# Findings of the CoCo4MT 2023 Shared Task on Corpus Construction for Machine Translation

**Organizers**

| | |
|---|---|
| **Ananya Ganesh**[1] | ananya.ganesh@colorado.edu |
| **Marine Carpuat**[2] | marine@umd.edu |
| **William Chen**[3] | williamchen@cmu.edu |
| **Katharina Kann**[1] | katharina.kann@colorado.edu |
| **Constantine Lignos**[4] | lignos@brandeis.edu |
| **John E. Ortega**[5] | j.ortega@northeastern.edu |
| **Jonne Saleva**[4] | jonnesaleva@brandeis.edu |
| **Shabnam Tafreshi**[2] | stafresh@umd.edu |
| **Rodolfo Zevallos**[6] | rodolfojoel.zevallos@upf.edu |

[1]University of Colorado Boulder
[2]University of Maryland
[3]Carnegie Mellon University
[4]Brandeis University
[5]Northeastern University
[6]Universitat Pompeu Fabra

**Abstract**

This paper provides an overview of the first shared task on choosing beneficial instances for machine translation, conducted as part of the CoCo4MT 2023 Workshop at MTSummit. This shared task was motivated by the need to make the data annotation process for machine translation more efficient, particularly for low-resource languages for which collecting human translations may be difficult or expensive. The task involved developing methods for selecting the most beneficial instances for training a machine translation system without access to an existing parallel dataset in the target language, such that the best selected instances can then be manually translated. Two teams participated in the shared task, namely the `Williams` team and the `AST` team. Submissions were evaluated by training a machine translation model on each submission's chosen instances, and comparing their performance with the chRF++ score. The system that ranked first is by the `Williams` team, that finds representative instances by clustering the training data.

## 1 Introduction

It is a well-known fact that machine translation (MT) systems, especially those that use deep learning, require massive amounts of data. Some of the types of resources available are monolingual, multilingual, translation memories, and lexicons. Those types of resources are gener-

*Proceedings of the Second Workshop on Corpus Generation and Corpus Augmentation for Machine Translation*, pages 22–27
September 5, 2023, Macau SAR, China.

ally created for formal purposes such as parliamentary proceedings (Koehn, 2005), particularly when the data is parallel. The quality and abundance of such resources for niche or rare domains such as medicine or science is limited, meaning that focused annotation efforts are required when an MT system for such domains needs to be developed. Additionally, corpora for low-resource languages, languages with less digital resources available, tends to be less abundant and of lower quality.

While MT systems developed using unsupervised methods and monolingual corpora have been effective to a great extent (Lample et al., 2018; Liu et al., 2020), parallel data is still crucial, particularly in the case of low-resource languages, as shown by Kim et al. (2020). However, collection or annotation of parallel data is constrained by access to bilingual translators, who may be rare or highly expensive. Therefore, making the data annotation process cost effective by ensuring that the translated instances are of high quality and will lead to high-performing MT systems when used for training. For maximum value, it is desirable to have access to this information *before* a dataset in the target language is actually constructed. That is, if the annotation budget only permits a limited number of sentences to be translated, but there is a large number of source language sentences, it is optimal to choose sentences for human translation that are expected to be highly beneficial.

Towards making the annotation process more efficient, in this shared task, we solicit methods for the identification of such beneficial instances effectively without requiring training data in the target language [1]. We provide multi-way parallel data from several high-resource languages such as English and German, which can be used to identify instances that are helpful for model training, such as by observing training dynamics (Bhatnagar et al., 2022). Participants are required to submit the English sentences corresponding to instances chosen by their algorithms as the most beneficial. Notably, this task does not necessarily require training MT models – simple heuristics that can indicate the quality of an instance can also be submitted. The performance of all submissions, including the baselines, are evaluated by training an MT model (specifically, mBART) on the selected instances. We use the chrF++ metric (Popović, 2017) to compare all systems.

The shared task officially began on May 19, 2023 with the release of all training data. Baselines were then added on June 6, 2023 [2]. Interested participants were asked to officially register for the shared task through a Google Forms submission, on which four teams registered. The participation phase concluded on July 21, 2023, until which date submissions could be made by sending text files with the chosen instances to the official CoCo4MT 2023 email address. Of the four teams that registered, only two teams made a submission before the conclusion of the shared task. Both teams described their methods and shared an open-source implementation through a system description paper.

## 2 Data

All data used for model training, evaluation and instance selection is sourced from the Johns Hopkins University Bible corpus (McCarthy et al., 2020). This is a multi-way parallel corpus containing verses from the Christian Bible translated into more than 1600 languages.

