NRC-CNRC Machine Translation Systems for the 2021 AmericasNLP Shared Task

Rebecca Knowles and **Darlene Stewart** and **Samuel Larkin** and **Patrick Littell**
National Research Council Canada
{Rebecca.Knowles, Darlene.Stewart, Samuel.Larkin, Patrick.Littell}@nrc-cnrc.gc.ca

Abstract

We describe the NRC-CNRC systems submitted to the AmericasNLP shared task on machine translation. We submitted systems translating from Spanish into Wixárika, Nahuatl, Rarámuri, and Guaraní. Our best neural machine translation systems used multilingual pretraining, ensembling, finetuning, training on parts of the development data, and subword regularization. We also submitted translation memory systems as a strong baseline.

1 Introduction

This paper describes experiments on translation from Spanish into Wixárika, Nahuatl, Rarámuri, and Guaraní, as part of the First Workshop on Natural Language Processing (NLP) for Indigenous Languages of the Americas (AmericasNLP) 2021 Shared Task on open-ended machine translation. Our approach to this task was to explore the application of simple, known methods of performing neural machine translation (NMT) for low-resource languages to a subset of the task languages. Our initial experiments were primarily focused on the following questions: *(1)* How well does multilingual NMT work in these very low resource settings? *(2)* Is it better to build multilingual NMT systems using only closely-related languages or does it help to add data from additional languages? *(3)* Is applying subword regularization helpful?

As we progressed through the task, it raised questions regarding domain and about use cases for low-resource machine translation. The approaches that we used for this task are not entirely language-agnostic; they might be more appropriately characterized as "language naïve" in that we applied some simple language-specific pre- and post-processing, but did not incorporate any tools that required in-depth knowledge of the language.

We submitted four systems, including ensembles, single systems, and a translation memory baseline. Our best system (S.0) consisted of an ensemble of systems incorporating multilingual training and finetuning (including on development data as pseudo-in-domain data).

Language	Family	Train	Dev
Nahuatl	Uto-Aztecan	16145	672
Rarámuri	Uto-Aztecan	14720	995
Wixárika	Uto-Aztecan	8966	994
Guaraní	Tupian	26032	995

Table 1: Language, language family, and number of lines of training and development data.

2 Data and Preprocessing

The shared task provided data for 10 language pairs, all with the goal of translating from Spanish. We chose to start with Wixárika (hch; Mager et al., 2018), Nahuatl (nah; Gutierrez-Vasques et al., 2016), and Rarámuri (tar; Brambila, 1976) as our main three languages of interest, all of which are languages in the Uto-Aztecan family indigenous to Mexico. We added Guaraní (gn; Chiruzzo et al., 2020) as an unrelated language (as spoken in Paraguay), to explore building multilingual NMT systems within and across language families. Ebrahimi et al. (2021) describes work on collecting development and test sets for the languages in the shared task. The datasets vary in size, dialect and orthographic variation/consistency, and level of domain match to the development and test data. Due to space considerations, we direct readers to the task page and the dataset information page for more information on the languages and on the datasets provided for the task.[1]

Given the size of the data (Table 1), additional data collection (particularly of data in the domain of interest) is likely one of the most effective ways to improve machine translation quality. However,

[1] Task page: http://turing.iimas.unam.mx/americasnlp/, Dataset descriptions: https://github.com/AmericasNLP/americasnlp2021/blob/main/data/information_datasets.pdf

noting both ethical (Lewis et al., 2020) and quality (Caswell et al., 2021) concerns when it comes to collecting or using data for Indigenous languages without community collaboration, we limited our experiments to data provided for the shared task.

2.1 Preprocessing and Postprocessing

We used standard preprocessing scripts from Moses (Koehn et al., 2007): `clean-corpus-n.perl` (on training data only), `normalize-punctuation.perl`, and `tokenizer.perl` (applied to all text, regardless of whether it already appeared tokenized).[2] The only language-specific preprocessing we performed was to replace "+" with an alternative character (reverted in postprocessing) for Wixárika text to prevent the tokenizer from oversegmenting the text. We note that the `13a` tokenizer used by `sacrebleu` (Post, 2018) tokenizes "+", meaning that scores that incorporate word n-grams, like BLEU (Papineni et al., 2002), are artificially inflated for Wixárika.

