The Helsinki submission to the AmericasNLP shared task

Raúl Vázquez   Yves Scherrer   Sami Virpioja   Jörg Tiedemann

Department of Digital Humanities
University of Helsinki
firstname.lastname@helsinki.fi

Abstract

The University of Helsinki participated in the AmericasNLP shared task for all ten language pairs. Our multilingual NMT models reached the first rank on all language pairs in track 1, and first rank on nine out of ten language pairs in track 2. We focused our efforts on three aspects: (1) the collection of additional data from various sources such as Bibles and political constitutions, (2) the cleaning and filtering of training data with the OpusFilter toolkit, and (3) different multilingual training techniques enabled by the latest version of the OpenNMT-py toolkit to make the most efficient use of the scarce data. This paper describes our efforts in detail.

1 Introduction

The University of Helsinki participated in the AmericasNLP 2021 Shared Task on Open Machine Translation for all ten language pairs. The shared task is aimed at developing machine translation (MT) systems for indigenous languages of the Americas, all of them paired with Spanish (Mager et al., 2021). Needless to say, these language pairs pose big challenges since none of them benefits from large quantities of parallel data and there is limited monolingual data. For our participation, we focused our efforts mainly on three aspects: (1) gathering additional parallel and monolingual data for each language, taking advantage in particular of the OPUS corpus collection (Tiedemann, 2012), the JHU Bible corpus (McCarthy et al., 2020) and translations of political constitutions of various Latin American countries, (2) cleaning and filtering the corpora to maximize their quality with the OpusFilter toolbox (Aulamo et al., 2020), and (3) contrasting different training techniques that could take advantage of the scarce data available.

We pre-trained NMT systems to produce back-translations for the monolingual portions of the data. We also trained multilingual systems that make use of language labels on the source sentence to specify the target language (Johnson et al., 2017). This has been shown to leverage the information available data across different language pairs and boosts performance on the low-resource scenarios.

We submitted five runs for each language pair, three in track 1 (development set included in training) and two in track 2 (development set not included in training). The best-performing model is a multilingual Transformer pre-trained on Spanish–English data and fine-tuned to the ten indigenous languages. The (partial or complete) inclusion of the development set during training consistently led to substantial improvements.

The collected data sets and data processing code are available from our fork of the organizers' Git repository.[1]

2 Data preparation

A main part of our effort was directed to finding relevant corpora that could help with the translation tasks, as well as to make the best out of the data provided by the organizers. In order to have an efficient procedure to maintain and process the data sets for all the ten languages, we utilized the OpusFilter toolbox[2] (Aulamo et al., 2020). It provides both ready-made and extensible methods for combining, cleaning, and filtering parallel and monolingual corpora. OpusFilter uses a configuration file that lists all the steps for processing the data; in order to make quick changes and extensions programmatically, we generated the configuration file with a Python script.

Figure 1 shows a part of the applied OpusFilter workflow for a single language pair, Spanish–Raramuri, and restricted to the primary training data. The provided training set and (concatenated)

[1] https://github.com/Helsinki-NLP/americasnlp2021-st

[2] https://github.com/Helsinki-NLP/OpusFilter, version 2.0.0-beta.

Proceedings of the First Workshop on Natural Language Processing for Indigenous Languages of the Americas, pages 255–264
June 11, 2021. ©2021 Association for Computational Linguistics

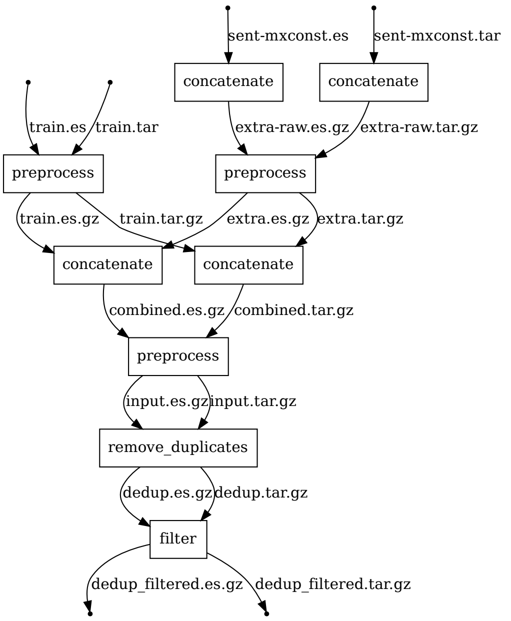

Figure 1: Diagram of the OpusFilter workflow used for Spanish (es) – Raramuri (tar) training data. Boxes are OpusFilter steps and arrows are data files.

additional parallel data are first independently normalized and cleaned (preprocess), then concatenated, preprocessed with common normalizations, filtered from duplicates, and finally filtered from noisy segments.

