Cross-lingual Transfer Learning for Intent Detection of Covid-19 Utterances

Abhinav Arora* Facebook Conversational AI

Akshat Shrivastava* Facebook Conversational AI

Mrinal Mohit Facebook Conversational AI

Lorena Sainz-Maza Lecanda Facebook Conversational AI

Ahmed Aly Facebook Conversational AI

Abstract

In times of a global pandemic, interactive chat bots are an indispensable tool to provide information to people. With this motivation, we study the problem of intent detection of user utterances, which is usually the first language understanding step in such systems. Specifically, we focus on cross-lingual transfer learning for intent detection of user utterances and zero-shot learning for code-switched (CS) utterances. We release a multilingual dataset, M-CID, containing 5271 utterances across English, Spanish, French and Spanglish (Spanish + English). We use this dataset to explore some cross-lingual transfer learning techniques to study: (1) monolingual and multilingual model baselines, (2) cross-lingual transfer from English to Spanish and French, and (3) zero-shot code-switching for Spanglish. In our experiments, we observe that XLM-R models are able to significantly outperform cross lingual word embedding techniques for all of the above settings. We also show that it is possible to obtain a strong performance on code-switched data by only using monolingual data from substrate languages.

1 Introduction

In the wake of the Covid-19 crisis, it is of paramount importance to build interactive tools that can provide essential information such as Covid symptoms, treatment options, etc. These could either be information retrieval systems that fetch relevant articles (Zhang et al., 2020; Esteva et al., 2020; MacAvaney et al., 2020) or they could be interactive chat bots (WHO, 2020; Martin et al., 2020) that users can interact with. In this work, we explore the problem of intent classification; which is the first step of a natural language understanding system. For example, for an utterance such as *What are the indicators of covid infection?*, the first step in responding to this request, is to identify that the user's intent is to ask for Covid-19 symptoms.

While neural models dominate intent prediction (Liu and Lane, 2016; Zhang and Wang, 2016; Zhang et al., 2018) they require a lot of training data. Given this requirement, developing these systems for many new languages can be a highly resource-intensive task, especially during global pandemic situations, where new languages support is needed in a very short amount of time. Furthermore, multilingual systems often also need to support code-switching (CS), which is the alternation of languages within an utterance (Poplack, 2004). Collecting CS data is even harder as it requires bilingual annotators and the number of CS pairs grows quadratically with languages. Thus, there is a need to explore techniques that enable transfer learning from one or more languages to other languages and CS dialects.

In order to further study multilingual intent detection for Covid-19, **we release *M-CID (Multiingual Covid Intent Detection)***, an open source intent detection dataset for Covid-19 chat bots. M-CID contains 5271 utterances across 16 intents, with a provided train, eval, and test split across 3 languages: English, Spanish, and French along with a Spanglish test set for CS. **We provide several strong baselines** to show the impact of cross lingual embedding such as MUSE (Conneau et al., 2017), SentencePiece embeddings from XLM-R (Kudo and Richardson, 2018; Conneau et al., 2020), aligning ELMo representations (Peters et al., 2018; Schuster et al., 2019) and pre-trained multilingual transformers, XLM-R (Conneau et al., 2020), comparing monolingual training against cross lingual training. On our dataset, we show that XLM-R models significantly out perform cross lingual embeddings and **cross lingual training improves performance across most models compared to**

* Correspondence to {abhinavarora,akshats} @fb.com

	EN	ES	FR	Spanglish
Train	1258	1106	1105	0
Eval	148	161	173	0
Test	339	333	315	333
Total	1745	1600	1593	333

Table 1: Summary statistics of the dataset. Note that *Spanglish* only has a test set for zero-shot transfer evaluations.

monolingual training. In addition **we also show the impact of cross lingual transfer learning**, where we train with the full English train set and small portions of other languages, and also show strong zero-shot transfer with XLM-R based models. Lastly, we study the impact on our code switching test set, we show that **monolingual training on English and Spanish for XLM-R based models is sufficient for code switching**.