We detokenize (after unBPEing) the text and perform a small amount of language-specific postprocessing, which we found to have minimal effect on CHRF (Popović, 2015) and some effect on BLEU on development data.

2.2 BPE and BPE-Dropout

Following (Ding et al., 2019), we sweep a range of byte-pair encoding (BPE; Sennrich et al., 2016) vocabulary sizes: 500, 1000, 2000, 4000, and 8000 merges (we do not go beyond this, because of sparsity/data size concerns, though some results suggest we should consider larger sizes).

For each language pair or multilingual grouping, we learned a BPE model jointly from the concatenation of the source and target sides of the parallel data using `subword-nmt` (Sennrich et al., 2016), and then extracted separate source- and target-side vocabularies. We then applied the joint BPE model, filtered by the source or target vocabulary, to the corresponding data.

We apply BPE-dropout (Provilkov et al., 2020) in part to assist with data sparsity and in part because it may be an effective way of handing orthographic variation (as a generalization of the spelling errors that it helps systems become more robust to). Usually, BPE-dropout would be performed during training as mini-batches are generated, but we opted to generate 10 BPE-dropout versions of the training corpus using a dropout rate of 0.1 as part of our preprocessing. We then simply concatenate all 10 alternate versions to form the training corpus.

3 Models and Experiments

We report CHRF (Popović, 2015) scores computed with `sacrebleu` (Post, 2018).

3.1 Models

We trained Transformer (Vaswani et al., 2017) models using Sockeye-1.18.115 (Hieber et al., 2018) and cuda-10.1. We used the default value of 6 encoder/decoder layers, 8 attention heads, the Adam (Kingma and Ba, 2015) optimizer, label smoothing of 0.1, a cross-entropy loss, a model size of 512 units with a FFN size of 2048, and the vocabulary was not shared. We performed early stopping after 32 checkpoints without improvement. We chose custom checkpoint intervals of approximately two checkpoints per epoch. We optimized for CHRF instead of BLEU and used the whole validation set during validation. The batch size was set to 8192 tokens, and the maximum sequence length for both source and target was set to 200 tokens. We did not use weight tying, but we set gradient clipping to absolute and lowered the initial learning rate to 0.0001.

We performed preliminary experiments decreasing the number of encoder and decoder layers in our bilingual systems to 3 each, but did not observe improvements. Nevertheless, a wider search of architecture parameters, as in Araabi and Monz (2020), could yield improvements. After submission, we performed some additional experiments, building multilingual models with a range of numbers of decoder heads (1, 2, 4, 8), finding that a smaller number of decoder heads (e.g., 2) may be a promising avenue to explore in future work. Other approaches from Araabi and Monz (2020) also appear to show promise in our preliminary post-submission experiments, including a 4 layer encoder with a 6 layer decoder and changing layer normalization from pre to post, demonstrating that there are additional ways to improve upon our submitted systems.

3.2 MT Baselines

For each of the four language pairs, we build baseline systems translating out of Spanish. The best baseline systems with their respective BPE sizes

[2] See Appendix B for details.

System	gn	hch	nah	tar
Official (Organizer) Baseline	0.220	0.126	0.182	0.046
Baseline	0.222 (4k)	0.201 (2k)	0.201 (1k)	0.141 (2k)
+ Dropout	0.238 (8k)	0.226 (8k)	0.216 (4k)	0.127 (2k)
Multilingual-3	–	0.183 (4k)	0.203 (2k)	0.122 (4k)
Multilingual-4	0.222 (2k)	0.209 (4k)	0.213 (4k)	0.127 (8k)
+ Dropout	0.247 (8k)	0.226 (2k)	0.243 (4k)	0.142 (1k)
Multi.-4 + Dropout; Language Finetune (no dr.)	0.251 (8k)	**0.265** (2k)	0.250 (4k)	**0.149** (8k)
Multi.-4 + Dropout; Language Finetune	**0.258** (8k)	0.262 (2k)	**0.252** (2k)	0.134 (4k)

Table 2: System scores (CHRF) on the development set. Vocabulary size in parentheses.

are shown in Table 2. All of our baseline CHRF scores are higher than the official baselines released during the shared task,[3] likely due in part to more consistent tokenization between training and development/test (see Appendix C for additional discussion of training and development/test mismatch). For all languages except Rarámuri, adding BPE-dropout improved performance.

