Using Iterative Back-Translation to Improve Neural Poetry Translation

Stanford CS224N Custom Project

Andrew Chen
Stanford University
ayc227@stanford.edu

Abstract

Despite rapid improvements in neural machine translation, neural poetry translation remains a difficult task in NLP because of the extensive use of figurative language, restrictions of poetic structure, and lack of large datasets. The current literature on neural poetry translation is also limited, though fine-tuning a multilingual model on poetic data has proven to be beneficial for poetry translation based on both automatic and human evaluation. There is one series of moderately-sized parallel poetry datasets, but further progress is limited by the lack of more available data. I seek to address this low-resource issue by implementing iterative back-translation to generate synthetic data, which has been proven to improve translation model performance on automatic metrics. This synthetic data is combined with true parallel data to fine-tune a state-of-the-art multilingual transformer model, mBART50. Taking the Italian to English translation direction as a case study, I find that the chosen architecture and data do not give sufficient evidence to show that iterative back-translation is beneficial to neural poetry translation. Instead, I observe that iterative back-translation decreases the poetic quality of the data, which may hinder model performance. Further research should be performed on larger, multilingual datasets to confirm and extend these results.

1 Key Information to Include

- TA Mentor: Moussa Doumbouya

2 Introduction

The application of neural machine translation to the domain of translating poetry, referred to here as "neural poetry translation", is underrepresented in today's growing NLP literature. Neural poetry translation is an interesting but difficult task because of the extensive use of figurative language and restrictive structures in poetry. These challenges are what makes poetry translation difficult even for humans, where the final translation has to be a work of poetic art as much as original poem is. A lack of parallel datasets exacerbates this problem for NLP systems, especially for state-of-the-art transformers that require large amounts of data for training. But parallel data is difficult to obtain because it requires professional translators who need to be able to interpret and write poetry in multiple languages.

This paper addresses the latter problem of low-resource training by applying iterative back-translation to generate synthetic poetry translation data. The synthetic data will be used alongside true parallel poetry translation data to fine-tune a sequence-to-sequence transformer model. I consider three model setups: 1) an out-of-the-box mBART50 (Tang et al., 2020), a multilingual translation model that has been pre-trained on a general translation task; 2) an mBART50 model that has been fine-tuned on the dataset of true parallel poetry translations; and 3) another mBART50 model that has been fine-tuned

on both true parallel poetry translations and synthetic parallel poetry translations generated through the iterative back-translation process.

A comparison of each model's accuracy using automatic translation metrics, such as BLEU, BERTScore, and COMET, does not give clear evidence that one iteration of back-translation improves poetry translation. The results even indicate that two iterations of back-translation detracts from model performance. In considering these metrics and the data synthesized during the training process, I conclude that while iterative back-translation may generate increasingly semantically accurate data, the poetic nature of the data often gets reduced, which may lead to poorer results. Further research with larger, multilingual datasets, alternative training parameters, or human evaluation should be performed to provide additional evidence to corroborate this paper's findings on iterative back-translation in neural poetry translation.

3 Related Work

3.1 Neural Poetry Translation

The problem of neural poetry translation was first addressed by Ghazvininejad et al. (2018). Their paper introduced the aforementioned semantic and structural complexities of poetry translation, though the authors focused particularly on preserving rhythm and rhyme while translating poetry. Their most highly rated model was an Long Short-Term Memory neural network (LSTM) that used finite-state automata to detect rhythm and rhyme in the source poem, which were then used to constrain the generated translation. Their results were successful in retaining the targeted aspects of poetic structure and were satisfactory 78.2% of the time based on human evaluation. The structural constraints imposed on the translations, and possibly the quality of the base LSTM model, did result in certain lines of translation being incorrect or ungrammatical. For example, their model predicted "I used to close my hair" when the target was "And drank the morning air"; one can see that the rhythm and rhyme are correct in the prediction but the semantic meaning is completely wrong. The use of transformers, as in this paper, hopefully resolves this issue of semantically incorrect predictions.

