EmotionX-AlexU: Cascade BERT-based Emotion Classifier

Meena Alfons , **Marwan Torki** and **Nagwa El-Makky**
Computer and Systems Engineering Department
Alexandria University
Alexandria, Egypt
me@meenaalfons.com, {mtorki, nagwamakky}@alexu.edu.eg

Abstract

In this paper, we present a solution for the Dialogue Emotion Recognition Challenge, EmotionX-2019, based on Bidirectional Encoder Representations from Transformer (BERT) which is the state-of-the-art for Natural Language Processing with fine-tuning on dialogue utterance classification. We use cascade classification to tackle the dominance of a majority class present in the data. Cascading the classifiers allowed to improve our reported accuracy measures for emotion prediction in text.

1 Introduction

Understanding Emotions in textual conversations is a hard problem in the absence of voice modulations and facial expressions. However, as we increasingly communicate using text messaging applications and digital agents, contextual emotion detection in text is gaining importance to provide emotionally aware responses to users.

EmotionX-2019 is a classification task consists of building a model based on the given datasets that will predict an emotion for each utterance in the evaluation dialogues. See Table 1 for an example dialogue and annotated emotions.

Role	Utterance	Emotion
Rachel	Oh Okay, I'll fix that to. What's her e-mail address?	Neutral
Ross	Rachel!	Anger
Rachel	All right, I promise. I'll fix this. I swear. I'll-I'lll'll-I'll talk to her	Non-Neutral
Ross	Okay!	Anger
Rachel	Okay!	Neutral

Table 1: Example

Due to its promising applications, understanding emotions in textual conversations has been given great attention. [Gaind *et al.*, 2019] address the problem of detection, classification and quantification of emotions of text in any form. The problem of utterance emotion classification has been attempted by different techniques, such as convolutional neural networks (CNN) [Kim, 2014], CNN-DCNN autoencoder [Khosla, 2018], long short-term memory (LSTM) [Liu *et al.*, 2016], attention-based CNN [Kim *et al.*, 2018], CNN followed by an attention layer [Torres, 2018], bi-directional LSTB (BiLSTM) [Hsu *et al.*, 2018], self-attentive BiLSTM [Luo *et al.*, 2018], CNN+BiLSTM [AlBalooshi *et al.*, 2018] and Hierarchical Attention network (HAN) [Saxena *et al.*, 2018].

Multiple shared-tasks have been released to concentrate the efforts on this problem, such as EmotionX-2018 [Hsu and Ku, 2018] and EmoContext [Gupta *et al.*, 2017]. The best performer in EmotionX-2018 was [Khosla, 2018] using CNN-DCNN autoencoder based Emotion Classifier achieving 62.5% unweighted accuracy (UWA) on *Friends* dataset and 62.5% on *EmotionPush* dataset.

Emotion classification in conversations is inherently a multi-modal problem where emotions are expressed using auditory features, body language and words. In addition to that, the context of the conversation is important to infer the emotion of a specific utterance.

Detecting emotions in textual conversations mainly depends on two factors: language representation and contextual information.

In this work, we focus on the language representation part by utilizing Bidirectional Encoder Representations from Transformers (BERT) as the state-of-the-art in language model representation which can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks. We use a cascade classifier with fine-tuned BERT models on utterance emotion classification task.

The rest of the paper is organized as follows. Section 2 summarizes the given two datasets, gives the class distribution of the training and development datasets of each and describes the preprocessing operations. The proposed system is described in section 3. The training procedure including augmentation, loss weights, and max sequence length is investigated in section 4. Experiments with their validation results and test results are shown in section 5. Finally, section 6 concludes the paper.

2 Datasets

EmotionLines dataset consists of two datasets *Friends* and *EmotionPush*. The *Friends* dataset is based on annotated dialogues from the Friends TV sitcom. The *EmotionPush* dataset includes real Facebook Messenger chats which have been anonymized.