2 Data

We release M-CID, a dataset of **5271** natural language utterances across **16** Covid-19 specific intents and **3** languages: English, Spanish and French. In addition to these three languages, the dataset also contains a Spanglish test set for CS evaluation. All of these utterances were synthetically created by annotators based on an ontology describing all intents with few representative examples. No user data was used in this process. English, Spanish and French utterances were authored by native speakers of these languages using the described ontology. Spanglish utterances were created by one of the authors, who is bilingual in Spanish and English.

We believe that this data provides a great opportunity to explore cross-lingual classification for Covid-19 chat bots and to the best of our knowledge, this is the first multilingual dataset for an intent detection task for Covid-19 utterances. Table 1 contains the count of utterances for each language across the training, evaluation and test splits. More details about the intent labels, distribution of utterances across them, and some representative examples are presented in Appendix A.

We release the data at `https://fb.me/covid_mcid_dataset`.

3 Modeling Approaches

In the following section, we provide a brief description of all the models and the implementations used. We use accuracy as our evaluation metric, which works well for our setup because the intent labels

Model	Setting	Accuracy EN	Accuracy ES	Accuracy FR
MUSE	Mono	81.12	76.28	69.52
	XL	79.94	81.98	73.97
SP	Mono	82.89	79.58	73.97
	XL	83.48	78.76	75.56
ELMo	Mono	86.14	84.98	76.83
	XL	88.20	89.19	82.22
XLM-R Base	Mono	90.27	88.59	87.30
	XL	90.27	**91.89**	88.25
XLM-R Large	Mono	**91.45**	91.29	88.25
	XL	**91.45**	**93.09**	**88.89**

Table 2: Full training results for all languages. *Mono* refers to a monolingual model for each language and *XL* refers to a common multilingual model for all languages.

have a balanced distribution in the dataset. Appendix B contains details regarding reproducibility and model hyperparameters for further reference.

3.1 Cross-Lingual Word Embeddings

Our base model is a CNN based text classification model based on the architecture described by Kim (2014). For regularization, we add a dropout (Srivastava et al., 2014) after the convolution and pooling layers. In order to enable language transfer, we use pre-trained cross-lingual word embeddings as an input to the model. We experiment with the following embedding strategies:

- **MUSE**: We use MUSE word embeddings (Conneau et al., 2017), with a vocabulary size of 25K of for all the three languages. These are fastText (Bojanowski et al., 2017) Wikipedia supervised word embeddings, aligned in a single vector space. We refer to this model as simply MUSE.

- **SentencePiece Embeddings**: We experiment with pre-trained SentencePiece embeddings obtained from a large multilingual corpus. Specifically, we use the SentencePiece (Kudo and Richardson, 2018) tokenization and take the embedding values from the already-trained XLM-R (large) (Conneau et al., 2020) weights. Since these are sub-word embeddings, they tend to be robust to misspellings and rare tokens by breaking them down into better-known sub-tokens. We refer to this model as simply SP.

Model	Spanish % Training Zero-shot	10	20	50	80	French % Training Zero-shot	10	20	50	80
MUSE (F)	59.76	59.76	64.86	71.77	71.77	47.30	57.46	61.27	65.4	64.44
SP (F)	33.03	64.86	74.47	78.98	82.58	29.84	60.32	66.67	73.33	75.24
MUSE	25.83	50.15	56.46	72.07	75.68	24.76	47.30	55.56	60.32	65.71
SP	38.74	56.46	59.46	71.17	76.28	29.52	53.02	59.68	67.94	69.84
ELMo	71.17	74.47	82.58	83.48	85.29	63.17	66.03	70.48	75.87	76.83
XLM-R Base	84.98	88.29	89.49	91.29	**92.79**	78.73	**83.17**	82.86	**88.89**	88.57
XLM-R Large	**90.99**	**91.29**	**92.79**	**92.59**	**93.09**	**83.17**	**83.49**	**85.08**	87.62	**89.84**

Table 3: Results for cross-lingual transfer for all models. *(F)* refers to freezing the embeddings during training. In the zero-shot setting, only English data is used for training and model selection. For others, the specified percentage of target training data is also used along with English.