3.3 Multilingual Systems

Both Johnson et al. (2017) and Rikters et al. (2018) train multilingual systems by prepending a special token at the start of the source sentence to indicate the language into which the text should be translated. For example, the token "<nah>" prepended (space-separated) to a Spanish source sentence indicates that the text should be translated into Nahuatl. To train such a model, we concatenate all training data after adding these special tokens; the development data is similarly the concatenation of all development data. We do not perform any upsampling or downsampling to even out the distribution of languages in our training or development data (rather, we rely on language finetuning, as described in Section 3.4 to improve translation quality).

One of our initial questions was whether language relatedness mattered for building multilingual systems, so we first built a three-language (Wixárika, Nahuatl, Rarámuri) model, *Multiligual-3*, and then built a four-language (Guaraní, Wixárika, Nahuatl, Rarámuri) model, *Multilingual-4*. The Multilingual-4 system had consistently higher scores for all languages than the Multilingual-3 system, so we moved forward with experiments on Multilingual-4. Adding BPE-dropout to Multilingual-4 appeared to improve performance for all languages, but in the case of Wixárika (the language with the smallest amount of data), it was nearly identical to the baseline. Within the scope of this paper, we do not experiment with a wider range of languages (i.e., the remaining 6 languages), though it would not be surprising to find that additional language resources might also be beneficial.

Lang.	1k	2k	4k	8k
gn	889	1737	3299	5936
hch	516	728	1006	1389
nah	817	1502	2513	4033
tar	529	762	1072	1500

Table 3: Number of unique subwords in each language's training corpus (target side) for 1k, 2k, 4k, and 8k BPE merges in a Multilingual-4 scenario.

For the Multilingual-3 and Multilingual-4 models, the vocabulary is trained and extracted from the respective concatenated training corpus, so the target vocabulary is shared by all target languages as a single embedding matrix. Where languages share subwords, these are shared in the vocabulary (i.e., the language-specific tags are applied at the sentence level, not at the token level). The consequence of this is that each particular target language may not use the full multilingual vocabulary; we expect the system to learn which vocabulary items to associate (or not associate) with each language. For example, with a vocabulary produced through 8k merges, the full Multilingual-4 target side training corpus contains 7431 unique subwords, but the language-specific subcorpora that combine to make it only use subsets of that: Guaraní training data contains 5936 unique subwords, while Wixárika contains only 1389 (the overlap between Guaraní and Wixárika subwords is 1089 subwords). Table 3 shows the number of unique subwords in the target language training corpus for the Multilingual-4 setting. Our systems are free to generate any subword from the full combined vocabulary of target subwords since there is no explicit restriction during decoding. Thus, in some cases, our multilingual systems do generate subwords that were not seen in a specific language's training data vocabulary sub-

[3] https://github.com/AmericasNLP/americasnlp2021/tree/main/baseline_system

set; while some of these *could* result in translation errors, a preliminary qualitative analysis suggests that many of them may be either source language words (being copied) or numerical tokens, both of which point to potential benefits of having used the larger concatenated multilingual corpus.

3.4 Language Finetuning

We can then finetune[4] the multilingual models to be language-specific models.[5] The intuition here is that the multilingual model may be able to encode useful information about the source language, terms that should be copied (e.g., names/numbers), target grammar, or other useful topics, and can then be specialized for a specific language, while still retaining the most relevant or most general things learned from all languages trained on. We do this finetuning based on continued training on each language's training data, with that language's development data, building a new child system for each language based on the parent Multilingual-4 system (with or without dropout).[6] When we do this, we no longer use the language-specific tags used during multilingual model training.

Language finetuning appears to produce improvements, with some performing better with dropout and some better without, as seen in the final two lines of Table 2. Rarámuri appears to have a drop in performance after language finetuning with dropout. However, all Rarámuri scores are extremely low; it is likely that many of the decisions we make on Rarámuri do not represent real improvements or performance drops, but rather noise, so we have very low confidence in the generalizability of the choices (Mathur et al., 2020).

3.5 Development Finetuning

Noting that the development data was of a different domain, and sometimes even a different dialect or orthography than the training data, we followed an approach used in Knowles et al. (2020): we divided the development set (in this case in half), performing finetuning with half of it and using the remainder for early stopping (and evaluation). We acknowledge that, given the very small sizes of the development sets, minor differences we observe are likely to be noise rather than true improvements (or true drops in performance); while we made choices about what systems to submit based on those, we urge caution in generalizing these results or drawing strong conclusions.

