XLP at SemEval-2020 Task 9: Cross-lingual Models with Focal Loss for Sentiment Analysis of Code-Mixing Language

Yili Ma, Liang Zhao, Jie Hao
OPPO Research Institute, Beijing, China
{mayili,leonzhao,haojie}@oppo.com

Abstract

In this paper, we present an approach for sentiment analysis in code-mixed language on twitter defined in SemEval-2020 Task 9. Our team (referred as LiangZhao) employ different multilingual models with weighted loss focused on complexity of code-mixing in sentence, in which the best model achieved f1-score of 0.806 and ranked 1st of subtask- Sentimix Spanglish. The performance of method is analyzed and each component of our architecture is demonstrated.

1 Introduction

Sentiment analysis is in the area of research that perform the automatic comprehension of the subjective information from user-generated data, which helps to gain the views on certain topics. Due to the rise of social media such as micro-blogs (e.g., Twitter) and the trend of global communications, they have accelerated the use of multilingual expressions, raising the concerns on code-mixing behavior (Patwa et al., 2020). To develop cross-lingual encoders that can encode any sentence into a shared embedding space, by using monolingual transfer learning, multilingual extensions of pretrained (Lample et al., 2019) encoders have been shown effective.

As for code-mixed text, more complicated than cross-lingual sentence, it is crucial to consider the complexity of texts written in several different languages because different types of integration correlate with different social contexts (Gualberto A. et al., 2016). Sometimes, the user may post blogs in non-native language with grammar mistakes or even prefer to express the sentiment in the native language. The phenomena has encouraged the researchers to analyze the sentiment from multilingual code-mixed texts. Because Spanish and English share a lot of words with Latin roots, sometimes words with the same origin take a separate path in each language, or words with different origins resemble each other by coincidence, but have different meanings. For example, éxito from Spanish means success, which resembles exit from English, with different meaning and sentiment. In the task, the number of words in a sentence vary from different languages dramatically. Intuitively, the language that has a bigger presence in the tweet would contain the sentiment of the sentence. To tackle the problem, we adopt the focal loss through calculating the ratio of each language in code-mixing text.

The rest of the paper is structured as follows: Section 2 provides the detailed implementation method. Section 3 presents the results and performance of our models as well as experiment settings. Concluded remarks and future directions of our work are summarized in Section 4.

2 Implementation details

2.1 Preprocessing

Normally, deep learning models have a simple data processing pipeline, while in the task data is very messy. Therefore we have used a more detailed method according to characteristic of the code-mixing data (URLs, emoji, hash symbol etc.)

First, user name mentioned and URL are all removed because they are useless for sentiment prediction. Special characters like "RT" representing re-tweet is also deleted. Moreover, we also remove the hash

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 975–980
Barcelona, Spain (Online), December 12, 2020.

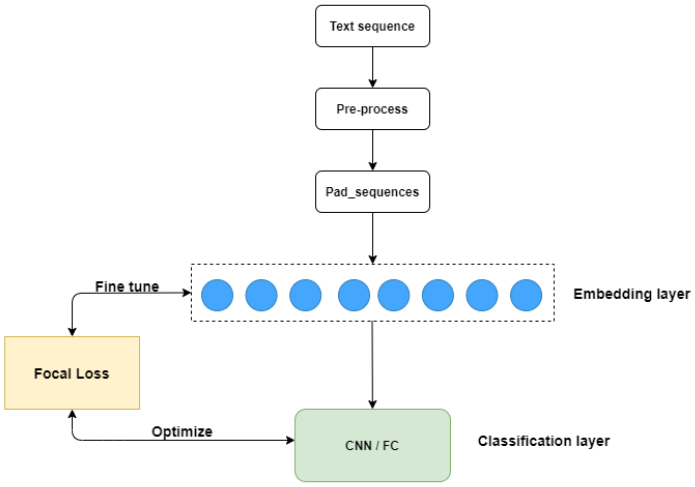

Figure 1: Illustration of our model

symbol from hash-tags as it can be problematic for tokenizers to work with. As for non-text symbol like emoji and emoticon, we use the (emoji, 2019) library from python and emoticon dictionary from wiki (List of emoticons, 2020) respectively to transform the symbols to text. Next all characters into lowercase and stop words are removed. Afterwords, we employ fastBPE to generate and apply BPE codes to get post-BPE vocabulary using vocabulary of XLM model for 100 languages including Hindi, Spanish, and English. Sentence size is limited to 256. This is enough for nearly all of the tweets after processing.