Neural poetry translation was revisited by Chakrabarty et al. (2021). They considered the task to be a low-resource domain within the broader subject of neural machine translation, and therefore contributed to neural poetry translation by applying techniques used for other low-resource translation tasks. Using automatic metrics like BLEU and BERTScore as well as human evaluation, the authors concluded that multilingual models fine-tuned only on "in-domain" poetry data outperform bilingual models or models fine-tuned only on general translation data, even if those dataset sizes are much larger. The authors also provide parallel datasets from six European languages to English, which I use for this paper's experiments.

3.2 Iterative Back-Translation

Iterative back-translation is a method for generating synthetic parallel translation data from monolingual corpora (Hoang et al., 2018). Iterative back-translation is particularly useful in low-resource contexts because there are generally more monolingual corpora, especially in English, which can then be leveraged to make synthetic data for language pairs with lesser amounts of parallel data. This algorithm is based on the previously developed strategy of back-translation, where training a model to generate parallel translation data, even if low-quality, was proven to improve low-resource translation models, typically by about 1 BLEU point (Currey et al., 2017). The iterative property of Hoang et al. (2018)'s algorithm extends upon this basic back-translation algorithm by repeatedly training the translation models to increase the quality of synthetic data, leading to performance increases of 0.7 to 5.9 BLEU points.

3.3 mBART50

On the basis of Chakrabarty et al. (2021)'s empirical observations on the efficacy of multilingual models, the primary model considered in this paper is mBART50 (Tang et al., 2020). mBART50 is a multilingual version of the BART sequence-to-sequence transformer model invented by Lewis et al. (2019) that has been pre-trained on a denoising task in 50 languages. The multilingual nature of mBART50 allows for one model to be used for multiple translation directions at once. As seen in Chakrabarty et al. (2021), the multilingual characteristic allows for transfer learning between

languages - that is, fine-tuning in one language can improve translation results on other languages, including languages not seen in fine-tuning.

Using a multilingual model also has benefits in the iterative back-translation process. The original iterative back-translation system had to maintain two bilingual models, one for each direction of translation - source-to-target and target-to-source (Hoang et al., 2018). Being a multilingual model, mBART50 simplifies this process because it can perform any direction of translation. Therefore, the iterative back-translation process only needs to maintain one multilingual model for both directions of data synthesization.

4 Approach

4.1 `mbart50-large-many-to-many-mmt`

The iterative back-translation system was fine-tuned using the `mbart50-large-many-to-many-mmt` checkpoint, retrieved from the corresponding Hugging Face repository. This is a checkpoint that has been pre-trained on the multilingual machine translation task using the ML50 Benchmark. This model has the ability to translate between any pair of languages, hence the label "many-to-many". This is a parallel to the `mbart50-large-many-to-one-mmt` checkpoint that was fine-tuned on in Chakrabarty et al. (2021), though now allowing for translation from English to the source languages, as is required in back-translation. This original checkpoint served as the baseline model for this project.

4.2 Supervised Fine-Tuning

To fine-tune mBART50, a sequence-to-sequence (seq2seq) transformer model based on the `mbart50-large-many-to-many-mmt` checkpoint was initialized. For each data point, a language ID was prepended to both the source and label in order for the multilingual model to determine the translation direction. These augmented data points were then passed to the mBART50 seq2seq model to minimize label-smoothed cross-entropy loss, which serves as a proxy to the non-differentiable BLEU metric.