We show performance of models finetuned on the first half of the development set (performance measured on the second half of the development set), both with and without first finetuning for language, in Table 4. We also compare these against the best systems we trained without training on development data, as well as with the translation memory approach (Section 4.3).

4 Submitted Systems

4.1 Systems with Dev. (S.0, S.2, and S.4)

We submitted single systems (not ensembled) that were trained using the first half of the development set (labeled S.2 in submission). They were selected based on highest scores on the second half of the development set (see Table 4 for scores and vocabulary sizes). For Guaraní, Wixárika, and Nahuatl, we selected systems of the type Multi.-4 + BPE Dr.; Lang. finetuning; 1/2 Dev. finetuning. For Rarámuri, we selected a system with only 1/2 dev. finetuning (Multi.-4 + BPE Dr.; 1/2 Dev. Ft.).

Our best systems were ensembles (labeled S.0 in submission) of the systems described above and their corresponding system trained with the second half of the development set. For Guaraní, we also submitted an ensemble of four systems; the two *Multi.-4 + BPE Dr.; Lang. finetuning; 1/2 Dev finetuning* systems and the two *Multi.-4 + BPE Dr.; 1/2 Dev Ft.* systems (S.4). It performed similarly to the two-system ensemble.

4.2 Systems without Dev. (S.1)

We also submitted systems that were not trained on development data. For these, we were able to select the best system from our experiments, based on its CHRF score on the full development set. For Guaraní and Nahuatl, these were *Multi.4 + BPE Dr.; Lang. ft.* systems, for Rarámuri it was the *Multi.4 + BPE Dr.; Lang. ft. (no dr.)* system, and for Wixárika it was an ensemble of the two.

4.3 Translation Memory (S.3)

Noting the very low automatic metric scores across languages and without target language expertise to

[4] In our tables, we use the following notation to indicate finetuning: "[parent model]; [child finetuning]" and this notation stacks, such that "X; Y; Z" indicates a parent model X, finetuned as Y, and then subsequently finetuned as Z.

[5] We note that all finetuning experiments reported in this paper used BPE-dropout unless otherwise noted.

[6] We note that some catastrophic forgetting may occur during this process; it may be worth considering modifying the learning rate for finetuning, but we leave this to future work.

System	gn	hch	nah	tar
Multi.-4 + Dropout	0.249 (8k)	0.228 (2k)	0.247 (8k)	0.145 (1k)
Multi.-4 + Dr.; Lang. Finetune	0.260 (8k)	0.261 (2k)	0.252 (2k)	0.137 (500)
Multi.-4 + Dr.; 1/2 Dev. Finetune	0.331 (4k)	0.367 (4k)	0.368 (8k)	**0.289** (4k)
Multi.-4 + Dr.; Lang. Finetune; 1/2 Dev. Ft.	**0.338** (4k)	**0.368** (8k)	**0.376** (8k)	0.280 (2k)
S.1 (no dev)	0.260 (8k)	0.266 (2k)	0.252 (2k)	0.150 (8k)
S.2 (1/2 dev, single system)	**0.338** (4k)	**0.368** (8k)	**0.376** (8k)	**0.289** (4k)
Translation Memory	0.257 (na)	0.273 (na)	0.285 (na)	0.246 (na)

Table 4: System scores on the second half of the development set.

System	gn	hch	nah	tar
S.0	0.304	0.327	0.277	0.247
S.4	0.303	–	–	–
S.2	0.288	0.315	0.273	0.239
S.3/TM	0.163	0.200	0.181	0.165
S.1/no dev	0.261	0.264	0.237	0.143
Helsinki 2	0.376	0.360	0.301	0.258

Table 5: Submitted systems scores (CHRF) on test data. Final row shows best overall submitted system for each language, Helsinki submission 2.

determine if the output is fluent but not adequate, adequate but not fluent, or neither fluent nor adequate, we decided to build a translation memory submission. In computer aided translation (CAT), a "translation memory" (TM) is a database of prior source-target translation pairs produced by human translators. It can be used in CAT as follows: when a new sentence arrives to be translated, the system finds the closest source-language "fuzzy match" (typically a proprietary measure that determines similarity; could be as simple as Levenshtein distance) and returns its translation (possibly with annotations about the areas where the sentences differed) to the translator for them to "post-edit" (modify until it is a valid translation of the new sentence to be translated).

