kk2018 at SemEval-2020 Task 9: Adversarial Training for Code-Mixing Sentiment Classification

Jiaxiang Liu, Xuyi Chen, Shikun Feng, Shuohuan Wang,
Xuan Ouyang, Yu Sun, Zhengjie Huang, Weiyue Su
Baidu Inc., China
{liujiaxiang, chenxuyi, fengshikun01, wangshuohuan}@baidu.com
{ouyangxuan, sunyu02, huangzhengjie, suweiyue}@baidu.com

Abstract

Code switching is a linguistic phenomenon that may occur within a multilingual setting where speakers share more than one language. With the increasing communication between groups with different languages, this phenomenon is more and more popular. However, there are little research and data in this area, especially in code-mixing sentiment classification. In this work, the domain transfer learning from state-of-the-art uni-language model ERNIE is tested on the code-mixing dataset, and surprisingly, a strong baseline is achieved. Furthermore, the adversarial training with a multi-lingual model is used to achieve 1st place of SemEval-2020 Task 9 Hindi-English sentiment classification competition.

1 Introduction

In today's society, the use of social media has become a necessary daily activity. As a result, a large amount of text content has been produced. Among those corpora, there are many code-mixed texts. Code-mixing is a common phenomenon in societies in which two or more languages are used. It is the change of one language to another within the same utterance or in the same oral/written text (Ho and others, 2007). Code-mixing poses several unseen difficulties in NLP tasks on lexical/syntax/semantic levels.

In recent years, a lot of models based on Transformer (Vaswani et al., 2017) have been proposed such as BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019) and ALBERT (Lan et al., 2019). Those models take advantage of both available large corpora and computational power and have outperformed the traditional techniques in various sentiment analysis tasks (Kim, 2014; Hochreiter and Schmidhuber, 1997). Meanwhile, many multi-lingual models are proposed, e.g. (Ma et al., 1999; Conneau et al., 2019). (Sarkar et al., 2019) have tested the multilingual BERT with Hierarchical Attentive Network, Shakeel and Karim (2020) have tested BERT for code-mixing informal short text classification.

However, the code-mixed corpus is small. Large models like BERT (Devlin et al., 2019) and ERNIE (Sun et al., 2019) have millions of parameters, and are prone to overfit. To tackle this problem, adversarial training can be used as a regularizer in the training phase (Szegedy et al., 2013; Goodfellow et al., 2014). Adversarial examples are a way of fooling a neural network to behave incorrectly (Szegedy et al., 2013). They are created by applying a small perturbation to the original inputs. The original algorithm was designed for use in the image field, and the perturbation added to the image was deliberately designed to be invisible to the naked eye, but causing neural networks to output a completely different response from the true one. By introducing those adversarial examples, which the neural nets make mistakes on, to the network during training, the performance can be improved. This is called "adversarial training" which acts as a regularizer to help the network generalize better (Goodfellow et al., 2014). For natural language, it is not feasible to produce perturbed examples due to the nature of the text. As a solution, Miyato et al. (2016) apply this perturbation to the embedding space and design a new loss for semi-supervised adversarial training.

Our contributions are twofold. First, the domain transfer learning method is applied for code-mixing classification, which surprisingly shows promising efficacy. And furthermore, the adversarial training is applied during the training process with a multilingual model. Finally, this method has achieved the state-of-the-art score and rank 1st in SemEval-2020 Task 9 for code-mixing Hindi-English sentiment classification.

2 Related Work

In this section, the existing research on multi-lingual sentiment classification and adversarial learning is discussed, which are closely related to this work.

2.1 Multilingual Sentiment Classification

Most of the state-of-the-art multilingual models are based on Transformer (Vaswani et al., 2017). In mBert (Devlin et al., 2019), a 12 layers model trained on a multilingual corpus including 104 languages is proposed. (Conneau and Lample, 2019) has proposed a new learning target to leverage both unsupervised and supervised corpus on monolingual and cross-lingual corpus respectively to train the cross-lingual language model XLM. (Conneau et al., 2019) upgrades the XLM to XLM-R with more training data, which significantly outperforms mBERT and XLM in many datasets including a sentiment dataset SST-2 (Socher et al., 2013) and a cross-lingual entailment classification dataset MNLI (Williams et al., 2017).

