	XML-chrF						XML-Match					
	en→de	en→fi	en→fr	en→nl	en→ru	Avg.	en→de	en→fi	en→fr	en→nl	en→ru	Avg.
<i>0-shot</i>												
Baseline	33.6	18.3	54.3	33.3	24.1	32.7	96.5	95.6	95.3	95.3	94.4	95.4
<i>1-shot</i>												
Plain	39.0	17.2	58.6	38.6	30.9	36.9	96.2	89.9	96.4	95.8	94.1	94.5
Clean	39.5	17.6	59.0	38.6	30.9	37.1	96.3	90.0	96.0	95.6	94.0	94.4
AST-4	39.3	17.7	58.6	38.8	30.8	37.0	96.1	90.5	96.2	95.6	93.3	94.3
AST-6	39.4	17.6	58.2	38.8	30.6	36.9	96.2	90.3	95.9	95.6	92.9	94.2
LST	39.5	17.2	58.4	38.7	30.9	36.9	95.8	88.4	95.7	95.5	92.6	93.6
<i>4-shot</i>												
Plain	41.5	19.3	61.3	40.7	32.9	39.1	96.9	90.9	96.8	96.1	94.1	95.0
Clean	41.1	18.9	61.5	40.7	32.9	39.0	96.1	89.7	96.6	93.4	94.3	94.1
AST-4	41.0	18.9	61.2	40.9	32.1	38.8	95.9	89.6	96.5	96.1	92.3	94.1
AST-6	40.9	19.0	61.4	40.7	32.3	38.9	96.0	90.0	96.7	95.8	92.6	94.2
LST	41.3	18.0	60.7	40.3	32.3	38.5	96.0	86.6	96.3	95.4	92.6	93.4

Table 1: XML-chrF and XML-Match of different types of data for few-shot prompting. Best results in each direction are **bolded**. **Plain** refers to data w/o markup, and **Clean** means markup data created by humans from the dataset.

Prompting Data Type	XML-chrF						XML-Match					
	en→de	en→fi	en→fr	en→nl	en→ru	Avg.	en→de	en→fi	en→fr	en→nl	en→ru	Avg.
<i>Reference: Prompting Results</i>												
Plain 4-shot	41.5	19.3	61.3	40.7	32.9	39.1	96.9	90.9	96.8	96.1	94.1	95.0
<i>Instruction Fine-Tuning Results</i>												
Plain	57.5	46.5	72.1	59.4	45.4	56.2	94.0	95.2	96.7	95.0	88.5	93.9
Clean	60.1	47.9	75.3	60.7	50.0	58.8	97.2	96.8	98.7	96.4	95.4	96.9
AST-4	58.5	46.3	72.7	59.3	47.5	56.9	96.2	95.0	96.6	95.4	93.5	95.3
LST	55.9	44.9	73.6	59.2	46.4	56.0	91.8	92.1	97.0	94.8	89.6	93.1

Table 2: XML-chrF and XML-Match of different types of data for instruction fine-tuning. Best results are **bolded**. **Plain** refers to data w/o markup, and **Clean** means markup data created by humans from the dataset.

Finding 1: *Base LLMs are fairly good at markup transfer of well-supported languages, and demonstrations (or shots) mainly affect the content translation quality.*

5.1.1 Does Synthetic Data Even Matter for Prompting?

Comparing the 1- and 4-shot results in Table 1, it is clear that there is no notable difference in performance between using examples with (Clean, AST, LST) and without (Plain) markup for translating sentences with markup. Among synthetic data (AST-4, AST-6 and LST), the approach for synthetic data does not matter. This leads to the following finding:

Finding 2: *The LLM likely sees markup tags as tokens to be transferred from source to target and does not distinguish them from regular words/tokens, and it uses shots only to know how to translate.*

5.2 Synthetic Data for Instruction Fine-Tuning

Having shown the impact of various types of data with and without markup for prompting, we now show results for using the aforementioned data for instruction fine-tuning. Table 2 shows the fine-tuning results, and in-context-learning results as a reference, where we prompt the LLM with 5-shot translation samples in the target domain without markup.