4.3 Synthetic Data Generation

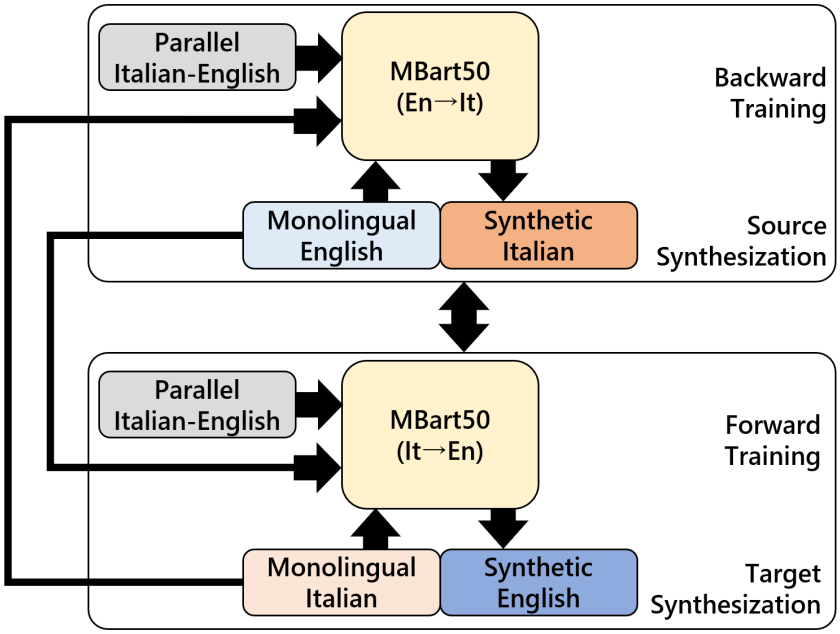

Figure 1: A visual representation of the iterative back-translation algorithm, using the case study of Italian to English (It→En).

The iterative back-translation model relies on the generation of synthetic parallel poetry translation data. This synthetic data was generated by the mBART50 model after each round of training in each direction. In order to do so, I fine-tuned the mBART50 model on the given parallel dataset in the "backward" English-to-source direction (En→S). Synthetic source-language lines were then generated using the monolingual English corpus. The system used the parallel and synthetic data to fine-tune the same mBART50 model in the "forward" source-to-English direction (S→En). This fine-tuned the model on the original source-to-English task. Lastly, the model was used to generate synthetic English data from the monolingual source-language corpus, which was used for fine-tuning in the next iteration.

Synthetic data was generated at a ratio of 1 synthetic translation pair for every 1 real translation pair. Hoang et al. (2018) indicates that higher ratios of synthetic data to real data, such as 2 or 3 times, leads to slightly better performance by 0.1 to 0.3 BLEU points, but due to the limited size of the provided source-language datasets and limited computational resources, I chose a more modest ratio to ensure there was still sufficient true parallel data to ground the fine-tuning process.

4.4 Iterative Procedure

The process described in Section 4.3 constitutes just one iteration of back-translation. In iterating upon the back-translation process, the quality of the synthetic data has previously been shown to increase (Hoang et al., 2018), resulting in a significantly larger but still accurate dataset for the mBART50 model to be fine-tuned on. This paper considers models trained on synthetic data produced after one and two iterations of back-translation. Hoang et al. (2018)'s algorithm is presented below with a slight adjustment to account for the multilingual nature of the translation model.

Algorithm 1 Iterative Back-Translation

Input: parallel data D^p, monolingual sources D^s, monolingual targets D^t
Let synthetic data $S \leftarrow \emptyset$
Let $T \leftarrow D^p$
while number of iterations not reached **do**
  Train multilingual model Θ on T from target to source
  Use Θ to generate synthetic sources $\hat{D}^s$ using D^t
  Let $S_1 \leftarrow \{\hat{D}^s, D^t\}$
  Let $T \leftarrow D^p \cup S_1$
  Train multilingual model Θ on T from source to target
  Use Θ to generate synthetic targets $\hat{D}^t$ using D^s
  Let $S_2 \leftarrow \{D^s_1, \hat{D}^t\}$
  Let $T \leftarrow D^p \cup S_2$
end while
Output: Fine-tuned multilingual model Θ

5 Experiments

5.1 Data

For parallel poetry translation datasets, I used the datasets provided by Chakrabarty et al. (2021) at https://github.com/tuhinjubcse/PoetryTranslationEMNLP2021, which includes parallel translation data from six source languages - Italian, Spanish, Portuguese, Dutch, German, and Russian - to the target language English. Each dataset contains about 10000 to 30000 training points, with around 1000 validation and testing points each. This paper focuses on the Italian→English direction as a case study. I split the provided parallel Italian-English data into two halves; one half remained the true parallel data, the other half was used only for the Italian data to become the monolingual Italian dataset. Exact dataset sizes used in this project can be found in Table 1.

A monolingual English poem dataset containing 3.09 million lines was taken from the open-source Gutenberg Poetry Corpus, located at https://huggingface.co/datasets/biglam/gutenberg-poetry-corpus (Parrish, 2018). A subset of data points that equaled the size of the parallel training dataset was

randomly sampled to be used for iterative back-translation.