2.1 Data collection

We collected parallel and monolingual data from several sources. An overview of the resources, including references and URLs, is given in Tables 3 and 4 in the appendix.

Organizer-provided resources The shared task organizers provided parallel datasets for training for all ten languages. These datasets are referred to as *train* in this paper. For some of the languages (Ashaninka, Wixarika and Shipibo-Konibo), the organizers pointed participants to repositories containing additional parallel or monolingual data. We refer to these resources as *extra* and *mono* respectively. Furthermore, the organizers provided development and test sets for all ten language pairs of the shared task (Ebrahimi et al., 2021).

OPUS The OPUS corpus collection (Tiedemann, 2012) provides only few datasets for the relevant languages. Besides the resources for Aymara and Quechua provided by the organizers as official training data, we found an additional parallel dataset for Spanish–Quechua, and monolingual data for Aymara, Guarani, Hñähñu, Nahuatl and Quechua. These resources are also listed under *extra* and *mono*.

Constitutions We found translations of the Mexican constitution into Hñähñu, Nahuatl, Raramuri and Wixarika, of the Bolivian constitution into Aymara and Quechua, and of the Peruvian constitution into Quechua.[3] We extracted the data from the HTML or PDF sources and aligned them with the Spanish version on paragraph and sentence levels. The latter was done using a standard length-based approach with lexical re-alignment, as in hunalign[4] (Varga et al., 2005), using paragraph breaks as hard boundaries. They are part of the *extra* resources.

Bibles The JHU Bible corpus (McCarthy et al., 2020) covers all languages of the shared task with at least one Bible translation. We found that some translations were near-duplicates that only differed in tokenization, and removed them. For those languages for which several dialectal varieties were available, we attempted to select subsets based on the target varieties of the shared task, as specified by the organizers (see Tables 3 and 4 for details). All Spanish Bible translations in the JHUBC are limited to the New Testament. In order to maximize the amount of parallel data, we substituted them by full-coverage Spanish Bible translations from Mayer and Cysouw (2014).[5]

Since we have multiple versions of the Bible in Spanish as well as in some of the target languages, we applied the `product` method in OpusFilter to randomly take at most 5 different versions of the same sentence (skipping empty and duplicate lines).

2.2 Data normalization and cleaning

We noticed that some of the corpora in the same language used different orthographic conventions and had other issues that would hinder NMT model training. We applied various data normalization

[3] Two additional resources, a translation of a Peruvian law into Shipibo-Konibo and a translation of the Paraguayan constitution into Guarani, are provided on our repository, but they became available too late to be included in the translation models. They are listed under *extra** in Tables 3 and 4.

[4] `https://github.com/danielvarga/hunalign`

[5] We would like to thank Garrett Nicolai for helping us with the conversion.

language	code	train	extra	combined	dedup	filtered	bibles	monoling	backtr	dev
Ashaninka	cni	3883	0	3883	3860	3858	38846	13195	17278	883
Aymara	aym	6531	8970	15501	8889	8352	154520	16750	17886	996
Bribri	bzd	7508	0	7508	7303	7303	38502	0	0	996
Guarani	gn	26032	0	26032	14495	14483	39457	40516	62703	995
Hñähñu	oto	4889	2235	7124	7056	7049	39726	537	366	599
Nahuatl	nah	16145	2250	18395	17667	17431	39772	9222	8450	672
Quechua	quy	125008	284517	409525	260680	228624	154825	60399	68503	996
Raramuri	tar	14720	2255	16975	16815	16529	39444	0	0	995
Shipibo-Konibo	shp	14592	28936	43528	28854	28854	79341	23595	38329	996
Wixarika	hch	8966	2654	11620	11541	11525	39756	511	493	994

Table 1: Numbers of segments in the data sets (train: training set provided by the organizers, extra: additional training data collected by the organizers and us, combined: combined training data, dedup: combined training without duplicates, filtered: training data filtered with all filters, bibles: generated Bible data segments after filtering, monoling: monolingual data after filtering, backtr: back-translations created from monolingual data after filtering, dev: development set)

and cleaning steps to improve the quality of the data, with the goal of making the training data more similar to the development data (which we expected to be similar to the test data).