Different from few-shot prompting, the impact of different types of data is visible. While fine-tuning using data without markup (Plain) significantly improves XML-chrF, the markup transfer itself (XML-match) is negatively affected. In fact, few-shot prompting does better. Since we want our model to translate as well as transfer markup, fine-tuning on data without markup is not viable. On the other hand, fine-tuning with human-created data (Clean) not only has better markup transfer but also

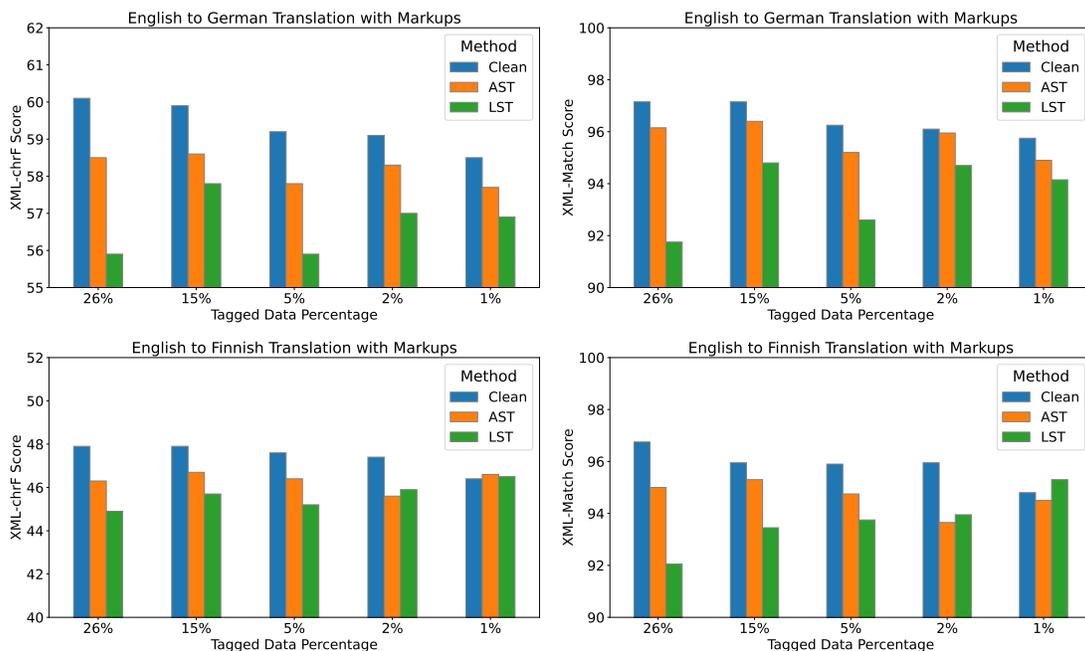


Figure 5: Results of IFT using varying percentages of data with markup. We show XML-chrF (left figure) scores and XML-Match scores (right figure) of randomly choosing X percentage of pairs with markup from the original data (Clean), and data generated by our AST (with max span of 4) and LST approaches. X ranges from natural max, that is the markup data percentage in Clean of 26%, to 15%, 5%, 2% and 1%.

has better content translation quality, mostly indicated by the significant increase in chrF by about 5 points. However, it's not always possible to have human created data with markup, but in this case synthetic data appears to be useful. Comparing AST-4,¹² and Clean, we see that while the former is expectedly slightly inferior to the latter, the gap is rather small. Although we expected LST to be better than AST, its performance was disappointing. Our analysis in the following subsection will shed some light on this. Our finding is:

Finding 3: *IFT requires high-quality data with markup for the best performance, however synthetically generated data is certainly a viable option.*

5.2.1 Does Synthetic Data Quantity Matter?

Previously, we did not focus on the ratio of data without and with markup and created as much synthetic data as was present in the human created ver-

sion. However, it is not clear what the optimal ratio is. To this end, we explore varying markup data ratios in the training set. Figure 5 shows the result for English to German and Finnish. Here, we have 3 important observations: a. Clean data is almost always better than synthetic data, but the gap keeps diminishing as the amount of markup data drops. b. Even having 1% data with markup is still better than having no data with markup. c. LST is inferior to AST in most settings. We put the full table of five language pairs in Appendix C. The finding is:

Finding 4: *High-quality markup data is always useful at any scale even if it forms 1% of the overall IFT data, however synthetic data generated using alignment is a viable alternative at all scales.*

5.3 Evaluation of Synthetic Data

We briefly evaluate synthetic data to understand its quality. Consider the following Clean, AST and

¹²In our preliminary experiments for IFT, we did not notice any difference between AST-4 and AST-6, so we only report results for AST-4.