Corpus Language(s)	Train	Valid	Test
Italian-English	12,086	2,590	2,591
Monolingual Italian	12,086	-	-
Monolingual English	12,086	-	-

Table 1: Overview of the datasets used for training, validation, and testing. Note that the monolingual datasets are used solely for generating synthetic data to train on.

5.2 Evaluation method

Similar to the combination of metrics used in Chakrabarty et al. (2021), I used four metrics to evaluate model accuracy: BLEU as implemented in SacreBLEU (Post, 2018), BERTScore (Zhang et al., 2020), COMET (Rei et al., 2022), and chrF++ (Popović, 2017). BLEU is the canonical translation metric which measures n-gram similarity between a model translation and the references (Papineni et al., 2002). BERTScore leverages BERT's contextual embeddings to measure semantic similarity between prediction and gold truths. COMET is a similar metric in that it uses a transformer model to rate translations, but it is designed specifically for machine translation tasks. In this project, COMET is evaluated using the recommended `Unbabel/wmt22-comet-da` model. Lastly, chrF++ is another n-gram similarity metric that extends on BLEU by using F-scores and adding additional n-grams that more accurately capture morphological similarity. All four metrics were accessed through the Hugging Face `evaluate` library (Wolf et al., 2020).

5.3 Experimental details

The baseline model was the `mbart50-large-many-to-many-mmt` multilingual transformer model of 611 million parameters that has already been trained on multilingual translation Tang et al. (2020). This model was used out-of-the-box with no further tweaking.

The second model was the mBART50 model fine-tuned on just the parallel data, following the implementation by Chakrabarty et al. (2021). The training process consisted of 3 epochs, which took about 6 hours using a single Nvidia RTX 4060. To be consistent with Chakrabarty et al. (2021), the default Hugging Face training parameters were used except for batch size, which was lowered to 8 due to computation limitations.

The final models used the iterative back-translation algorithm to generate data and then fine-tune the mBART50 model on both the parallel and synthetic data. Two models were considered under this approach: one after one iteration of back-translation and the other after two iterations. Each iteration consisted of 3 epochs of training using the parallel data (and additionally the synthetic data in the second iteration) in the English→Italian direction for data synthesizing, and then 3 epochs using parallel and synthetic data in the Italian→English direction for the target translation task. The first iteration took about 18 hours on a single Nvidia RTX 4060. The second iteration took another 24 hours on the same machine. The models were trained using BFloat16 mixed precision training for increased efficiency.

All models were evaluated by producing English translations from the test set of Italian poetry. The metrics were obtained by comparison to the provided English labels.

6 Results and Discussion

6.1 Automatic Evaluation Metrics

The results in Table 2 do not provide clear evidence that iterative back-translation improves fine-tuning mBART50 for poetry translation compared to standard fine-tuning. The model trained with one iteration of back-translation (BT_1) gives similar metrics compared to the standard fine-tuned model (FT), with a higher BLEU by 0.12 points but lower BERTScore and COMET by just 0.001 and 0.005, respectively. Due to the noisy nature of the data, such small and inconsistent differences

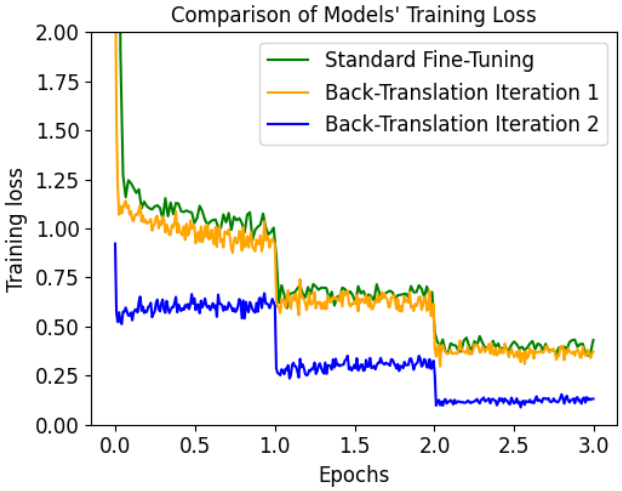

Figure 2: Training loss for the standard fine-tuning and both iterative back-translation models after 3 epochs of training.