With the understanding that the development and test sets are closer to one another in terms of domain and dialect than they are to the training data, we treat the development set as a TM. Following Simard and Fujita (2012), we use an MT evaluation metric (CHRF) as the similarity score between the test source sentences and the TM source sentences, with the translation of the closest source development set sentence as the output.[7]

We validated this approach on the two halves of the development set (using the first half as a TM for the second half and vice versa). On half the development set, for all languages except for Guaraní, the TM outperformed the system trained without any development data (S.1), highlighting the differences between the training and development/test data (Table 4), particularly striking because the TM used for these experiments consisted of only half the development set (<500 lines) as compared to the full training set.[8] On the test set, only the Rarámuri TM outperformed the best of our MT systems built without training on development.

5 Results

Our results consistently placed our submissions as the second-ranking team (behind Helsinki's top 2-3 submissions) in the with-development-set group, and second or third ranking team (2nd, 3rd, or 4th submission) within the no-development-set cluster as measured by CHRF. For Wixárika and Rarámuri particularly, our TM submission proved to be a surprisingly strong baseline.

We note that CHRF and BLEU are not strictly correlated, and for all languages, scores are low. This raises questions about goals, metrics, and use cases for very low resource machine translation. We provide a short discussion of this in Appendix A. It will require future work and human evaluation to determine whether such systems are useful or harmful in downstream tasks.

Acknowledgements

We thank the translators who provided their expertise and time to make this work possible:[9] Giovany Martinez Sebastián, Pedro Kapoltitan, José Antonio (Nahuatl), Silvino González de la Crúz (Wixárika), Perla Alvarez Britez (Guaraní), and María del Cármen Sotelo Holguín (Rarámuri). We thank the anonymous reviewers and our colleagues Michel Simard, Gabriel Bernier-Colborne, and Chi-kiu Lo for their comments and suggestions.

[7] In the event of a tie, we chose the first translation.

[8] See Appendix C for additional detail on vocabulary coverage between training, development, and test data.

[9] Full list for all languages available here: https://github.com/AmericasNLP/americasnlp2021/blob/main/data/information_datasets.pdf

References

Ali Araabi and Christof Monz. 2020. Optimizing transformer for low-resource neural machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3429–3435, Barcelona, Spain (Online). International Committee on Computational Linguistics.

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

Isaac Caswell, Julia Kreutzer, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Javier Ortiz Suárez, Iroro Orife, Kelechi Ogueji, Rubungo Andre Niyongabo, Toan Q. Nguyen, Mathias Müller, André Müller, Shamsuddeen Hassan Muhammad, Nanda Muhammad, Ayanda Mnyakeni, Jamshidbek Mirzakhalov, Tapiwanashe Matangira, Colin Leong, Nze Lawson, Sneha Kudugunta, Yacine Jernite, Mathias Jenny, Orhan Firat, Bonaventure F. P. Dossou, Sakhile Dlamini, Nisansa de Silva, Sakine Çabuk Ballı, Stella Biderman, Alessia Battisti, Ahmed Baruwa, Ankur Bapna, Pallavi Baljekar, Israel Abebe Azime, Ayodele Awokoya, Duygu Ataman, Orevaoghene Ahia, Oghenefego Ahia, Sweta Agrawal, and Mofetoluwa Adeyemi. 2021. Quality at a glance: An audit of web-crawled multilingual datasets.

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.

Shuoyang Ding, Adithya Renduchintala, and Kevin Duh. 2019. A call for prudent choice of subword merge operations in neural machine translation. In *Proceedings of Machine Translation Summit XVII Volume 1: Research Track*, pages 204–213, Dublin, Ireland. European Association for Machine Translation.

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.

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).

Felix Hieber, Tobias Domhan, Michael Denkowski, David Vilar, Artem Sokolov, Ann Clifton, and Matt Post. 2018. The sockeye neural machine translation toolkit at AMTA 2018. In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 200–207, Boston, MA. Association for Machine Translation in the Americas.

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.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Rebecca Knowles, Darlene Stewart, Samuel Larkin, and Patrick Littell. 2020. NRC systems for the 2020 Inuktitut-English news translation task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 156–170, Online. Association for Computational Linguistics.