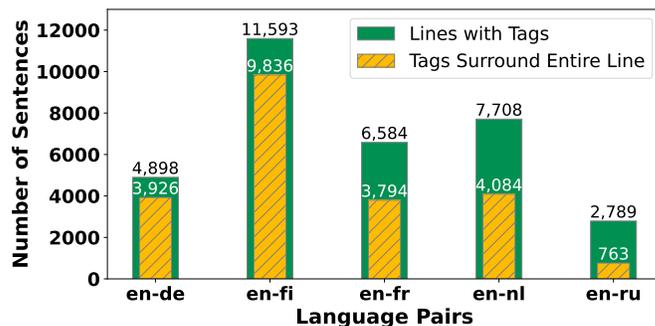


Figure 6: Maximum number of sentence pairs with tags, out of 20,000, that could be generated using LST. Of this, we select pairs corresponding to a maximum of 26% of the training/prompting data. We also show the number of pairs with tags surrounding entire sentences (tags only at the beginning and the end).

LST variations of the same English-German pair:

Clean (En): From Setup, enter `<userinput>Salesforce Classic Configurations</userinput>` in the `<parname>Quick Find</parname>` box, then select `<uicontrol>Salesforce Classic Configurations</uicontrol>`.

Clean (De): Geben Sie unter “Setup” im Feld `<userinput>Schnellsuche</userinput>` den Text `<parname>Konfigurationen für Salesforce Classic</parname>` ein und wählen Sie dann `<uicontrol>Konfigurationen für Salesforce Classic</uicontrol>` aus.

AST (En): From Setup, enter `<uinolabel>Salesforce Classic Configurations in the Quick Find</uinolabel>` box, then select Salesforce Classic Configurations.

AST (De): Geben Sie unter “Setup” im Feld `<uinolabel>Schnellsuche den Text Konfigurationen für Salesforce Classic ein</uinolabel>` und wählen Sie dann Konfigurationen für Salesforce Classic aus.

LST (En): From Setup, enter `<uicontrol>Salesforce Classic Configurations</uicontrol>` in the Quick Find box, then select `<uicontrol>Salesforce Classic Configurations</uicontrol>`.

LST (De): Geben Sie unter “Setup” im Feld Schnellsuche den Text `<uicontrol>Konfigurationen für Salesforce Classic</uicontrol>` ein und wählen Sie dann `<uicontrol>Konfigurationen für Salesforce Classic</uicontrol>` aus.

It is clear that LST data is more similar to Clean data in which shorter phrases corresponding to keywords are wrapped with tags, whereas AST covers a

longer phrase. Although not evident in this example, AST can tag unnatural phrases and given discrepancies compared to Clean data, it makes sense that models trained with AST data will always underperform those trained with Clean data. However, the confounding factor is why models trained on LST data are worse than on AST, despite LST data looking similar to Clean data. We found that LST tends to wrap entire sentences with tags more often than AST, with examples in Appendix D.

As shown in Figure 6, many of the LST examples are with tag pairs surrounding the entire sentence. For English to German, of 4,898 LST tagged examples, 3,926 are entire sentences. Whereas in the AST data, out of 5,235 tagged examples, only 460 are entire sentences. For reference, in Clean, out of 5,235 tagged sentences, only 60 are entire sentences. This large proportion of entire tagged sentences appears to have a larger impact than having non-keyword or unnatural phrases. For the sentences with tags but not surrounding the entire sentence, the average number of words surrounded by one tag pair is approximately 2 for all languages which is reasonable. Furthermore, despite our best efforts, we could not compel BLOOM to generate the desired number of tagged sentences. Finally, there is a significant variation in the number of sentences with tags across different language pairs, which contributes to the variation in MT performance, implying the need for future study.