Direction	Model	BLEU	BERTScore	COMET	chrF++
Italian→English	$Base$	8.40	0.797	0.573	26.73
	FT	15.73	**0.847**	**0.629**	**36.20**
	BT_1	**15.85**	0.846	0.624	35.97
	BT_2	14.49	0.841	0.611	34.36

Table 2: Model metrics after fine-tuning. $Base$ is the non-fine-tuned baseline; FT is the standard fine-tuned model after 3 epochs; BT_1 and BT_2 are the iterative back-translation models after 1 and 2 iterations, respectively.

cannot be interpreted in favor of either model. The close performance of these two models is also supported by their similar training loss after 3 epochs, as seen in Fig. 2.

With two iterations of back-translation (BT_2), the metrics see more notable disparities compared to the FT model, with BLEU, BERTScore, and COMET lower by 1.44, 0.006, and 0.018, respectively. However, the training loss continued to decrease over the second iteration of back-translation, becoming noticeably lower than the other models and suggesting better performance (see Fig. 2). This discrepancy is either an indicator that the model was overfitting to the training set or that the training set was becoming less representative of the target task. Considering that the training set changed with the iterations of synthetic data, it is likely to be the case that the data itself negatively influenced training results.

The performance of the iterative back-translation models was ultimately lower than expected, considering that iterative back-translation has proven to improve BLEU by up to 6 points in general contexts (Hoang et al., 2018) and was hypothesized to improve poetry translations in Chakrabarty et al. (2021). It should also be noted that iterative back-translation is much more costly than standard fine-tuning, as the model must be fine-tuned in both directions and on additional data, leading to training times that are 3-4 times longer than the standard fine-tuning strategy. Considering this additional cost, iterative back-translation would need to provide substantial improvements to translation quality in order to be worth using in practice.

6.2 Quality of Synthetic Data

Due to the artistic nature of poetry, it is important to consider evaluation beyond automatic metrics for neural poetry translation. An analysis of synthesized data will therefore provide insight into how the translation system is working qualitatively. This discussion will focus only on synthesized targets in English for the sake of the reader's comprehension, but the findings are consistent across both the synthetic source and target texts.

Both synthesized sources and targets saw considerable shifts across the two iterations of back-translation. While most changes were semantically insignificant, such as the use of a synonym ("With stars upon its head," → "With stars about its head,"), many of the synthetic data points underwent significant structural change across iterations, such as in the word order, tone, or type of vocabulary. These factors may not necessarily change the literal semantic meaning of the texts, but they are crucial aspects to writing poetry.

	First Target Synthesization	**Second Target Synthesization**
Ex. 1	we pass by ourselves.	passing alone.
Ex. 2	To look at another under new moon;	beneath the new moon, men look at each other.
Ex. 3	The earliest age, fair as gold was,	The primal age was beautiful as gold;
Ex. 4	Is left to thee to see, thou shalt be satisfied;	comes into view, you shall be satisfied;

Table 3: Example of synthetic target (English) data across two iterations of back-translation.

Table 3 provides several examples of significant changes in data synthesization across iterations. Although the length of the average synthetic target point was similar across the two iterations (7.42 words for iteration 1 and 7.46 words for iteration 2), many of the translations in the second iteration used language that is more plain. This means turning florid language into more concise phrases, as in Ex. 1 in Table 3 and conversely turning more vague language into more grammatically sound phrases, as in Ex. 2.

Sentence structures also became simpler. In Ex. 3 of Table 3, the first target synthesization has an unusual, yet poetic, subject-object-verb structure. The second target synthetization converts this to the standard subject-verb-object form, and while it uses a stronger adjective - "primal" instead of "earliest" - the simpler sentence structure may detract from the text's poetic character.

Finally, the use of poetic pronouns decreased in the second iteration, as exemplified by Ex. 4 of Table 3. In this case, "thee" and "thou" were changed to the vernacular "you". This trend is observed across the entire synthetic dataset, where the first iteration contained 271 instances of "thee" while the second iteration contained only 192 instances.