- **Cross-lingual ELMo**: We also experiment with aligned multi-lingual deep contextual embeddings obtained by aligning monolingual ELMo embeddings (Peters et al., 2018). We use the ELMo models and alignments released by Schuster et al. (2019). Specifically, we use the alignments of the first LSTM layer, which the authors found best in their experiments.

3.2 Pre-trained Cross-Lingual Language Models

Using the same accuracy metric as above, we also examine the performance of pre-trained XLM-R (Conneau et al., 2020) models. These models are pre-trained via an unsupervised Masked Language Modeling (MLM) objective (Devlin et al., 2019) on massive multilingual data. They share a SentencePiece representation and a common transformer encoder (Vaswani et al., 2017) for different languages. In order to use this for intent classification, we add a linear classifier on top of the first hidden state of the Transformer and fine-tune the network on our dataset. For our experiments, we report results with both XLM-R Base and XLM-R Large which are pre-trained on 100 languages and are provided by the PyText framework (Aly et al., 2018).

Results and Discussion Table 2 shows the test set accuracy for all of the above models using the full training data. In the *mono* setting a model is trained per language using the data of only that language. In the *XL* setting a single cross-lingual model is trained using the data for all the languages together. For these experiments, MUSE and SP embeddings were not frozen during training. While we get different results for each language, there are several consistent patterns. XLM-R models significantly outperform other models. We also see that cross-lingual models trained with all the 3 languages mostly do better than their monolingual counterparts, barring few exceptions. Amongst the cross-lingual embeddings, SP embeddings are generally better than MUSE, which is expected as they operate on subword units that are shared across languages. Contextual ELMo embeddings perform better than both of these due their contextual nature.

4 Cross-lingual Learning

4.1 Language Transfer

In this set of experiments, we examine the language transfer abilities of our models. Specifically, we treat English as our source language, and Spanish and French as the target languages. For each of the models discussed above, we first run zero-shot experiments where only English data is used for training and model selection. We then run learning curve experiments, where we progressively sample 10, 20, 50 and 80 percent of the target language training data and upsample it so that it roughly matches the size of the English data. Here, model selection is done using the evaluation splits of all languages.

Results and Discussions Table 3 shows the cross-lingual transfer results for all the models. From these results, it is evident that XLM-R large can achieve very strong performance for zero-shot transfer from English. For Spanish, the zero-shot performance is about two absolute points lesser than using 80% Spanish training data. For French, this gap is higher and there is a progressive improvement from zero-shot to 80% training. For both the languages, we see that having target language training data yields better performance than zero-shot. XLM-R base follows a similar trend as

Model	Setting		
	EN	ES	EN + ES
MUSE (F)	63.06	48.65	70.57
SP (F)	62.76	43.24	78.38
MUSE	69.67	42.94	76.88
SP	68.77	55.86	79.88
XLM-R Base	83.78	77.78	88.29
XLM-R Large	**87.39**	**91.29**	**88.89**

Table 4: Zero-shot code-switching results for each of the training settings. *(F)* refers to freezing the embeddings during training.

large. Interestingly, for French, XLM-R base has slightly better results with only 50% training data than 80%. We attribute this to the high sensitivity of XLM-R fine-tuning to the learning rate.