For Bribri, Raramuri and Wixarika, we found normalization scripts or guidelines on the organizers' Github page or sources referenced therein (cf. the *norm* entries in Tables 3 and 4). We reimplemented them as custom OpusFilter preprocessors.

Bribri, Hñähñu, Nahuatl, and Raramuri training sets were originally tokenized. Following our decision to use untokenized input for unsupervised word segmentation, we detokenized the respective corpora with the Moses detokenizer supported by OpusFilter, using the English patterns.

Finally, for all datasets, we applied OpusFilter's WhitespaceNormalizer preprocessor, which replaces all sequences of whitespace characters with a single space.

2.3 Data filtering

The organizer-provided and extra training data sets were concatenated before the filtering phase. Then all exact duplicates were removed from the data using OpusFilter's duplicate removal step. After duplicate removal, we applied some predefined filters from OpusFilter. Not all filters were applied to all languages; instead, we selected the appropriate filters based on manual observation of the data and the proportion of sentences removed by the filter. Appendix A describes the filters in detail.

2.4 Back-translations

We translated all monolingual data to Spanish, using early versions of both Model A and Model B (see Section 3), in order to create additional synthetic parallel training data. A considerable amount of the back-translations produced by Model A ended up in a different language than Spanish, whereas some translations by Model B remained empty. We kept both outputs, but aggressively filtered them (see Appendix A), concatenated them, and removed exact duplicates.

2.5 Data sizes

For most language pairs, the Bibles made up the largest portion of the data. Thus we decided to keep the Bibles separate from the other smaller, but likely more useful, training sources. Table 1 shows the sizes of the training datasets before and after filtering as well as the additional datasets. It can be seen that there is a difference of almost two orders of magnitude between the smallest (cni) and largest (quy) combined training data sets. The addition of the Bibles and back-translations evens out the differences to some extent.

2.6 Spanish–English data

Model B (see below) takes advantage of abundant parallel data for Spanish–English. These resources come exclusively from OPUS (Tiedemann, 2012) and include the following sources: *OpenSubtitles, Europarl, JW300, GlobalVoices, NewsCommentary, TED2020, Tatoeba, bible-uedin*. All corpora are again filtered and deduplicated, yielding 17,5M sentence pairs from OpenSubtitles and 4,4M sentence pairs from the other sources taken together. During training, both parts are assigned the same weight to avoid overfitting on subtitle data. The Spanish–English *WMT-News* corpus, also from OPUS, is used for validation.

Data	Model	Run	aym	bzd	cni	gn	hch	nah	oto	quy	shp	tar	Average
dev	B-50dev	1	0.390	0.392	0.414	0.408	0.409	0.426	0.313	0.457	0.452	0.317	0.398
	A-50dev	3	0.330	0.322	0.385	0.337	0.351	0.359	0.251	0.361	0.352	0.272	0.332
	B-0dev	5	0.327	0.238	0.268	0.311	0.299	0.298	0.147	0.338	0.317	0.196	0.274
	A-0dev	4	0.245	0.188	0.240	0.260	0.255	0.251	0.138	0.245	0.292	0.159	0.227
test	B-100dev	2	0.310	0.213	0.332	0.376	0.360	0.301	0.228	0.394	0.399	0.258	0.317
	B-50dev	1	0.302	0.204	0.324	0.367	0.348	0.294	0.191	0.383	0.380	0.248	0.304
	A-50dev	3	0.261	0.177	0.306	0.311	0.311	0.273	0.181	0.318	0.286	0.216	0.264
	B-0dev	5	0.283	0.165	0.258	0.336	0.304	0.266	0.147	0.343	0.329	0.184	0.262
	A-0dev	4	0.216	0.130	0.236	0.276	0.254	0.243	0.141	0.252	0.294	0.155	0.220

Table 2: chrF2 scores for the five submissions, computed on the development set and test set. Note that only 50% of the development set is used for evaluation for the *50dev* submissions. The chrF2 scores for *B-100dev* on the development set are all above 0.98, but they are not meaningful since it was fully included in training. The Run column provides the numeric IDs with which our submissions are listed in the overview paper.