Philipp Koehn and Ulrich Germann. 2014. The impact of machine translation quality on human post-editing. In *Proceedings of the EACL 2014 Workshop on Humans and Computer-assisted Translation*, pages 38–46, Gothenburg, Sweden. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Jason Edward Lewis, Angie Abdilla, Noelani Arista, Kaipulaumakaniolono Baker, Scott Benesiinaabandan, Michelle Brown, Melanie Cheung, Meredith Coleman, Ashley Cordes, Joel Davison, Kūpono Duncan, Sergio Garzon, D. Fox Harrell, Peter-Lucas Jones, Kekuhi Kealiikanakaoleohaililani, Megan Kelleher, Suzanne Kite, Olin Lagon, Jason Leigh, Maroussia Levesque, Keoni Mahelona, Caleb Moses, Isaac ('Ika'aka) Nahuewai, Kari Noe, Danielle Olson, 'Ōiwi Parker Jones, Caroline Running Wolf, Michael Running Wolf, Marlee Silva, Skawennati Fragnito, and Hēmi Whaanga. 2020. Indigenous protocol and artificial intelligence position paper. Project Report 10.11573/spectrum.library.concordia.ca.00986506, Aboriginal Territories in Cyberspace, Honolulu, HI. Edited by Jason Edward Lewis.

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.

Nitika Mathur, Timothy Baldwin, and Trevor Cohn. 2020. Tangled up in BLEU: Reevaluating the evaluation of automatic machine translation evaluation metrics. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4984–4997, Online. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. 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, Brussels, Belgium. Association for Computational Linguistics.

Ivan Provilkov, Dmitrii Emelianenko, and Elena Voita. 2020. BPE-dropout: Simple and effective subword regularization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1882–1892, Online. Association for Computational Linguistics.

Matīss Rikters, Mārcis Pinnis, and Rihards Krišlauks. 2018. Training and adapting multilingual NMT for less-resourced and morphologically rich languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Michel Simard and Atsushi Fujita. 2012. A poor man's translation memory using machine translation evaluation metrics. In *Proceedings of the 10th Bienniall Conference of the Association for Machine Translation in the Americas*. Association for Machine Translation in the Americas.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5998–6008.

A When does it make sense to build MT systems?

Our recent participation in shared tasks has made us consider scenarios and use cases for low-resource MT, which we discuss in this appendix.

At the WMT 2020 News translation task, the Inuktitut-English translation task was arguably mid-resource (over a million lines of parallel legislative text), with the Hansard (legislative assembly) portion of the development and test set being a strong domain match to the training data. The news data in the development and test sets represented a domain mismatch.

In the supervised low-resource task at WMT, there was an arguably low-resource (approximately 60,000 lines of parallel text) language pair of German-Upper Sorbian. However, the test set was extremely well-matched to the training data (though not exact duplicates), resulting in surprisingly high automatic metric scores (BLEU scores in the 50s and 60s).

In this AmericasNLP shared task, we observed perhaps the hardest scenario (outside of zero-shot): low resource with domain/dialect/orthographic mismatch. It should come as no surprise, then, that we observe extremely low automatic metric scores for this task.

	Domain Match	**Mismatch**
Low-Res.	Upper Sorbian	AmericasNLP
Mid-Res.	Inuktitut Hansard	Inuktitut News

Table 6: Comparison of recent shared tasks on low-resource machine translation.

For both the Inuktitut and Upper Sorbian systems, we know of community and/or government organizations that may be interested in using machine translation technology, for example as part of a computer aided translation (CAT) tool.[10] Provided that human evaluation found the quality level of the machine translation output appropriately high (no human evaluation was performed in the Upper Sorbian task, and the Inuktitut human evaluation is ongoing), there appear to be clear suitable use cases here, such as as part of a human translation workflow translating the Hansard as it is produced or translating more of the same domain Upper Sorbian/German text. It is less clear, where there is a domain mismatch, whether the quality is anywhere near high enough for use in a CAT setting. We know that the usefulness of machine translation in CAT tools varies by translator (Koehn and Germann, 2014); some find even relatively low-quality translations useful, while others benefit only from very high-quality translations, and so on. There are also potential concerns that MT may influence the way translators choose to translate text.