These aspects of the synthetic data show that the data became less poetic over multiple iterations of back-translation. This could have impacted the performance of the translation model, as the model was then being trained on data that did not represent the desired task. Yet, it was expected that with the true parallel translation dataset to ground the model in the task of poetry translation, additional data and epochs of training should still improve model performance, even if the synthetic data was not particularly poetic.

7 Conclusion

This project provides an implementation of iterative back-translation, which was used to train a neural translation model for the literary sub-domain of poetry. The results did not indicate a clear advantage to using iterative back-translation, with metrics on par or even lower than standard fine-tuning and requiring greater computational resources. The limited scope of the project may have induced the negative results, particularly with regards to the limited data and computational resources. This fortunately leaves much room for future research, which would provide more substantive evidence to complement the findings presented here. Specifically, I would like to extend this experiment with the full multilingual dataset provided by Chakrabarty et al. (2021). I also want to reconsider the base model checkpoint, `mbart50-large-many-to-many-mmt`, especially with regards to the data that it was pre-trained on. Lastly, incorporating human judgment in the evaluation process would be very valuable to determine if the predicted translations are truly maintaining a poetic character. My research here will hopefully provide a stepping stone to these future improvements and spur greater interest in neural machine translation for literary forms like poetry.

8 Ethics Statement

As machine translation, and neural poetry translation in particular, continues to improve, we run into concerns about the role of machine learning within an art form such as poetry. One issue

is the accuracy of transferring cultural traditions across languages. In an increasingly globalized world, being able to translate poetry to other languages allows for increased cultural exchange and appreciation of global literary traditions. Although modern NLP has proven to be quite successful at creative tasks, relying on machine translation systems to perform poetry translation runs the risk of losing the artistic value of this poetry. Such inaccurate or low-quality translations would serve to undermine the merit of a particular poem and its author. In a more severe case, an entire literary culture may be deemed as inferior because of the quality of its translations, which completely defeats the original purpose of global cultural exchange through translating poetry.

Another issue is the threat of automatic translation systems to professional translators. Many people rely on their translation work to earn a living and consequently spend significant effort crafting artistic translations that reflect all of the original text's meaning, both literal and figurative. The rise of neural machine translation puts this line of work, and these results of human creativity, at risk if people embrace neural model outputs too strongly. In order to mitigate the risks presented here, we must first reconsider what art is; this aesthetic debate is beyond the scope of this paper, but arguably, the core of art is human expression. Therefore, especially with literary mediums such as poetry, it is important to still maintain a professional human translator's judgment and mediation if a neural translation system is to be used. In deployment, these neural translation models should just be used as tools, not entire pipelines that produce publication-ready content.

A final ethical issue concerns the collection of the parallel poetry data. As I only used publicly published data for this project, I will defer the reader to the ethical statement from the original datasets' paper, which states that ethical considerations were taken to minimize potential harm in creating these datasets (Chakrabarty et al., 2021).

References

Tuhin Chakrabarty, Arkadiy Saakyan, and Smaranda Muresan. 2021. Don't go far off: An empirical study on neural poetry translation. *arXiv preprint arXiv:2109.02972*.

Anna Currey, Antonio Valerio Miceli-Barone, and Kenneth Heafield. 2017. Copied monolingual data improves low-resource neural machine translation. In *Proceedings of the second conference on machine translation*, pages 148–156.

Marjan Ghazvininejad, Yejin Choi, and Kevin Knight. 2018. Neural poetry translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 67–71.

Cong Duy Vu Hoang, Philipp Koehn, Gholamreza Haffari, and Trevor Cohn. 2018. Iterative back-translation for neural machine translation. In *2nd Workshop on Neural Machine Translation and Generation*, pages 18–24. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. pages 311–318.

Allison Parrish. 2018. Gutenberg Poetry Corpus. `https://huggingface.co/datasets/biglam/gutenberg-poetry-corpus`. Accessed: 2024-05-20.

Maja Popović. 2017. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022. COMET-22: Unbabel-IST 2022 submission for the metrics shared task. In *Proceedings of the Seventh Conference on Machine*

Translation (WMT), pages 578–585, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning.

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.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.