2.2 Adversarial Training

Adversarial examples were first found in (Szegedy et al., 2013) where a malicious example called adversarial example can be designed while it is not perceptible for human eyes. Both (Szegedy et al., 2013) and (Goodfellow et al., 2014) found that by putting the adversarial example back into training, the model will be regularized and can provide an additional regularization benefit beyond that provided by using dropout. Since those methods are proposed and tested in image classification, (Miyato et al., 2016) designed a semi-supervised loss based on adversarial training for sentence classification and shows promising results on text classification results. Therefore, in this work, the impact of applying adversarial training with the state-of-the-art multilingual language model is studied.

3 Model

In this section, we describe the two models that were used in this competition are proposed. The first one is ERNIE proposed by (Sun et al., 2019) which uses the paradigm of continuous learning to pretrain language models which are adapted to sentiment classification using domain transfer learning. The other one is a multi-lingual model XLM-R and we have further trained it with adversarial training.

3.1 Backbone Model

The backbone models ERNIE (Sun et al., 2019) and XLM-R (Conneau et al., 2019) used in this tasks are all based on Transformer. For this code-mixing sentiment classification task, the input sentence will be organized as

$$[CLS], w_1, w_2, ..., w_n, [SEP]$$

where the $[CLS]$ token is inserted in the beginning place of the sequence which is used as an indicator of the whole sentence, and specifically it is used to perform sentiment classification. The $[SEP]$ is a token to separate a sequence from the subsequent one and is an indicator of the end of a sentence. w_i is token of the sequence. After they go through the backbone model, for each item of the sequence, a vector representation of the size H, size of specified hidden layers, is computed. The representation of $[CLS]$ is applied with a fully connected layer to classify the three sentiment labels including negative, neutral, and positive.
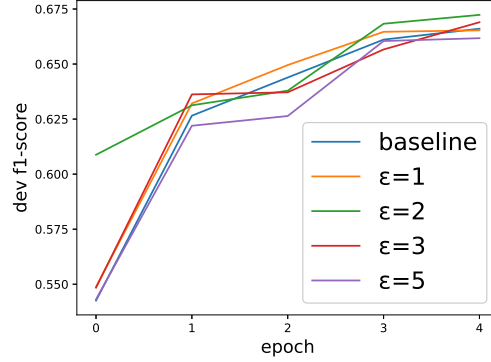
Figure 1: The effect of the hyper-parameter ϵ for adversarial training

3.2 Training with Mono-lingual Model

The ERNIE model is trained with a continuous learning method. The training tasks of ERNIE include different granularity, such as the lexical level task of capital letter predictions and masked language model for entity, syntax level task of predicting document order, and relation, e.g. entailment relation prediction.

This model is trained with monolingual corpus, including Wikipedia, BookCorpus, and Reddit. The off-the-shell model is used directly for finetuning on the code-mixing corpus.

After the code-mixing sentence goes through ERNIE, the representation of the token $[CLS]$ is used to classify the sentiment. Softmax is used to calculate the probability distribution of each class. The cross entropy is used to calculate the loss:

$$L_{\text{ce}} = -y \log p(y|x; \theta)$$

where, $p(y|x)$ is the probability for target label y.

3.3 Adversarial Training Multilingual Model

XLM-R (Conneau et al., 2019) is used as the multilingual backbone model for this task. XLM-R is trained with 100 languages with 2.5 TB corpus. The training objective is the same as RoBERTa (Liu et al., 2019) which is a masked language model, that is, the input is a corrupted sentence and the goal is to recover those corrupted tokens.

Adversarial examples are inputs to machine learning models that an attacker has intentionally designed to cause the model to make a mistake (Goodfellow et al., 2014). According to the access to the model parameters, it can be divided into two types of attack algorithms: white box attack and black box attack. The former adversarial one has full access to the model parameters (e.g. embedding, linear mapping weights), and the latter uses only on the input and output, e.g. token id, the probability distribution of model prediction. Considering that the goal of adversarial training in our experiment is to regularize the training process, the white-box method allows it to access the embeddings.