6 Conclusion

In this paper, we have studied the effectiveness of synthetic data and instruction fine-tuning for translation with markup. We observed that an LLM without few-shot prompting or IFT already has impressive markup transfer capabilities, but suffers from low translation ability in the document domain. Although few-shot prompting can help improve translation quality, IFT is more effective, while also improving markup transfer capabilities regardless of whether high-quality or synthetic data was used. In the future, we would like to explore more controllable and scalable ways to generate synthetic data and eliminate the need for human curated data.

References

- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language models are few-shot learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.
- Buschbeck, B., Dabre, R., Exel, M., Huck, M., Huy, P., Rubino, R., and Tanaka, H. (2022). A multilingual multiway evaluation data set for structured document translation of Asian languages. In He, Y., Ji, H., Li, S., Liu, Y., and Chang, C.-H., editors, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2022*, pages 237–245, Online only. Association for Computational Linguistics.
- Cho, K., van Merriënboer, B., Bahdanau, D., and Bengio, Y. (2014). On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar. Association for Computational Linguistics.
- Dabre, R., Buschbeck, B., Exel, M., and Tanaka, H. (2023). A study on the effectiveness of large language models for translation with markup. In Utiyama, M. and Wang, R., editors, *Proceedings of Machine Translation Summit XIX, Vol. 1: Research Track*, pages 148–159, Macau SAR, China. Asia-Pacific Association for Machine Translation.
- Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A simple, fast, and effective reparameterization of IBM model 2. In Vanderwende, L., Daumé III, H., and Kirchhoff, K., editors, *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.
- Hanneman, G. and Dinu, G. (2020a). How should markup tags be translated? In *Proceedings of the Fifth Conference on Machine Translation*, pages 1160–1173.
- Hanneman, G. and Dinu, G. (2020b). How should markup tags be translated? In Barrault, L., Bojar, O., Bougares, F., Chatterjee, R., Costa-jussà, M. R., Federmann, C., Fishel, M., Fraser, A., Graham, Y., Guzman, P., Haddow, B., Huck, M., Yepes, A. J., Koehn, P., Martins, A., Morishita, M., Monz, C., Nagata, M., Nakazawa, T., and Negri, M., editors, *Proceedings of the Fifth Conference on Machine Translation*, pages 1160–1173, Online. Association for Computational Linguistics.
- Hashimoto, K., Buschiazzo, R., Bradbury, J., Marshall, T., Socher, R., and Xiong, C. (2019a). A high-quality multilingual dataset for structured documentation translation. In Bojar, O., Chatterjee, R., Federmann, C., Fishel, M., Graham, Y., Haddow, B., Huck, M., Yepes, A. J., Koehn, P., Martins, A., Monz, C., Negri, M., Névéal, A., Neves, M., Post, M., Turchi, M., and Verpoor, K., editors, *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 116–127, Florence, Italy. Association for Computational Linguistics.
- Hashimoto, K., Buschiazzo, R., Bradbury, J., Marshall, T., Socher, R., and Xiong, C. (2019b). A high-quality multilingual dataset for structured documentation translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 116–127.
- Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. (2021). Lora: Low-rank adaptation of large language models.