3 Models

We experimented with two major model setups, which we refer to by A and B below. Both are multilingual NMT models based on the Transformer architecture (Vaswani et al., 2017) and are implemented with OpenNMT-py 2.0 (Klein et al., 2017). All models were trained on a single GPU.

The training data is segmented using SentencePiece (Kudo and Richardson, 2018) subword models with 32k units, trained jointly on all languages. Following our earlier experience (Scherrer et al., 2020), subword regularization (Kudo, 2018) is applied during training. Further details of the configurations are listed in Appendix B.

3.1 Model A

Model A is a multilingual translation model with 11 source languages (10 indigenous languages + Spanish) and the same 11 target languages. It is trained on all available parallel data in both directions as well as all available monolingual data. The target language is specified with a language label on the source sentence (Johnson et al., 2017).

The model was first trained for 200 000 steps, weighting the Bibles data to occur only 0.3 times as much as all the other corpora. We picked the last checkpoint, since it attained the best accuracy and perplexity in the combined development set. This model constitutes submission *A-0dev*.

Then, independently for each of the languages, we fine-tuned this model for another 2 500 steps on language-specific data, including 50% of the development set of the corresponding language. These models, one per language, constitute submission *A-50dev*.

3.2 Model B

Model B is a multilingual translation model with one source language (Spanish) and 11 target languages (10 indigenous languages + English). It is trained on all available parallel data with Spanish on the source side using target language labels.[6]

The training takes place in two phases. In the first phase, the model is trained on 90% of Spanish–English data and 1% of data coming from each of the ten American languages. With this first phase, we aim to take advantage of the large amounts of data to obtain a good Spanish encoder. In the second phase, the proportion of Spanish–English data is reduced to 50%.[7]

We train the first phase for 100k steps and pick the best intermediate savepoint according to the English-only validation set, which occurred after 72k steps. We then initialize two phase 2 models with this savepoint. For model *B-0dev*, we change the proportions of the training data and include the back-translations. For model *B-50dev*, we additionally include a randomly sampled 50% of each language's development set. We train both models until 200 000 steps and pick the best intermediate savepoint according to an eleven-language validation set, consisting of *WMT-News* and the remaining halves of the ten development sets.

Since the inclusion of development data showed massive improvements, we decided to continue training from the best savepoint of *B-50dev* (156k), adding also the remaining half of the development

[6] To generate the back-translations, we used an analogous, but distinct model trained on 11 source languages and one target language.

[7] We experimented also with language-specific second phase training, but ultimately opted for a single run combining all eleven language pairs.