To create adversarial examples, the formula proposed by (Miyato et al., 2016) is used, where the perturbation is created using the gradient of the loss function. Assume the input embedding x and the model parameters θ is given, to find the adversarial examples the following problem should be solved:

$$\boldsymbol{r}_{\text{adv}} = \epsilon g / \|g\|_2 \quad \text{where} \quad g = \nabla_{x+d} \text{KL}[p(\cdot|\boldsymbol{x}; \hat{\boldsymbol{\theta}}) \| p(\cdot|\boldsymbol{x} + \boldsymbol{d}; \hat{\boldsymbol{\theta}})]$$

where $\text{KL}[p\|q]$ denotes the KL divergence between distributions p and q, r denotes the perturbation on the input and $\hat{\theta}$ is a constant copy of θ in order not to allow the gradients to propagate in the process of constructing the artificial examples. Solving the above problem means that we are searching for the worst perturbation while trying to maximize the divergence of the outputs between benign input x and adversarial input $x + d$ of the model, where d is a small random vector. Therefore, the perturbations are injected to the input embeddings to create new adversarial sentences in the embedding space. ϵ is the

size of the perturbation. To find values that outperform the original results, an ablation study was carried out on four values for ϵ whose results are presented in Figure 1 and discussed in Section Model Robust Analysis. After the

	Precision	Recall	F1
Positive	0.8426	0.7600	0.7992
Negative	0.7850	0.7544	0.7694
Neutral	0.6521	0.7309	0.6892
Macro Average	-	-	0.7526

Table 1: Results of our test set submissions

adversarial examples go through the network, their loss is calculated as follows:

$$L_{\mathrm{adv}}(\boldsymbol{\theta}) = \mathrm{KL}\left[p\left(\cdot|\boldsymbol{x};\hat{\boldsymbol{\theta}}\right)||p\left(\cdot|\boldsymbol{x}+\boldsymbol{r}_{\mathrm{adv}};\boldsymbol{\theta}\right)\right]$$

Effectively, the total loss is calculated by

$$L = L_{\mathrm{ce}} + \alpha L_{\mathrm{adv}}$$

as the adversarial term favors label smoothness in the embedding neighborhood, and α is a hyperparameter that controls the trade-off between standard errors and robust errors.

4 Experiments

Implementation details There are 14000/3000/3000 examples in the train/dev/test dataset respectively, for more detailed dataset information please refer to the work of (Patwa et al., 2020). To train a better model, data cleaning strategies have been applied, e.g. the URLs, hashtags, usernames are all removed. When processing the data, to increase the data diversity, we tried both retaining and discarding case information. To fully make use of the data provided by the competition, the Spanish-English code-mixing dataset is combined with Hindi-English dataset, however, it shows a negative effect with training those two datasets together. We performed all our experiments on a GPU (Nvidia Tesla V100) with 32 GB of memory. Except for the code specific to our model, we adapted the codebase utilized by XLM-R. To carry out the ablation study of adversarial training, batches of 32 were specified. For optimization, the Adam optimizer (Kingma and Ba, 2014) with a learning rate of $3e-5$ was used. Also, we have applied the linear warm strategy for the learning rate. We set $\alpha = 1$ for the adversarial loss. The experiments are conducted for 5 epochs and we pick the best model using the dev set.

Improving Generalization As the adversarial training can be a regularizer to model, it is expected to have a more robust generalization on both development and test set. From Figure 1 it can be seen that the hyperparameter ϵ is important. When $\epsilon = 2$, we get the best F1 score for the development dataset, and furthermore, the scores for this parameter are better than the baseline most of the time (green line in Figure 1). From Table 2, there is a trend with decreasing development set score but increasing score in test set score, which shows the overfitting is easing. From this, we can see the increasing generalization and robustness of the trained model.

Domain Transfer Learning Discussion. From Table 2, it can be seen that ERNIE is tested as the baseline. ERNIE is proposed as a uni-language pretrained model trained for English. We have applied the model finetuned on the Hindi-English code-mixed dataset directly as domain transfer learning, and surprisingly this model provided a strong baseline (0.6725 test score) for this task. We speculate that this result comes first because of the structural similarity between the two languages as they both belong to Indo-European languages, which is important and fully discussed in the in (Pires et al., 2019), and second, thanks to the BPE tokenizer (Sennrich et al., 2016), tokens can be processed at sub word level, thus reducing the proportion of unknown words. As a result, the ERNIE shows a strong domain transfer learning ability.