2.2 Data augmentation

In order to get more training data and based on the statistics of dataset, we have utilized machine translation (Sennrich et al., 2016) for generating more text to boost up the performance. After the original code-mixed text is translated to the target language Spanish, both source sentences and translated sentences are mixed to train a model.

2.3 Tested architectures

2.3.1 Pre-trained Models for Feature Encoding

To extract valid representation features of tweet, two state-of-the-art pre-trained sentence embedding models are utilized. Details are deliberated in the following section.

- **XLMs**: We use pretrained embeddings made available by Facebook research (Lample et al., 2019), which is unsupervised that only relies on monolingual data, and support 100 languages including English and Spanish. After fine-tuning an XLM model on the training corpus, the model is still able to make accurate predictions at test time in code-mixed languages, for which there is not enough training data. This approach is usually referred to as "zero-shot cross-lingual classification". Based on the pretrained XLM model, the sentence is indexed by vocabulary and then independently fed into the pretrained transformer model, which is also optimized during training. The single column of last hidden layer of transformer model is used as the representation of sentence, fed into a projection layer using linear transformation. While for CNN model, all columns of last hidden layer are utilized as the sentence embedding.

- **MUSE**: MUSE are multilingual embeddings based on fastText (Conneau et al., 2017), available in different languages, where the words are mapped into the same vector space across languages. We use the average representations of all words in a sentence, which is modified during training as well.

2.3.2 Output layer

Two models are examined with MUSE and XLM respectively: CNN based and linear layer based.

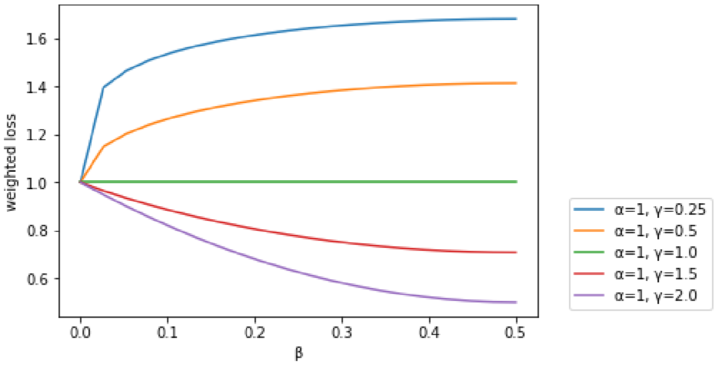

Figure 2: Weighted loss with different hyper-parameters

- **CNN Classifier**: The CNN classifier is composed of three 2-D convolution layers with filter widths ranging from three to five. Each convolution layer has 100 filters. The intermediate layers use ReLu activation. The output is global max pooled and fed into the fully connected layer with dropout rate 0.5 and the final layer is a softmax layer.
- **Linear Classifier**: The pretrained embeddings are just directly fed into a linear layer, also referred as fully connected layer and softmax afterwards to get the final predictions.

2.3.3 Optimized loss

As analysis above, non-native English speaker may misuse English due to the culture differences and lack of vocabulary, and so on. The monolingual corpus in Spanish will be more accurate in the expression of the sentiment than multilingual. On the other hand, the quality of monolingual sample may be decreased due to error from augmentation data from translation. In view of data analysis of training and test corpus, we also found that the percentage of each language e.g., English and Spanish is biased. According to the statistics, the percentage of Spanish words is twice more than English in training dataset, and almost three times in valid and test data. The test data also have 560 monolingual sentences, in which half are in English and the other are in Spanish. In this case, the model is prone to learn the unbalanced semantic information.

To benefit the gain from the samples and focus on the majority language model, we weighed the loss L_W based on the complexity of code-mixing (Gamb¨ack et al., 2014).The formula is listed as followings, where β is the percentage of Spanish words in a sentence, CE is the initial cross entropy, $\gamma > 0$ and α is a constant positive scaling factor. To better explore the trend, weighted loss with different hyper-parameters is shown in shown in Figure 2.

$$L_W = \alpha * CE * (1 - \beta)^\gamma + \alpha * CE * \beta^\gamma$$

The γ is a focusing parameter that control the loss. Larger values of γ correspond to large losses for low complexity of code-mixing sentences. When $\gamma < 1$, the model is prone to learn the multilingual data and on the contrary, if $\gamma > 1$, it's more likely to learn the monolingual data. When γ equals 1, the loss is just the cross entropy as default.