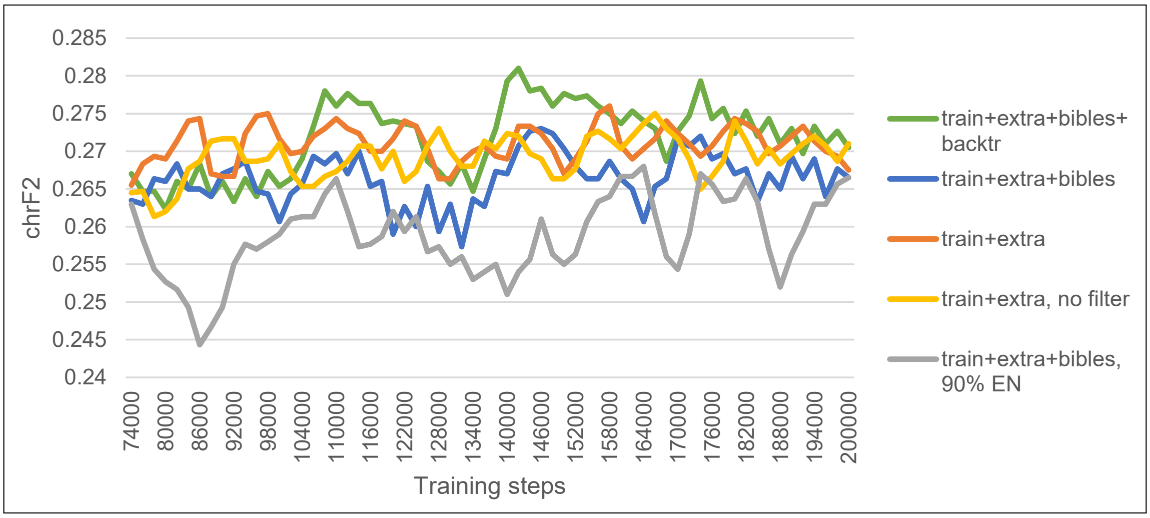

Figure 2: ChrF2 scores obtained with different training configurations of model B. Note: to improve the readability of the graph, the plotted values are smoothed by averaging over three consecutive training steps.

set to the training data. This model, referred to as *B-100dev*, was trained for an additional 14k steps until validation perplexity reached a local minimum.

4 Results

We submitted three systems to track 1 (development set allowed for training), namely *A-50dev*, *B-50dev* and *B-100dev*, and two systems to track 2 (development set not allowed for training), namely *A-0dev* and *B-0dev*. The results are in Table 2.

In track 1, our model *B-100dev* reached first rank and *B-50dev* reached second rank for all ten languages. Model *A-50dev* was ranked third to sixth, depending on the language. This shows that model B consistently outperformed model A, presumably thanks to its Spanish–English pre-training. Including the full development set in training (*B-100dev*) further improves the performance, although this implies that savepoint selection becomes guesswork.

For track 2, the tendency is similar. Model *B-0dev* was ranked first for nine out of ten languages, taking 2nd rank for Spanish–Quechua. *A-0dev* was ranked second to fourth on all except Quechua.[8]

4.1 Ablation study

We investigate the impact of our data selection strategies via an ablation study where we repeat the second training phase of model B with several variants of the *B-0dev* setup. In Figure 2 we show intermediate evaluations on the concatenation of the 10 development sets every 2000 training steps.

The green curve, which corresponds to the *B-0dev* model, obtains the highest maximum scores. The impact of the back-translations is considerable (blue vs. green curve) despite their presumed low quality. The addition of Bibles did not improve the chrF2 scores (blue vs. orange curve). We presume that this is due to the mismatch in linguistic varieties, spelling and genre. It would be instructive to break down this effect according to the language.

The application of the OpusFilter pipeline to the *train* and *extra* data (yellow vs. orange curve) shows a positive effect at the beginning of the training, but this effect fades out later.

Finally, and rather unsurprisingly, our corpus weighting strategy (50% English, 50% indigenous languages, blue curve) outperforms the weighting strategy employed during the first training phase (90% English, 10% indigenous languages, grey curve). It could be interesting to experiment with even lower proportions of English data, taking into account the risk of catastrophic forgetting.

5 Conclusions

In this paper, we describe our submissions to the AmericasNLP shared task, where we submitted translations for all ten language pairs in both tracks. Our strongest system is the result of gathering additional relevant data, carefully filtering the data for each language pair and pre-training a Transformer-based multilingual NMT system with large Spanish-English parallel data. Except for Spanish-Quechua in track 2, all our submissions ranked top for both tracks.

[8] After submission, we noticed that the Quechua backtranslations were generated with the wrong model. This may explain the poor performance of our systems on this language.

Acknowledgments

This work is part of the FoTran project, funded by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement № 771113).

References

Željko Agić and Ivan Vulić. 2019. JW300: A wide-coverage parallel corpus for low-resource languages. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3204–3210, Florence, Italy. Association for Computational Linguistics.

Mikko Aulamo, Sami Virpioja, and Jörg Tiedemann. 2020. OpusFilter: A configurable parallel corpus filtering toolbox. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 150–156, Online. Association for Computational Linguistics.

David Brambila. 1976. *Diccionario Raramuri – Castellano (Tarahumara)*. Obra Nacional de la Buena Prensa, México.

Gina Bustamante, Arturo Oncevay, and Roberto Zariquiey. 2020. No data to crawl? monolingual corpus creation from PDF files of truly low-resource languages in Peru. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 2914–2923, Marseille, France. European Language Resources Association.

Luis Chiruzzo, Pedro Amarilla, Adolfo Ríos, and Gustavo Giménez Lugo. 2020. Development of a Guarani - Spanish parallel corpus. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2629–2633, Marseille, France. European Language Resources Association.