Models	Dev set score	Test set score
ERNIE 2.0	0.6571	0.6725
XLM-R	0.6660	0.7177
$+\epsilon = 1$	0.6653	0.7305
$+\epsilon = 2$	0.6723	0.7320
$+\epsilon = 3$	0.6689	0.7332
$+\epsilon = 5$	0.6674	0.7349
Ensemble	–	**0.7526**

Table 2: Score analysis for candidates of final ensemble

Ensemble Results The training set is very small compared to the large scale of the pre-training corpus and model parameters for the backbones we have used. To take full advantage of the training dataset and increase training diversity, the strategy of constructing a training dataset is important. In practice, a 10-fold cross validation is executed. During each fold, we obtain results for the test dataset which provides a large number of candidates for the ensemble. Besides, we also used different backbone network and hyperparameter settings to increase the result diversity. The ensemble method we used was mainly based on the average prediction result, that is, the probability of each category was averaged across all the ensemble candidates, and finally the category with the highest probability among the three categories was selected as the prediction result. As a result, we have reached 0.75 F1-score and achieved the 1st place in competition.

However, from Table 1, it can be seen the F1 score of the neutral class is considerably low, which is mainly caused by the low precision of the neutral class. We speculate that, despite using the adversarial training as a regularizer, the model is still more inclined to learn specific keywords as emotional category indicators, resulting in low precision for neutral class. For the positive and negative class, the model performs better, but both the precision is higher than the recall score, which shows that the sentiment is not always obvious. We leave this research in future further work.

5   Conclusion

In this paper, we tested the performance of ERNIE for domain transfer learning and applied adversarial to code-mixed sentiment analysis with XLM-R. The experiments show that ERNIE provides a strong baseline in the cross-lingual setting. After replacing the backbone model with the multilingual model XLM-R, the F1-score of sentiment classification was significantly improved, further improved after the adversarial learning was applied. In particular, we have carried out experiments on hyperparameters in adversarial learning. Different parameters have a significant influence on the experimental results, but they all achieve the purpose of enhancing the generalization of the model. As a result, we have achieved the top rank for the competition of Hindi-English code-mixed sentiment classification and our user name is kk2018. As future work, other white-box adversarial examples as well as black-box ones will be utilized to compare adversarial training methods for various sentiment analysis tasks.

References

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, pages 7057–7067.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

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, June. Association for Computational Linguistics.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Judy Woon Yee Ho et al. 2007. Code-mixing: Linguistic form and socio-cultural meaning. *The International Journal of Language Society and Culture*, 21(1).

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October. Association for Computational Linguistics.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Xin L Ma, Sanjay Kumar, Feng Gao, Calvert S Louden, Bernard L Lopez, Theodore A Christopher, Chuanlin Wang, John C Lee, Giora Z Feuerstein, and Tian-Li Yue. 1999. Inhibition of p38 mitogen-activated protein kinase decreases cardiomyocyte apoptosis and improves cardiac function after myocardial ischemia and reperfusion. *Circulation*, 99(13):1685–1691.

Takeru Miyato, Andrew M. Dai, and Ian Goodfellow. 2016. Adversarial training methods for semi-supervised text classification. cite arxiv:1605.07725Comment: Published as a conference paper at ICLR 2017.

Parth Patwa, Gustavo Aguilar, Sudipta Kar, Suraj Pandey, Srinivas PYKL, Björn Gambäck, Tanmoy Chakraborty, Thamar Solorio, and Amitava Das. 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, December. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual bert? *CoRR*, abs/1906.01502.

Anindya Sarkar, Sujeeth Reddy, and Raghu Sesha Iyengar. 2019. Zero-shot multilingual sentiment analysis using hierarchical attentive network and bert. In *Proceedings of the 2019 3rd International Conference on Natural Language Processing and Information Retrieval*, NLPIR 2019, page 49–56, New York, NY, USA. Association for Computing Machinery.

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, August. Association for Computational Linguistics.

Muhammad Haroon Shakeel and Asim Karim. 2020. Adapting deep learning for sentiment classification of code-switched informal short text. *arXiv preprint arXiv:2001.01047*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2019. *arXiv preprint arXiv:1907.12412*.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

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.

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*.