For MUSE and SP, we show results with both freezing and fine-tuning the embeddings during training. For MUSE, we find that freezing the word embeddings yields a significantly better performance compared to fine-tuning in the lower resource settings (<50%), as the model does not overfit to the source language. For SP, freezing the embeddings is almost always better than fine-tuning. This can be attributed to the overlap of subwords across languages. Similar to table 2, we observe better language transfer with SP as compared to MUSE. Similarly, contextual ELMo embeddings perform better than both of these. Compared to XLM-R, all of these approaches have a much bigger performance gap between zero-shot and 80% target language training. This suggests that XLM-R is very effective at zero-shot cross-lingual transfer, which aligns with the findings of Wu and Dredze (2019).

4.2 Zero-shot Code-Switching

Since code-switching is a big part of spoken language in many cultures, we also investigate the performance of our models on Spanglish, which is a mix of English and Spanish. These are zero-shot experiments where we neither use CS data for model training nor for model selection. The only data available is monolingual English and Spanish data. For each of our models discussed above, we experiment with three training data settings. We first train two models using the training data of each of the two languages one by one, and then a model using both Spanish and English data.

Results and Discussions Table 4 show the zero-hot CS performance of different models. We do not perform ELMo experiments for CS as it is not intuitive to represent Spanglish context with monolingual ELMo. From the results, we can see that XLM-R models perform very well even when fine-tuned on English only or Spanish only. XLM-R large fine-tuned on Spanish only, outperforms all other model settings. We also see that for MUSE and SP, training on English only gives better performance than Spanish only setting. We believe this is because for Spanglish utterances, the trigger words such as *treatment*, *vaccine*, *donation*, etc are usually in English and thus the English only model is able to do well. Further, freezing the embeddings is usually worse for all settings.

5 Related Work

Cross-lingual Transfer Learning Majority of the initial work on cross-lingual transfer was centered around aligning pre-trained word embeddings to a common vector space (Xing et al., 2015; Zhang et al., 2017; Conneau et al., 2017). Schuster et al. (2019) and Aldarmaki and Diab (2019) further build on this by exploring context-aware cross-lingual alignment of contextualized representations from ELMo (Peters et al., 2018). More recently, pre-trained multilingual masked language models such as mBERT (Devlin et al., 2019), XLM (Lample and Conneau, 2019) and XLM-R (Conneau et al., 2020) have been introduced. XLM-R obtains state-of-the-art performance on the XNLI (Conneau et al., 2018) benchmark.

6 Conclusion

In this paper, we release M-CID, a dataset for mulitilngual Covid-19 intent detection across English, Spanish, French, and Spanglish. We provide several baselines to show the impact of various cross lingual representations and pre-trained transformers on this dataset, along with a zero-shot, few-shot and code-switching studies of cross lingual transfer for intent detection. We show XLM-R based models provide very strong baselines compared to cross lingual embedding models. We hope that the release of M-CID will allow for further research for cross lingual intent detection in Covid chat bots.

Acknowledgments

We thank Caitlin Lohman and Claire Lesage for driving the data collection process.

References

Hanan Aldarmaki and Mona Diab. 2019. Context-aware cross-lingual mapping. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3906–3911, Minneapolis, Minnesota. Association for Computational Linguistics.

Ahmed Aly, Kushal Lakhotia, Shicong Zhao, Mrinal Mohit, Barlas Oguz, Abhinav Arora, Sonal Gupta, Christopher Dewan, Stef Nelson-Lindall, and Rushin Shah. 2018. Pytext: A seamless path from NLP research to production. *CoRR*, abs/1812.08729.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.

Alexis Conneau, Guillaume Lample, Ruty Rinott, Adina Williams, Samuel R Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Andre Esteva, Anuprit Kale, Romain Paulus, Kazuma Hashimoto, Wenpeng Yin, Dragomir R. Radev, and Richard Socher. 2020. Co-search: COVID-19 information retrieval with semantic search, question answering, and abstractive summarization. *CoRR*, abs/2006.09595.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. 2017. Allennlp: A deep semantic natural language processing platform.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1746–1751. ACL.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

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.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *Advances in Neural Information Processing Systems (NeurIPS)*.