**Languages:** As outlined above, we provide data for a set of "high-resource" languages, which are intended to be used for developing systems to select beneficial instances. For this purpose, we choose the languages English, German, Indonesian, and Korean. We also provide data in

[1] Website describing the workshop and shared task is at https://sites.google.com/view/coco4mt

[2] Data, baselines and processing scripts can be found at https://github.com/ananyaganesh/coco4mt-shared-task

the “low-resource” languages of Gujarati, French and Burmese for participants to evaluate the performance of their methods. This setting can be considered to be a simulated low-resource setting, since for the purpose of this dataset, all languages are multi-way parallel. Finally, we evaluate all submissions on the surprise languages of Vietnamese, Lithuanian and Kazakh, not revealed to the participants until the conclusion of the shared task. The data is in the form of source–target translation pairs, with the source language always being English.

**Size and splits:** The multi-way parallel section of the corpus for our languages of interest consists of 34831 sentences. From this, we create training, validation and test sets of sizes 22204, 3919, and 8708 respectively by randomly sampling the original data.

## 3 Evaluation

**Submission format:** Participants were asked to submit indices of the top 20% of the training data (or 4440 sentences), corresponding to the best instances chosen by their systems. We then extract the source and target language sentences corresponding to the indices to prepare data files for MT models.

**Model:** We evaluate submissions by *finetuning* the mBART model (Liu et al., 2020) on the chosen instances. mBART is a multilingual denoising autoencoder trained on data from 25 languages, extracted from Common Crawl (Wenzek et al., 2020). We use the mbart-large-cc25 checkpoint from Facebook, which contains all 10 of our languages of interest in its pretraining data.

**Training:** We use the implementation and the default hyperparameters of mBART-large from the Huggingface hub (Wolf et al., 2020). All data is tokenized with the corresponding mBART-cc-25 sentence-piece tokenizer, and any empty lines on the source side are filtered out prior to training. We train each model for 20 epochs on a single nvidia V100 GPU, and use early-stopping based on validation set performance. We train five random runs of each model, and report the averaged score across all runs.

**Baselines:** We develop two baselines for comparing the submissions to, namely Random and Max. The random baseline randomly samples 20% of all instances from the training set. The max baseline sorts the English sentences in descending order by number of tokens, and selects the top 20%.

**Metrics:** All systems are evaluated with the chrF++ score (Popović, 2017), computed using the sacrebleu toolkit (Post, 2018).

## 4 Submissions

Two teams participated in the CoCo4MT 2023 shared task, under the team names `Williams` and `AST`. We describe their submissions below, and further details can be found in the system description papers attached to the proceedings.

### 4.1 Williams

The algorithm proposed by team `Williams`[3] is based on the idea of clustering training examples to find representative instances that can be chosen for training. Following Zhao et al. (2020), they highlight the importance of “balancing representativeness with redundancy”, that is, making sure that the distribution of the training data is captured, without including multiple instances that are similar to each other. To achieve this objective, they use the SimCSE algorithm to obtain embeddings of each sentence in the training data, and then use cosine distance

[3] `https://github.com/Mark-Hopkins-at-Williams/coco4mt`

| Language | Model | ChrF++ Score |
|---|---|---|
| Development languages | | |
| Gujarati | Random | 28.43 |
| Gujarati | Max | 25.62 |
| Gujarati | Williams | 29.59 |
| Gujarati | AST | **29.80** |
| French | Random | 52.16 |
| French | Max | **54.75** |
| French | Williams | 54.09 |
| French | AST | 53.38 |
| Burmese | Random | 37.11 |
| Burmese | Max | 39.75 |
| Burmese | Williams | 40.00 |
| Burmese | AST | **40.13** |
| Test (Surprise) Languages | | |
| Lithuanian | Random | 42.65 |
| Lithuanian | Max | **43.51** |
| Lithuanian | Williams | 43.43 |
| Lithuanian | AST | 43.11 |
| Kazakh | Random | 31.45 |
| Kazakh | Max | 32.08 |
| Kazakh | Williams | **33.16** |
| Kazakh | AST | 32.30 |
| Vietnamese | Random | 45.13 |
| Vietnamese | Max | 44.85 |
| Vietnamese | Williams | **45.68** |
| Vietnamese | AST | 44.81 |

Table 1: Performances of all models on all development and test languages.

to compute the nearest neighbor of each sentence. They then iteratively select the sentence that is found to be the nearest neighbor of the most number of documents, until 20% of the original training data is selected. They report that this method out-performed other cluster based methods such as selecting cluster centroids.