3 Experiments and Results evaluation

3.1 Dataset

Subtask in Spanish of the SemEval-2020 task 9 is to predict the sentiment of a given code-mixed tweet. The sentiment labels are positive, negative, or neutral, and the code-mixed languages will be English-Spanish. Besides the sentiment labels, also the language labels at the word level are provided. The word-level language tags are en (English), spa (Spanish), hi (Hindi), mixed, and univ (e.g., symbols, @ mentions, hashtags).

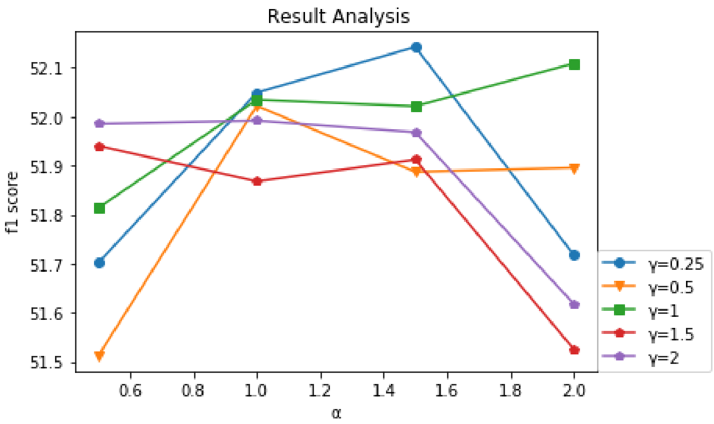

Figure 3: Mean f1 score for validation set of different hyper-parameters with XLM-FC

Models	Parameters	Valid f1-score	Test f1-score
organizer baseline	-	-	0.656
MUSE-FC	(γ =1.0)	0.4092	0.738
MUSE-FC	(γ =0.25)	0.4104	0.742
MUSE-CNN	(γ =1.0)	0.4761	0.739
MUSE-CNN	(γ =0.25)	0.4878	0.755
XLM-FC	(γ =1.0)	0.5211	**0.776**
XLM-FC	(γ =0.25)	0.5214	**0.806**
XLM-CNN	(γ =1.0)	0.5157	0.794
XLM-CNN	(γ =0.25)	0.5294	0.805

Table 1: Performance metrics of different models on validation and test sets. The average f1 scores of validation set are reported for ten runs using different random seeds to choose hyper-parameters, and the test scores are generated by using the trained model to predict on released labeled test data.

1. The trial data have 2000 tweets.

2. The train data have 15004 tweets. (They include trial data as well).

3. The train data after split have 12002 tweets.(used for training).

4. The validation data have 2998 tweets.

5. The test data have 3789 tweets.

Moreover, 5000 back-translated data are added into training. Validation subset is used as an unbiased accuracy evaluation in order to fine-tune hyper parameters during training. To evaluate the performance of the system, Precision, Recall, and F-measure are measured.

	Precision	Recall	F1-score
Positive	0.883	0.926	0.904
Negative	0.599	0.395	0.476
Neutral	0.181	0.209	0.194
Weighted	-	-	0.806

Table 2: Performance metrics of Class-wise Classification

3.2 Experiment Setup and Results

Hyper-parameter optimization is performed using a simple grid search. All models are trained with 10 epochs with a batch size of 8 and an initial learning rate 0.000005 by Adam optimizer. The linear layers are dropped out with a probability of 0.5. Unless otherwise stated, default settings are used for other parameters. In the process of searching for optimal architecture and parameters, we experimented CNN and fully connected layer (marked as FC) respectively with MUSE and XLM.

To explore and compare the optimal parameters α and γ, as shown in Figure 3, there is an obvious increasing tendency of f1 score until α >1.5 when $\gamma <= 1.0$, and reaches the highest score as γ = 0.25 and second highest as γ = 1.0, which indicates that the model has found optimal parameters prone to high level of code-mixing data. Based on the results of validation set, to select best model, we expect that the best performance is always achieved in optimal parameters as above which are γ= 0.25 or γ= 1.0. The scores are summarized in Table 1. XLM model with a fully connected layer achieved best when γ= 0.25, and from its class-wise scores, we conclude that the model performs best in classification of positive samples, while worst in neutral samples. The result can be caused by unbalanced distribution of data and complexity of code-mixing, such as the expression of positive sentiment mainly focused in specific language. CNN based model has not shown significant increase in performance compared to linear classifier.