- Joanis, E., Stewart, D., Larkin, S., and Kuhn, R. (2013). Transferring markup tags in statistical machine translation: a two-stream approach. In O’Brien, S., Simard, M., and Specia, L., editors, *Proceedings of the 2nd Workshop on Post-editing Technology and Practice*, Nice, France.
- Le Scao, T., Fan, A., Akiki, C., Pavlick, E., Ilić, S., Hessel, D., Castagné, R., Luccioni, A. S., Yvon, F., Gallé, M., et al. (2022). BLOOM: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.
- Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., et al. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.
- Lin, X. V., Mihaylov, T., Artetxe, M., Wang, T., Chen, S., Simig, D., Ott, M., Goyal, N., Bhosale, S., Du, J., Pasunuru, R., Shleifer, S., Koura, P. S., Chaudhary, V., O’Horo, B., Wang, J., Zettlemoyer, L., Kozareva, Z., Diab, M., Stoyanov, V., and Li, X. (2022). Few-shot learning with multilingual generative language models. In Goldberg, Y., Kozareva, Z., and Zhang, Y., editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9019–9052, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Muennighoff, N., Wang, T., Sutawika, L., Roberts, A., Biderman, S., Scao, T. L., Bari, M. S., Shen, S., Yong, Z.-X., Schoelkopf, H., et al. (2022). Crosslingual generalization through multitask finetuning. *arXiv preprint arXiv:2211.01786*.
- Müller, M. (2017). Treatment of markup in statistical machine translation. In Webber, B., Popescu-Belis, A., and Tiedemann, J., editors, *Proceedings of the Third Workshop on Discourse in Machine Translation*, pages 36–46, Copenhagen, Denmark. Association for Computational Linguistics.
- Och, F. J. (1999). An efficient method for determining bilingual word classes. In Thompson, H. S. and Lascarides, A., editors, *Ninth Conference of the European Chapter of the Association for Computational Linguistics*, pages 71–76, Bergen, Norway. Association for Computational Linguistics.
- Popović, M. (2015). chrF: character n-gram F-score for automatic MT evaluation. In Bojar, O., Chatterjee, R., Federmann, C., Haddow, B., Hokamp, C., Huck, M., Logacheva, V., and Pecina, P., editors, *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.
- Post, M. (2018). A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191.
- Rei, R., Stewart, C., Farinha, A. C., and Lavie, A. (2020). COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702.
- Ryu, Y., Choi, Y., and Kim, S. (2022). Data augmentation for inline tag-aware neural machine translation. In Koehn, P., Barrault, L., Bojar, O., Bougares, F., Chatterjee, R., Costa-jussà, M. R., Federmann, C., Fishel, M., Fraser, A., Freitag, M., Graham, Y., Grundkiewicz, R., Guzman, P., Haddow, B., Huck, M., Jimeno Yepes, A., Kocmi, T., Martins, A., Morishita, M., Monz, C., Nagata, M., Nakazawa, T., Negri, M., Névéal, A., Neves, M., Popel, M., Turchi, M., and Zampieri, M., editors, *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 886–894, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. (2023). Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.
- Wang, Y., Ivison, H., Dasigi, P., Hessel, J., Khot, T., Chandu, K. R., Wadden, D., MacMillan, K., Smith, N. A., Beltagy, I., and Hajishirzi, H. (2023). How far can camels go? exploring the state of instruction tuning on open resources.
- Wei, J., Bosma, M., Zhao, V., Guu, K., Yu, A. W., Lester, B., Du, N., Dai, A. M., and Le, Q. V. (2022). Fine-tuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Zenkel, T., Wuebker, J., and DeNero, J. (2021). Automatic bilingual markup transfer. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3524–3533.

Limitation

One limitation of this work is that we only used BLOOM-7B1, thus the performance of different LLM families such as Llama-3 or Gemma, or LLMs with different amounts of parameters such as BLOOM-560M or Llama-3 70B, is not verified. It is possible that larger language models can have higher markup transfer capability, have higher translation capability on data in structured document domain, generate better synthetic data, and are more controllable.

A Prompting Details

A.1 Few-shot prompting for MT

The prompting template is as follows:

Translate the following sentence from E to F . The translation should be in F and no other language.

E : [S_1]

F : [T_1]

...

E : [S_N]

F : [T_N]

E : [S_t]

F :

In the template above, E is the source language, F is the target language, and S_t is the test example for which we want to obtain a translation. Note that in the template, each source and target language sentence is wrapped in opening and closing square brackets ([,]). After the model produces outputs, we remove the prompted prefix and retain the first segment produced by the model within the [and] brackets as the model's translation.

A.2 Few-shot prompting for synthetic data creation

This section formats the prompt, and the real prompt with five demonstrations is shown in Table 4.

Insert tag pairs to parallel sentences in E and F .

Here is a list of possible tags: <ph> <uicontrol> <parmname> <codeph> <xref> <userinput> <varname> <filepath> <i> <systemoutput> <term> <title> <p> <note> <cite> <indexterm> <fn> <u>.

Input:

E : S_i

F : T_i

Output:

E : S'_i

F : T'_i

...

Input:

E : S_t

F : T_t

Output:

In the template above, E is the source and F is the target language. (S_i, T_i) is parallel sentences without markup and (S'_i, T'_i) is parallel sentences with markup. (S_t, T_t) is the test example which contains parallel sentences without markup. After generating the parallel sentences, we post-process the output to extract S'_i and T'_i , and verify whether there are tags in both of them and whether the detagged version of them equal to S_t and T_t . We only use the outputs that passed these verification processes.