### 4.2 AST

The algorithm proposed by team AST[4] is based on maximizing the information provided by each sentence, by selecting sentences that are the long, but also contain diverse sets of n-grams. They then greedily select sentences that optimize this objective. Additionally, they use the LaBSE model to compute sentence embeddings for each translation pair in the training set, and filter out sentences that have an LaBSE similarity score of less than 0.5. They also aim to filter out training instances that may potentially be mis-aligned. They do this by translating all English sentences to German and Indonesian using the mBART-50 model, and compute chrF++ scores for all instances. They discard sentences with a score below 20 as they may be misaligned, as well as sentences with a score above 60, as the information contained in them may already be well-represented in the pre-training data.

[4] https://github.com/Mark-Hopkins-at-Williams/coco4mt

| Model | ChrF++ Score |
|---|---|
| Development Languages | |
| Random | 39.22 |
| Max | 40.04 |
| Williams | **41.22** |
| AST | 41.10 |
| Test (Surprise) Languages | |
| Random | 39.74 |
| Max | 40.14 |
| Williams | **40.75** |
| AST | 40.07 |

Table 2: Average performance on the development and test languages.

## 5 Results

We first report the performance of all models on all languages in Table 1, that is, both the development languages released to the participants, and the surprise languages used for judging. On the development languages, we observe the highest scores for all models for French, which is very well represented in the mBART pre-training data with 9000M tokens, and also bears similarities to English. However, all models perform higher on Burmese than Gujarati, despite Gujarati having 140M tokens in CC25 while Burmese only has 56M tokens. We also see that both submissions outperform the baselines on the lower-resource languages of Gujarati and Burmese, but not on French. Although the AST system achieves the best performance on two development languages, on average, as seen in Table 2, the Williams submission performs best, with a score of 41.22, while the AST submission closely follows with an average score of 41.10.

On the test set of languages, or surprise languages, we see some similar trends. Highest performance for all models is seen on Vietnamese, which is the most prevalent in CC25 with 24000M tokens. Lithuanian, which has 1800M tokens comes next, and lowest performance is on Kazakh which has 476M tokens in the pretraining data. Although the max baseline outperforms both submissions on Lithuanian, the Williams system outperforms all other systems on all other languages. We further see that the performance of the AST submission is very close to the max baseline, potentially due to the submission also focusing on the longest sentences in its ranking. Finally, as seen in Table 2, the best performing system on average on the test languages is also the Williams system, which we officially judge as the winner of the shared task.

We highlight the fact that all three non-random methods described here are model-agnostic methods, that can identify instances with just access to parallel data and sentence embedding methods. The simple heuristic of choosing the longest sentences holds up well in comparison to more nuanced methods, even outperforming the others for Lithuanian. We leave it to future work to explore more advanced heuristics as well as develop model-specific methods to choose beneficial instances for machine translation.

## 6 Conclusion

In this overview paper, we presented the results of the first CoCo4MT 2023 shared task. The goal of the task was to discover methods to improve cost-efficiency of the machine translation annotation process, by identifying beneficial instances even without an existing parallel dataset. Participants were given access to data from four languages from the JHU Bible corpus to develop their algorithms, and three more languages to evaluate their systems. The task received two submissions, which were evaluated on three surprise or test languages. The winner of the shared task is the submission by team `Williams`, which clusters all training set instances, and selects representative examples while minimizing redundancies. We hope that the findings of this task will spur more research on improving annotation efficiency, particularly for low-resource languages.

## References

Bhatnagar, R., Ganesh, A., and Kann, K. (2022). CHIA: CHoosing instances to annotate for machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 7299–7315, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Kim, Y., Graça, M., and Ney, H. (2020). When and why is unsupervised neural machine translation useless? In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 35–44, Lisboa, Portugal. European Association for Machine Translation.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand.

Lample, G., Conneau, A., Denoyer, L., and Ranzato, M. (2018). Unsupervised machine translation using monolingual corpora only. In *International Conference on Learning Representations*.

Liu, Y., Gu, J., Goyal, N., Li, X., Edunov, S., Ghazvininejad, M., Lewis, M., and Zettlemoyer, L. (2020). Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

McCarthy, A. D., Wicks, R., Lewis, D., Mueller, A., Wu, W., Adams, O., Nicolai, G., Post, M., and Yarowsky, D. (2020). The Johns Hopkins University Bible corpus: 1600+ tongues for typological exploration. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2884–2892, Marseille, France. European Language Resources Association.

Popović, M. (2017). chrf++: words helping character n-grams. In *Proceedings of the second conference on machine translation*, pages 612–618.

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, Brussels, Belgium. Association for Computational Linguistics.

Wenzek, G., Lachaux, M.-A., Conneau, A., Chaudhary, V., Guzmán, F., Joulin, A., and Grave, E. (2020). CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Zhao, Y., Zhang, H., Zhou, S., and Zhang, Z. (2020). Active learning approaches to enhancing neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1796–1806, Online. Association for Computational Linguistics.