Rubén Cushimariano Romano and Richer C. Sebastián Q. 2008. Ñaantsipeta asháninkaki birakochaki. diccionario asháninka-castellano. versión preliminar. `http://www.lengamer.org/publicaciones/diccionarios/`.

Abteen Ebrahimi, Manuel Mager, Arturo Oncevay, Vishrav Chaudhary, Luis Chiruzzo, Angela Fan, John Ortega, Ricardo Ramos, Annette Rios, Ivan Vladimir, Gustavo A. Giménez-Lugo, Elisabeth Mager, Graham Neubig, Alexis Palmer, Rolando A. Coto Solano, Ngoc Thang Vu, and Katharina Kann. 2021. Americasnli: Evaluating zero-shot natural language understanding of pretrained multilingual models in truly low-resource languages.

Isaac Feldman and Rolando Coto-Solano. 2020. Neural machine translation models with back-translation for the extremely low-resource indigenous language Bribri. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3965–3976, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Ana-Paula Galarreta, Andrés Melgar, and Arturo Oncevay. 2017. Corpus creation and initial SMT experiments between Spanish and Shipibo-konibo. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 238–244, Varna, Bulgaria. INCOMA Ltd.

Ximena Gutierrez-Vasques, Gerardo Sierra, and Isaac Hernandez Pompa. 2016. Axolotl: a web accessible parallel corpus for Spanish-Nahuatl. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4210–4214, Portorož, Slovenia. European Language Resources Association (ELRA).

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. 2017. Opennmt: Open-source toolkit for neural machine translation. In *Proc. ACL*.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Manuel Mager, Diónico Carrillo, and Ivan Meza. 2018. Probabilistic finite-state morphological segmenter for wixarika (huichol) language. *Journal of Intelligent & Fuzzy Systems*, 34(5):3081–3087.

Manuel Mager, Arturo Oncevay, Abteen Ebrahimi, John Ortega, Annette Rios, Angela Fan, Ximena Gutierrez-Vasques, Luis Chiruzzo, Gustavo Giménez-Lugo, Ricardo Ramos, Anna Currey, Vishrav Chaudhary, Ivan Vladimir Meza Ruiz, Rolando Coto-Solano, Alexis Palmer, Elisabeth Mager, Ngoc Thang Vu, Graham Neubig, and Katharina Kann. 2021. Findings of the AmericasNLP 2021 Shared Task on Open Machine Translation for

Indigenous Languages of the Americas. In *Proceedings of theThe First Workshop on NLP for Indigenous Languages of the Americas*, Online. Association for Computational Linguistics.

Jesús Manuel Mager Hois, Carlos Barron Romero, and Ivan Vladimir Meza Ruíz. 2016. Traductor estadístico wixarika - español usando descomposición morfológica. *COMTEL*, 6.

Thomas Mayer and Michael Cysouw. 2014. Creating a massively parallel Bible corpus. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3158–3163, Reykjavik, Iceland. European Language Resources Association (ELRA).

Arya D. McCarthy, Rachel Wicks, Dylan Lewis, Aaron Mueller, Winston Wu, Oliver Adams, Garrett Nicolai, Matt Post, and David Yarowsky. 2020. The Johns Hopkins University Bible corpus: 1600+ tongues for typological exploration. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2884–2892, Marseille, France. European Language Resources Association.

Elena Mihas. 2011. *Añaani katonkosatzi parenini, El idioma del alto Perené*. Milwaukee, WI: Clarks Graphics.

Héctor Erasmo Gómez Montoya, Kervy Dante Rivas Rojas, and Arturo Oncevay. 2019. A continuous improvement framework of machine translation for Shipibo-konibo. In *Proceedings of the 2nd Workshop on Technologies for MT of Low Resource Languages*, pages 17–23, Dublin, Ireland. European Association for Machine Translation.

John Ortega, Richard Alexander Castro-Mamani, and Jaime Rafael Montoya Samame. 2020. Overcoming resistance: The normalization of an Amazonian tribal language. In *Proceedings of the 3rd Workshop on Technologies for MT of Low Resource Languages*, pages 1–13, Suzhou, China. Association for Computational Linguistics.