A.3 IFT prompt template

The data format fed to the LLM for IFT is as follows:

Translate the following sentence from E to F . The translation should be in F and no other language.

E : [S]

F : [T]

Here, [T] consists of the completion and everything before it is the prompt.

B Details of Evaluation Metrics.

We explain the calculation details of XML-Match and XML-chrF metrics. We first use etree to extract the XML structure of the output and reference. The XML-Match is the percentage of outputs that have exactly the same XML structures as their references. If the XML structures of an output and its reference match, then the translation and reference are split by the XML tags and we evaluate the chrF score by comparing each split segment. If the structures do not match, the chrF score is counted as zero to penalize the irrelevant outputs. We leave COMET score reporting (Rei et al., 2020) for the future.

C Full Results of IFT using Varying Percentages of Tagged Data

We present the results of instruction fine-tuning using varying percentages of data with markup for all five language pairs in Figure 7. As per the explanation in Section 5.3 and Appendix D, we were unable to control the amount of sentence pairs with synthetic markup for LST. Corresponding to Clean which naturally has 26% data with markup and AST where we can generate the exactly 26% of pairs with markup, LST was unable to generate more than 24% pairs with markup for English-German. For English-Russian, a maximum of 14% pairs with markup could be generated. Since we have no control over this, for English-German, the scores corresponding to 26% synthetic markup pairs using LST are actually scores for 24% synthetic markup pairs using LST. For English-Russian, the scores corresponding to 26% as well as 15% synthetic markup pairs using LST are actually scores for 14% synthetic markup pairs using LST.

Comparing different methods, we found that clean data is almost always better than synthetic data, and LST is inferior to AST in all directions except English→French, where LST showed higher performance even than clean data using 26% of tagged data. This may be because, for French, there is a large number of *normal* tagged sentences (not tags surrounding the entire sentence). For English→Dutch, which also has a large number of normal tagged sentences, the XML- chrF scores are

better than AST using 15% and 2% of tagged data. However, for language pairs where LST generates low-quality tagged data, such as English→German, the final performance is also low. Compared to LST, AST is more stable where the gap between Clean data is small (or comparable) for all language directions. Furthermore, we observed that AST performed better than Clean using 1% tagged in English→German and English→Dutch directions, and the gap with using 26% tagged data is small. This shows that we can achieve high-quality transfer learning by AST with a tiny amount of noisy data.

D LLM is not Always Controllable.

LLM-based (to be specific, *BLOOM7B1-based*) synthetic data creation is not stable because it does not always generate output with tags even if we always prompt the model to do so. In fact, English-German and English-Russian were especially hard. For English-German we were unable to generate more than approximately 24% and for English-Russian more than approximately 14% sentences with markup. What’s worse, it simply added tags at the beginning and the end of one sentence in many cases. We show examples in Table 3 and statistics in Figure 6, from which we can observe that for English-Finnish, English-French, and English-Dutch, a large percentage of the tagged data are not helpful with tag pairs only at the beginning and the end. In the future, we will explore larger LLMs such as Llama-3-70B-Instruct, which may generate more natural tagged sentence pairs.