But what about this low-resource, domain mismatch setting? While human evaluation would be the real test, we suspect that the output quality may be too low to be beneficial to most translators. As a brief example, we consider the CHRF scores that were generated between two Spanish sentences as a byproduct of the creation of our translation memory submission.

- Washington ha perdi**do todos los** partidos. (Washington has lost all the games.)
- Continuaron visitan**do todos los** días. (They continued visiting every day.)

In part on the basis of the 10-character (spaces ignored) substring "do todos los" (for which "todos los" can be glossed as "every", but the string-initial "do" suffix belongs to two different verbs, one of which is in its past participle form and the other of which is in its present participle form), these sentences have a score of 0.366 CHRF (if we consider the first to be the "system" output and the second to be the "reference").

Here of course both sentences are grammatical, but they are clearly not semantic equivalents. Nevertheless, comparing the two produces a CHRF score comparable to the the highest scores observed in this task.[11] We argue then, that if the goal is CAT, then it may be better to consider a TM-based approach, even though it has lower scores, given that CAT tools are well-equipped to handle TMs, and typically provide some sort of indication about the differences between the sentence to be translated and its fuzzy-match from the TM as a guide for the translator. In an MT-based approach, the translator may be confronted with fluent text that is not semantically related to the source, ungrammatical language, or types of other problematic output.

[10] For example, the presentation of the Upper Sorbian-German machine translation tool *sotra* (https://soblex.de/sotra/) encourages users to proofread and correct the output where necessary: https://www.powtoon.com/online-presentation/cr2llmDWRR9/

[11] We acknowledge that this is an imperfect comparison, since the scores in this task are of course not on Spanish output and thus should not be compared directly.

If the goal of these MT tools is *not* CAT, but rather for a reader to access text in their preferred language, we expect that neither the MT systems nor the TMs would provide the kind of quality that users of online MT systems have come to expect. This raises questions of how to alert potential users to the potential for low-quality MT.

It is possible that there may be other use cases, in which case a downstream evaluation may be more appropriate than automatic metrics.

B Pre- and Post-processing Details

Training corpora (but not development or test corpora) were processed using the Moses `clean-corpus-n.perl` script (Koehn et al., 2007), with a sentence length ratio of 15:1 and minimum and maximum lengths of 1 and 200, respectively. All corpora were preprocessed with the `normalize-punctuation.perl` script, with the language set to Spanish (since no language-specific rules are available for the other languages in this task), and all instances of U+FEFF ZERO WIDTH NO-BREAK SPACE were removed. The only additional language-specific preprocessing that we performed was to replace "+" with U+0268 LATIN SMALL LETTER I WITH STROKE in the Wixárika text; this prevents the text from being oversegmented by the tokenizer, and is reverted in postprocessing.[12] We note that it might be desirable to perform a similar replacement of apostrophes with a modifier letter apostrophe, but because some of the training data was released in tokenized format we were not confident that we could guarantee consistency in such an approach.[13]

All text is then tokenized with the Moses tokenizer `tokenizer.perl`, with aggressive hyphen splitting, language set to Spanish, and no HTML escaping.[14] Note that we apply the tokenization even to already-tokenized training data, in the hopes of making the different datasets as consistent as possible.

Postprocessing consists of unBPEing then detokenizing using Moses' `detokenizer.perl`. An extra step is needed for Wixárika to revert back to the "+" character. We also perform a small amount of extra language-specific postprocessing, which has limited effects on CHRF (it primarily involves tokenization) with some effect on BLEU. For example, for Guaraní, we delete spaces around apostrophes and replace sequences of three periods with U+2026 HORIZONTAL ELLIPSIS. For Wixárika, we add a space after the "¿" and "¡" characters. For Nahuatl, we make sure that "$" is separated from alphabetic characters by a space. For Rarámuri, we replace three periods with the horizontal ellipsis, convert single apostrophes or straight quotation marks before "u" or "U" to U+2018 LEFT SINGLE QUOTATION MARK and remove the space between it and the letter, and then convert any remaining apostrophes or single straight quotes to U+2019 RIGHT SINGLE QUOTATION MARK as well as removing any surrounding spaces. These are all heuristics based on frequencies of those characters in the development data, and we note that their effect on BLEU scores and CHRF scores is minimal (as measured on development data).