Prokopis Prokopidis, Vassilis Papavassiliou, and Stelios Piperidis. 2016. Parallel Global Voices: a collection of multilingual corpora with citizen media stories. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 900–905, Portorož, Slovenia. European Language Resources Association (ELRA).

Yves Scherrer, Stig-Arne Grönroos, and Sami Virpioja. 2020. The University of Helsinki and aalto university submissions to the WMT 2020 news and low-resource translation tasks. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1129–1138, Online. Association for Computational Linguistics.

Jörg Tiedemann. 2020. The Tatoeba translation challenge – realistic data sets for low resource and multilingual MT. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1174–1182, Online. Association for Computational Linguistics.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

D. Varga, L. Németh, P. Halácsy, A. Kornai, V. Trón, and V. Nagy. 2005. Parallel corpora for medium density languages. In *Proceedings of Recent Advances in Natural Language Processing (RANLP-2005)*, pages 590–596.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, Long Beach, California, USA.

Raúl Vázquez, Umut Sulubacak, and Jörg Tiedemann. 2019. The University of Helsinki submission to the WMT19 parallel corpus filtering task. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 294–300, Florence, Italy. Association for Computational Linguistics.

A OpusFilter settings

The following filters were used for the training data except for back-translated data and Bibles:

- LengthFilter: Remove sentences longer than 1000 characters. Applied to Aymara, Nahuatl, Quechua, Raramuri.
- LengthRatioFilter: Remove sentences with character length ratio of 4 or more. Applied to Ashaninka, Aymara, Guarani, Hñähñu, Nahuatl, Quechua, Raramuri, Wixarika.
- CharacterScoreFilter: Remove sentences for which less than 90% characters are from the Latin alphabet. Applied to Aymara, Quechua, Raramuri.
- TerminalPunctuationFilter: Remove sentences with dissimilar punctuation; threshold -2 (Vázquez et al., 2019). Applied to Aymara, Quechua.
- NonZeroNumeralsFilter: Remove sentences with dissimilar numerals; threshold 0.5 (Vázquez et al., 2019). Applied to Aymara, Quechua, Raramuri, Wixarika.

The Bribri and Shipibo-Konibo corpora seemed clean enough that we did not apply any filters for them.

After generating the Bible data, we noticed that some of the lines contained only a single 'BLANK' string. The segments with these lines were removed afterwards.

From the provided monolingual datasets, we filtered out sentences with more than 500 words.

The back-translated data was filtered with the following filters:

- LengthRatioFilter with threshold 2 and word units
- CharacterScoreFilter with Latin script and threshold 0.9 on the Spanish side and 0.7 on the other side
- LanguageIDFilter with a threshold of 0.8 for the Spanish side only.

B Hyperparameters

Model A uses a 6-layered Transformer with 8 heads, 512 dimensions in the embeddings and 1024 dimensions in the feed-forward layers. The batch size is 4096 tokens, with an accumulation count of 8. The Adam optimizer is used with beta1=0.9 and beta2=0.998. The Noam decay method is used with a learning rate of 3.0 and 40000 warm-up steps. Subword sampling is applied during training (20 samples, $\alpha = 0.1$).

Model B uses a 8-layered Transformer with 16 heads, 1024 dimensions in the embeddings and 4096 dimensions in the feed-forward layers. The batch size is 9200 tokens in phase 1 and 4600 tokens in phase 2, with an accumulation count of 4. The Adam optimizer is used with beta1=0.9 and beta2=0.997. The Noam decay method is used with a learning rate of 2.0 and 16000 warm-up steps. Subword sampling is applied during training (20 samples, $\alpha = 0.1$). As a post-processing step, we removed the `<unk>` tokens from the outputs of model B.