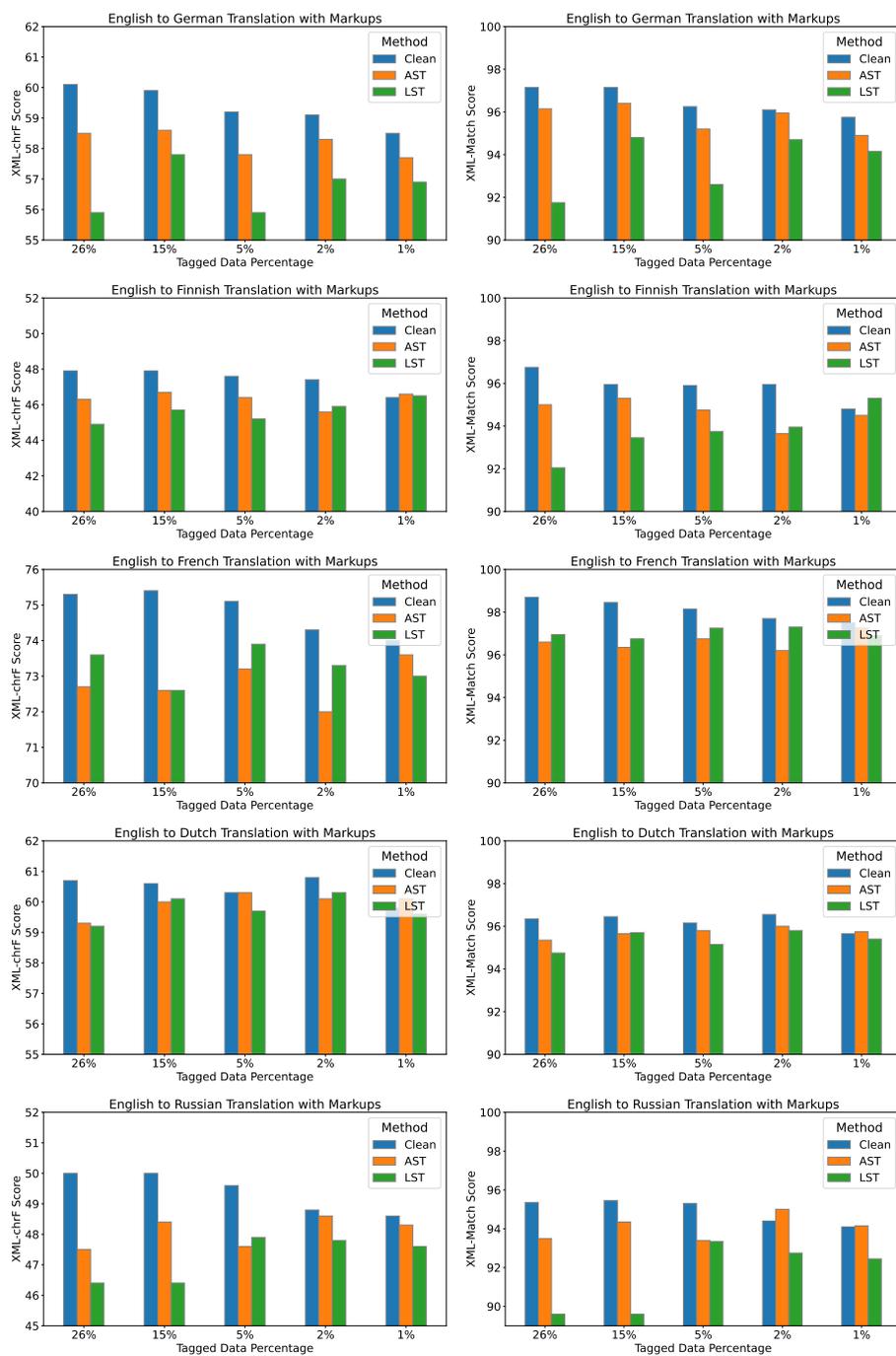


Figure 7: Results of IFT using varying percentages of data with markup for all five language pairs. We show XML-chrF (left column) scores and XML-Match scores (right column) of randomly sampling X percentage of tagged data from the original data (Clean), and data generated by our AST (with max span of 4) and LST approaches. X ranges from natural max, that is the tagged data percentage in Clean of 26%, to 15%, 5%, 2% and 1%.

<i>English:</i>	<xref>Size of the work items in the queue based on its routing configuration.</xref>
<i>German:</i>	<xref>Die Größe der Arbeitselemente in der Warteschlange basierend auf ihrer Weiterleitungskonfiguration.</xref>
<i>English:</i>	<codeph>Only included if you choose to import campaigns data to Sales Analytics through the configuration wizard.</codeph>
<i>Finnish:</i>	<codeph>Sisällytetään vain, jos päättät tuoda kampanjadataa Sales Analyticsiin ohjatun määrittötoiminnon kautta.</codeph>
<i>English:</i>	<ph>Support for macros is different in Salesforce Classic and Lightning Experience.</ph>
<i>French:</i>	<ph>La prise en charge des macros diffère entre Salesforce Classic et Lightning Experience.</ph>
<i>English:</i>	<parmname>Ability to manage all case comments</parmname>
<i>Dutch:</i>	<parmname>Mogelijkheid tot beheer van alle caseopmerkingen</parmname>
<i>English:</i>	<userinput>Running user does not have permission to access report type.</userinput>
<i>Russian:</i>	<userinput>Текущий пользователь не имеет права доступа к типу отчета.</userinput>

Table 3: Examples of sentence pairs with tags that only appear at the beginning and the end outputted by BLOOM.