C Coverage

The Wixárika and Guaraní data was provided untokenized, but Nahuatl and Rarámuri datasets contained training data that was tokenized while the development and test data was untokenized. Here we briefly illustrate the impact of the mismatch, through token and type coverage. In Table 7, we show what percentage of target language development tokens (and types) were also observed in the training data, before and after applying tokenization. Table 8 shows the same for source language. Table 9 shows source coverage for the test data instead of the development data. Finally, Table 10 shows what percentage of the source test data is contained in the *development set*. Unsurprisingly, coverage is higher across the board for Spanish (source), which is less morphologically complex than the target languages. Spanish-Rarámuri has the lowest coverage in both source and target. Spanish-Nahuatl has the second-highest coverage on the source side, but not on the target side, perhaps due to the historical content in the training data and/or the orthographic conversions applied. Spanish-Guaraní has the highest coverage on both source and target.

Applying BPE results in approximately 100% coverage, but it is still worth noting the low full-word coverage, as novel vocabulary may be hard

[12] Note, however, that the `13a` tokenizer used by `sacrebleu` (Post, 2018) tokenizes "+", meaning that BLEU scores and other scores that incorporate word n-grams are artificially inflated for Wixárika.

[13] With CHRF as the main metric, this is less of a concern than it would be were the main metric BLEU or human evaluation. We note that even the use of CHRF++, with its use of word bigrams, would make this a concern.

[14] `tokenizer.perl -a -l es -no-escape`

	Tokens		Types	
	Raw	Tok.	Raw	Tok.
es-hch	54.4%	65.3%	27.5%	31.7%
es-nah	53.8%	63.9%	25.3%	30.0%
es-tar	32.7%	55.0%	8.1%	14.4%
es-gn	61.4%	81.1%	35.0%	46.4%

Table 7: Target language training data coverage on development set.

	Tokens		Types	
	Raw	Tok.	Raw	Tok.
es-hch	70.5%	78.4%	35.2%	43.7%
es-nah	77.8%	89.1%	51.2%	68.6%
es-tar	66.7%	76.7%	30.4%	41.3%
es-gn	84.5%	90.9%	62.0%	72.8%

Table 8: Source language (Spanish) training data coverage on development set (compared against training data).

for the systems to translate or to generate.

For all languages except Guaraní, the first half of the development set had higher target language coverage on the second half of the development set, as compared to training target language coverage on the full development set (or second half of the development set), which may explain both the improved performance of systems that trained on development data and the quality of the translation memory system.

	Tokens		Types	
	Raw	Tok.	Raw	Tok.
es-hch	74.8%	83.1%	42.0%	51.2%
es-nah	77.6%	89.3%	48.6%	68.0%
es-tar	69.2%	80.9%	34.0%	48.5%
es-gn	83.5%	90.8%	59.5%	71.3%

Table 9: Source language (Spanish) training data coverage on test set (compared against training data).

	Tokens		Types	
	Raw	Tok.	Raw	Tok.
es-hch	73.3%	81.0%	34.1%	40.3%
es-nah	69.8%	78.2%	27.7%	33.3%
es-tar	73.3%	81.0%	34.1%	40.3%
es-gn	73.3%	81.0%	34.1%	40.3%

Table 10: Source language (Spanish) *development data* coverage on test set. Note that Wixárika, Rarámuri, and Guaraní share identical source data for the development set, and all languages share identical source data for the test set.

	Tokens		Types	
	Raw	Tok.	Raw	Tok.
es-hch	72.3%	80.4%	38.3%	45.7%
es-nah	69.0%	77.3%	37.4%	43.8%
es-tar	73.1%	81.1%	37.8%	45.8%
es-gn	72.7%	80.2%	37.2%	44.0%

Table 11: Source language (Spanish) *first half of the development data* coverage on *second half of the development data*. I.e., for raw es-hch data, 72.3% of source language tokens in the second half of the development set appeared somewhere in the first half of the development set.

	Tokens		Types	
	Raw	Tok.	Raw	Tok.
es-hch	66.3%	74.8%	36.8%	41.4%
es-nah	59.1%	67.8%	33.0%	37.4%
es-tar	73.8%	85.1%	39.1%	46.8%
es-gn	56.7%	77.7%	31.9%	40.2%

Table 12: Target language *first half of the development data* coverage on *second half of the development data*. I.e., for raw es-hch data, 66.3% of target language tokens in the second half of the development set appeared somewhere in the first half of the development set.