Aymara aym	train	GlobalVoices (Tiedemann, 2012; Prokopidis et al., 2016)
	extra	BOconst: https://www.kas.de/c/document_library/get_file?uuid=8b51d469-63d2-f001-ef6f-9b561eb65ed4&groupId=288373
	bibles	*ayr-x-bible-2011-v1, ayr-x-bible-1997-v1*
	mono	Wikipedia crawls (Tiedemann, 2020)
Bribri bzd	train	(Feldman and Coto-Solano, 2020)
	bibles	*bzd-x-bible-bzd-v1*
	norm	https://github.com/AmericasNLP/americasnlp2021/blob/main/data/bribri-spanish/orthographic-conversion.csv
Ashaninka cni	train	https://github.com/hinantin/AshaninkaMT (Ortega et al., 2020; Cushimariano Romano and Sebastián Q., 2008; Mihas, 2011)
	bibles	*cni-x-bible-cni-v1*
	mono	ShaShiYaYi (Bustamante et al., 2020): https://github.com/iapucp/multilingual-data-peru
Guarani gn	train	(Chiruzzo et al., 2020)
	extra*	PYconst: http://ej.org.py/principal/constitucion-nacional-en-guarani/
	bibles	*gug-x-bible-gug-v1*
	mono	Wikipedia crawls (Tiedemann, 2020)
Wixarika hch	train	https://github.com/pywirrarika/wixarikacorpora (Mager et al., 2018)
	extra	MXconst: https://constitucionenlenguas.inali.gob.mx/
	bibles	*hch-x-bible-hch-v1*
	mono	https://github.com/pywirrarika/wixarikacorpora (Mager et al., 2018)
	norm	https://github.com/pywirrarika/wixnlp/blob/master/normwix.py (Mager Hois et al., 2016)
Nahuatl nah	train	Axolotl (Gutierrez-Vasques et al., 2016)
	extra	MXConst: https://constitucionenlenguas.inali.gob.mx/
	bibles	*nch-x-bible-nch-v1, ngu-x-bible-ngu-v1, nhe-x-bible-nhe-v1, nhw-x-bible-nhw-v1*
	mono	Wikipedia crawls (Tiedemann, 2020)

Table 3: Data used for training (1). *train* refers to the official training data provided by the organizers, whereas *extra* refers to additional parallel non-Bible data. Corpora marked with *extra** are available on our repository but were not used in the translation experiments.

Hnähñu oto	train	Tsunkua: https://tsunkua.elotl.mx/about/
	extra	MXConst: https://constitucionenlenguas.inali.gob.mx/
	bibles	*ote-x-bible-ote-v1*
	mono	JW300 (Tiedemann, 2012; Agić and Vulić, 2019)
Quechua quy	train	JW300 (quy+quz) (Agić and Vulić, 2019)
		MINEDU + dict_misc: https://github.com/AmericasNLP/americasnlp2021/tree/main/data/quechua-spanish
	extra	Tatoeba (Tiedemann, 2012)
		BOconst: https://www.kas.de/documents/252038/253252/7_dokument_dok_pdf_33453_4.pdf/9e3dfb1f-0e05-523f-5352-d2f9a44a21de?version=1.0&t=1539656169513
		PEconst: https://www.wipo.int/edocs/lexdocs/laws/qu/pe/pe035qu.pdf
	bibles	*quy-x-bible-quy-v1, quz-x-bible-quz-v1*
	mono	Wikipedia crawls (Tiedemann, 2020)
Shipibo-Konibo shp	train	(Galarreta et al., 2017; Montoya et al., 2019)
	extra	Educational and Religious from http://chana.inf.pucp.edu.pe/resources/parallel-corpus/
	extra*	LeyArtesano: https://cdn.www.gob.pe/uploads/document/file/579690/Ley_Artesano_Shipibo_Konibo_baja__1_.pdf
	bibles	*shp-SHPTBL*
	mono	ShaShiYaYi (Bustamante et al., 2020): https://github.com/iapucp/multilingual-data-peru
Raramuri tar	train	(Brambila, 1976)
	extra	MXConst: https://constitucionenlenguas.inali.gob.mx/
	bibles	*tac-x-bible-tac-v1*
	norm	https://github.com/AmericasNLP/americasnlp2021/pull/5
Spanish	bibles	*spa-x-bible-americas, spa-x-bible-hablahoi-latina, spa-x-bible-lapalabra, spa-x-bible-newworld, spa-x-bible-nuevadehoi, spa-x-bible-nuevaviviente, spa-x-bible-nuevointernacional, spa-x-bible-reinavaleracontemporanea*

Table 4: Data used for training (2). *train* refers to the official training data provided by the organizers, whereas *extra* refers to additional parallel non-Bible data. Corpora marked with *extra** are available on our repository but were not used in the translation experiments.