<p>Insert tag pairs to parallel sentences in English and German. Here is a list of possible tags: <ph> <uicontrol> <parmname> <codeph> <xref> <userinput> <varname> <filepath> <i> <systemoutput> <term> <title> <p> <note> <cite> <indexterm> <fn> <u></p>	
Input:	English: For more information, see Using the Agent Console List View. German: Weitere Informationen finden Sie unter Verwenden der Listenansicht der Agentenkonsole.
Output:	English: For more information, see <xref><ph>Using the <ph>Agent Console</ph> List View</ph></xref>. German: Weitere Informationen finden Sie unter <xref><ph>Verwenden der Listenansicht der <ph>Agentenkonsole</ph></ph></xref>.
Input:	English: In the Folders pane on the Reports tab, select Opportunity Reports. German: Wählen Sie im Fenster Ordner auf der Registerkarte "Berichte" die Option Opportunity-Berichte.
Output:	English: In the <parmname>Folders</parmname> pane on the Reports tab, select <uicontrol>Opportunity Reports</uicontrol>. German: Wählen Sie im Fenster <parmname>Ordner</parmname> auf der Registerkarte "Berichte" die Option <uicontrol>Opportunity-Berichte</uicontrol>.
Input:	English: Measure – Choose a measure for the metric chart German: Maßeinheit: Wählen Sie eine Maßeinheit für das metrische Diagramm aus
Output:	English: Measure – Choose a measure for the metric chart German: Maßeinheit: Wählen Sie eine Maßeinheit für das metrische Diagramm aus
Input:	English: On your profile page, click Update under your photo (or Add Photo if you haven't added a photo yet), then select Show in communities with publicly accessible pages. German: Klicken Sie auf Ihrer Profilseite unter Ihrem Foto auf Aktualisieren (bzw. auf Foto hinzufügen, wenn Sie noch kein Foto hinzugefügt haben) und wählen Sie dann die Option In Communities mit öffentlich zugänglichen Seiten anzeigen.
Output:	English: On your profile page, click <uicontrol>Update</uicontrol> under your photo (or <uicontrol>Add Photo</uicontrol> if you haven't added a photo yet), then select <parmname>Show in communities with publicly accessible pages</parmname>. German: Klicken Sie auf Ihrer Profilseite unter Ihrem Foto auf <uicontrol>Aktualisieren</uicontrol> (bzw. auf <uicontrol>Foto hinzufügen</uicontrol>, wenn Sie noch kein Foto hinzugefügt haben) und wählen Sie dann die Option <parmname>In Communities mit öffentlich zugänglichen Seiten anzeigen</parmname>.
Input:	English: Salesforce supports the open-standard cross-domain identity management SCIM specification 1.1, and provides a few extensions to the spec so you can edit and manage user properties using the REST API. German: Salesforce unterstützt die Open-Standard-SCIM-Spezifikation 1.1 (System for Cross-domain Identity Management) und stellt einige Erweiterungen für die Spezifikation bereit, sodass Sie Benutzereigenschaften mit der REST-API bearbeiten und verwalten können.
Output:	English: <ph>Salesforce</ph> supports the open-standard cross-domain identity management SCIM specification 1.1, and provides a few extensions to the spec so you can edit and manage user properties using the <ph>REST API</ph>. German: <ph>Salesforce</ph> unterstützt die Open-Standard-SCIM-Spezifikation 1.1 (System for Cross-domain Identity Management) und stellt einige Erweiterungen für die Spezifikation bereit, sodass Sie Benutzereigenschaften mit der <ph>REST-API</ph> bearbeiten und verwalten können.
Input:	English: \${English Sentence} German: \${German Sentence}

Table 4: The full prompt for LLM to generate tagged sentences in English and German.