Bing Liu and Ian Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. *arXiv preprint arXiv:1609.01454*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Sean MacAvaney, Arman Cohan, and Nazli Goharian. 2020. SLEDGE: A simple yet effective baseline for coronavirus scientific knowledge search. *CoRR*, abs/2005.02365.

Alistair Martin, Jama Nateqi, Stefanie Gruarin, Nicolas Munsch, Isselmou Abdarahmane, and Bernhard Knapp. 2020. An artificial intelligence-based first-line defence against covid-19: digitally screening citizens for risks via a chatbot. *bioRxiv*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proc. of NAACL*.

Shana Poplack. 2004. *Code-Switching*, pages 589–596.

Tal Schuster, Ori Ram, Regina Barzilay, and Amir Globerson. 2019. Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1599–1613, Minneapolis, Minnesota. Association for Computational Linguistics.

Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

WHO. 2020. *WHO launches a chatbot on Facebook Messenger to combat COVID-19 misinformation*.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.

Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. 2015. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1006–1011, Denver, Colorado. Association for Computational Linguistics.

Chenwei Zhang, Yaliang Li, Nan Du, Wei Fan, and Philip S Yu. 2018. Joint slot filling and intent detection via capsule neural networks. *arXiv preprint arXiv:1812.09471*.

Edwin Zhang, Nikhil Gupta, Rodrigo Nogueira, Kyunghyun Cho, and Jimmy Lin. 2020. Rapidly deploying a neural search engine for the COVID-19 open research dataset: Preliminary thoughts and lessons learned. *CoRR*, abs/2004.05125.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017. Adversarial training for unsupervised bilingual lexicon induction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1959–1970, Vancouver, Canada. Association for Computational Linguistics.

Xiaodong Zhang and Houfeng Wang. 2016. A joint model of intent determination and slot filling for spoken language understanding. In *IJCAI*.

Appendix

A Dataset Details

As an extension of table 1, we show the intent distribution across languages and across train, eval, and test split in table 5.

B Hyperparameters for Models

We detail the experimental set up for each of our models below. For hyperparameter tuning, we sweep over the learning rate and batch size across model architectures.

Baseline DocNN Model For all of our DocNN experiments we keep the DocNN model architecture consistent and sweep the learning rate and batch size. Here we detail the architecture. We use a CNN model with kernel sizes [3,4,5] and 100 feature maps per kernel. We employ dropout (Srivastava et al., 2014) of 0.25. We then add an MLP with hidden dimension 128 to project to the output classes. We optimize for the cross entropy loss, and leverage the AdamW optimizer (Loshchilov and Hutter, 2017). All our models are trained across 8 GPUs using distributed data parallel training with PyTorch (Paszke et al., 2019). Our effective batch size is computed by multiplying the batch size per worker by the number of workers.

MUSE DocNN We initialize our embedding layer with 300 dimension MUSE embeddings. We train for 100 epochs with an effective batch size of 512 and learning rate 0.000691 for cross lingual, 256 and 0.00135 for English, 256 and 0.000876 for Spanish, 256 and 0.00135 for French.

Frozen MUSE DocNN We use the same setup as the MUSE DocNN model however, noteably we freeze the MUSE embeddings. We train 100 epochs and use batch size 256 and learning rate 0.00134 for cross lingual, 256 and 0.00135 english 256 and 0.00135 for Spanish, 512 and 0.00233 for French.

SentencePiece (SP) DocNN We use sentence piece embeddings loaded from the XLM-R Large model with embedding dimension 1024. We use an effective batch size of 256 and learning rate 0.00178 for cross lingual, English, Spanish and French.

Frozen SP DocNN We use the same configeration as SP DocNN, however we freeze the sentence piece embeddings. We use an effective batch size of 256 and learning rate 0.000217 for cross lingual, English, Spanish and French.