4 Conclusion

In this paper, we have introduced a novel approach with weighted loss of different multilingual models with weighted loss focused on complexity of code-mixing sentences for sentiment analysis task in SemEval-2020. The method is effective in situation where the distribution of different languages is unbalanced, and has a better control of language preference for sentiment by the level of how languages mix. Moreover, we conclude that the quality of word representations used has a significant impact on the performance of a model. Results indicate the potency of XLM on code-mixed lingual classification, leading to 4-5 % increase in f1 score compared to MUSE. In the future, we will continue to do model optimization and also try ensemble models.

References

Patwa Parth, Aguilar Gustavo, Kar Sudipta, Pandey Suraj, PYKL Srinivas, Gambäck Björn ,Chakraborty Tanmoy, Solorio Thamar and Das Amitava. December, 2020. SemEval-2020 Task 9: Overview of Sentiment Analysis of Code-Mixed Tweets. In *Proceedings of the 14th International Workshop on Semantic Evaluation* (SemEval-2020). Barcelona, Spain. Association for Computational Linguistics,

Conneau A., Rinott R., Lample G., Williams A., Bowman S., Schwenk H., and Stoyanov V. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of EMNLP 2018, pp. 2475–2485, 2018b.*

Lample Guillaume and Conneau Alexis. 2019. Cross-lingual Language Model Pretraining. In *Advances in Neural Information Processing Systems 32*, pages 7059-7069

Sennrich Rico, Haddow Barry, and Birch Alexandra. 2016. Improving Neural Machine Translation Models with Monolingual Data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86-96.

Conneau Alexis, Lample Guillaume, Ranzato Marc'Aurelio, Denoyer Ludovic and Jégou, Hervé. 2017. Word Translation Without Parallel Data. *arXiv preprint arXiv:1710.04087.*

Jonathon Byrd and Zachary C. Lipton. 2019. What is the effect of Importance Weighting in Deep Learning? In *Proceedings of the 36 th International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019*

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*, pages 6000–6010.

Wikipedia contributors. List of emoticons — Wikipedia, The Free Encyclopedia. 2020. URL https://en.wikipedia.org/w/index.php?title=List_of_emoticons&oldid=949712309

emoji. URL https://pypi.org/project/emoji/

X. Zhang, J. Zhao and Y. LeCun. 2015. Character-level Convolutional Networks for Text Classification. In *Proceedings of the 28th International Conference on Neural Information Processing Systems,* Volume 1, pages 649–657

Xie Qizhe, Dai Zihang, Hovy Eduard, Luong Minh-Thang and Le Quoc V. 2019 Unsupervised Data Augmentation for Consistency Training. *arXiv preprint arXiv:1904.12848*

Matos Veliz Claudia, de clercq Orphee and Hoste Veronique. 2019. Benefits of Data Augmentation for NMT-based Text Normalization of User-Generated Content. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*

Van Hee Cynthia, Van de Kauter, Marjan, de clercq Orphee, Lefever Els and Hoste Veronique. 2018. Noise or music? Investigating the usefulness of normalisation for robust sentiment analysis on social media data. *Revue Traitement Automatique des Langues* , 58(1):63–87.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. 2018. Qanet: Combining local convolution with global self-attention for reading comprehension. *arXiv preprint arXiv:1804.09541, 2018.*

Rudra Koustav, Rijhwani Shruti, Begum Rafiya, Choudhury Monojit, Bali Kalika and Ganguly Niloy. 2016. Understanding Language Preference for Expression of Opinion and Sentiment: What do Hindi-English Speakers do on Twitter? In *Proceedings of EMNLP 2016*, pages 1131-1141

Guzman Gualberto A. , Serigos Jacqueline, Bullock Barbara E. , and Toribio, Almeida Jacqueline. 2016. Simple Tools for Exploring Variation in Code-switching for Linguists. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 12-20

David Vilares, Miguel Alonso Pardo, and Carlos Gómez-Rodríguez. 2016. EN-ES-CS: An English-Spanish Code-Switching Twitter Corpus for Multilingual Sentiment Analysis. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4149–4153.

Bj¨orn Gamb¨ack and Amitava Das. 2014. On Measuring the Complexity of Code-Mixing. In *Proceedings of the 11th International Conference on Natural Language Processing*, Goa, India, pages 1–7.