Cross-lingual ELMo DocNN We use ELMo embeddings from AllenNLP (Gardner et al., 2017) and get 1024 dimension aligned embedding representations using the alignments released by Schuster et al. (2019). We train 100 epochs with an effective batch size of 512 and learning rate of 0.00223, 256 and 0.00115 for English, 256 and 0.00115 for Spanish, and 512 and 0.00222 for French.

XLM-R Base We train our XLM-R base models for 40 epochs with an effective batch size of 512. We leverage the Adam (Kingma and Ba, 2014) optimizer, and use a learning rate of 0.00002 for cross lingual training, 0.000075 for English monolingual training, 0.00005 for Spanish monolingual training, and 0.000075 for French monolingual training.

XLM-R Large Similar to XLM-R Base we train our models for 40 epochs, we leverage an effective batch size of 128. We use the Adam optimizer, and use a learning rate of 0.00002 for cross lingual training, 0.00002 for English monolingual training, 0.00001 for Spanish monolingual training, and 0.00001 for French monolingual training.

Intent	Split	Number of Occurrences EN	ES	FR	Spanglish
what_is_corona	Train	82	73	71	-
"what is coronavirus"	Eval	6	15	12	-
"can you tell me about the virus"	Test	22	12	17	15
what_if_i_visited_high_risk_area	Train	72	68	71	-
"i traveled to new york recently am i infected"	Eval	9	8	8	-
"how do i protect myself in high risk areas"	Test	24	24	21	25
what_are_treatment_options	Train	92	70	60	-
"do we have a cure yet"	Eval	9	8	8	-
"do hospitals know how to fix this"	Test	24	24	21	25
what_are_symptoms	Train	72	66	75	-
"i have a cold should i be worried"	Eval	15	16	8	-
"is coughing a sign of the virus"	Test	23	18	17	21
travel	Train	87	63	71	-
"is it safe to travel now"	Eval	5	11	10	-
"can i take the bus to work"	Test	18	26	19	26
share	Train	82	67	62	-
"share this with jack"	Eval	9	12	11	-
"send this info to my friends"	Test	19	21	27	24
protect_yourself	Train	76	68	72	-
"how can i stay safe"	Eval	14	15	8	-
"what should i do to prevent"	Test	20	17	20	25
okay_thanks	Train	71	70	61	-
"thanks for doing this"	Eval	13	9	16	-
"this is amazing"	Test	26	21	16	7
news_and_press	Train	80	73	73	-
"what's the latest"	Eval	8	7	11	-
"did anything big happen today"	Test	22	20	16	26
myths	Train	70	68	75	-
"what are myths about covid"	Eval	8	7	10	-
"what are the misconceptions"	Test	32	24	15	21
latest_numbers	Train	78	74	68	-
"what's the latest statistics"	Eval	7	9	10	-
"what do the numbers look like now"	Test	25	17	22	24
how_does_corona_spread	Train	81	71	64	-
"how does the virus spread"	Eval	9	7	10	-
"can people with masks transmit to other people"	Test	20	22	26	24
hi	Train	82	74	67	-
"hello"	Eval	9	8	15	-
"hey covid bot"	Test	19	18	18	7
donate	Train	81	67	75	-
"this is great how do i help you"	Eval	9	8	8	-
"i wish i could do something about this"	Test	20	25	17	20
can_i_get_from_packages_surfaces	Train	73	71	69	-
"is it safe to get food delivered"	Eval	8	7	11	-
"how often should i clean my table"	Test	24	22	20	25
can_i_get_from_feces_animal_pets	Train	79	62	71	-
"can i get the virus from dogs"	Eval	10	13	7	-
"should i stop eating meat"	Test	16	25	22	21

Table 5: Dataset details by intent labels. For each intent listed are the occurrences of each label in the train, eval, and test set by language. Italicised underneath each label are two samples of utterances for that intent. *Note: Spanglish is only available as a test set hence there are no training or validation samples*