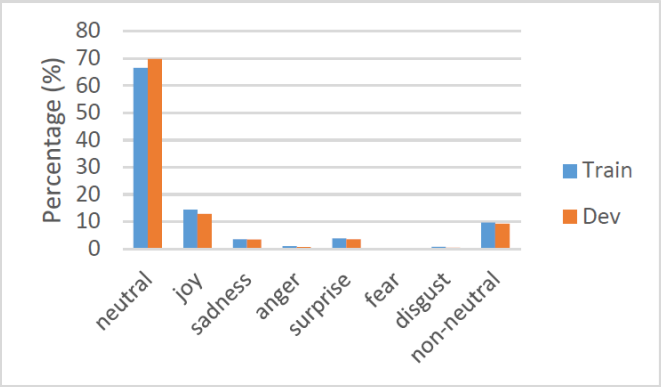

Figure 1: Friends class distribution

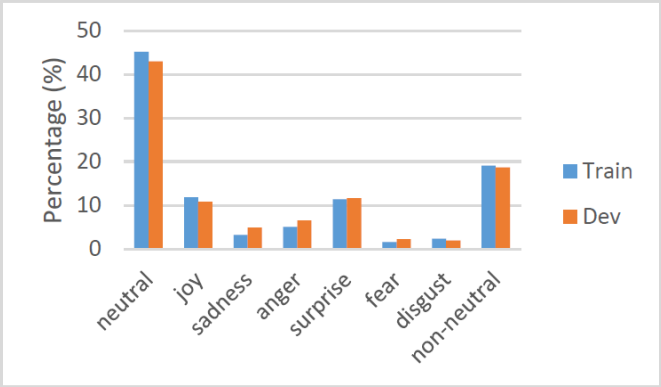

Figure 2: EmotionPush class distribution

Each of these two datasets contains 1000 English-language dialogues. Each utterance of each dialogue is annotated by one of eight labels which are *neutral*, *joy*, *sadness*, *anger*, *surprise*, *fear*, *disgust* and *non-neutral*. Refer to [Hsu *et al.*, 2018] for details on the dataset collection and construction.

2.1 Training and Development Datasets

The provided data by EmotionX-2019 task contains 1000 dialogues for *Friends* dataset with 14503 utterances and 1000 dialogues for *EmotionPush* dataset with 14742 utterances. The EmotionX-2019 challenge coordinators keep the test data with its gold labels confidential for evaluation purposes and it is only released after the results are announced.

We split the provided data to Training and Development datasets with percentages 90% and 10% respectively. The ratio is applied to the dialogues in order to preserve complete dialogues in each portion of the dataset.

Dataset	**Friends**		**EmotionPush**	
	Dialogues	Utterances	Dialogues	Utterances
Train	900	13029	900	13247
Dev	100	1474	100	1495

Table 2: Train and Development datasets

2.2 Class Distribution

Understanding the class distribution of the data is important to properly design a solution and train the model to solve the problem. Figure 1 and Figure 2 show the class distribution of the *Friends* and *EmotionPush* datasets respectively. The class distribution shows two main problems in the datasets, namely:

- The dominance of the *neutral* class.
- The imbalanced number of the samples of the other classes.

2.3 Preprocessing

Multiple preprocessing have been applied to clean up the data of both datasets. Careful inspection of the datasets shows that *EmotionPush* dataset, being chat-based dialogues, needs more clean up steps than *Friends* dataset which is well-written manuscripts from the Friends TV sitcom.

Here are the preprocessing steps applied to both datasets:

- The full text of the utterance is changed to lower case.
- Person names, locations, numbers, websites and email addresses are replaced with special tokens.
- The Emoji symbols either ASCII or Unicode are converted to their corresponding meanings.
- Chat acronyms are replaced with their corresponding meanings, for example, ("sth", "something") and ("smh", "shaking my head").
- Any other Unicode symbols are removed.
- Elongated words like "Noooooo" are replaced with "No $\langle elong \rangle$"
- Duplicated punctuation and symbols like "No!!!!!!" is replaced with "No! $\langle duplicate \rangle$"
- We use NLTK's TwitterTokenizer [Bird, 2006] to split the sentences into tokens.

3 Proposed System Description

Our solution consists of incorporating cascade classification by using two *UtteranceClassifier* instances in sequence. The first classifier is called Majority Classifier which is trained to detect the *neutral* label. The output of this classifier is a yes/no answer indicating whether the emotion of this utterances is *neutral* or not. The second classifier is called Others Classifier which is trained to distinguish between the other labels (*joy*, *sadness*, *anger*, *surprise*, *fear*, *disgust* and *non-neutral*) giving the probability of each label. The output of this classifier is the label with the maximum probability. Figure 3 shows the overall architecture of our model.

Our intuition behind making a separate classifier to detect the *neutral* label is that the *neutral* label represents around 50% of the samples. So accurately detecting the *neutral* label is an essential task to the solution and contributes significantly to the performance of the solution.

The *UtteranceClassifier* block consists of a pre-trained BERT model [Devlin *et al.*, 2018] followed by an additional

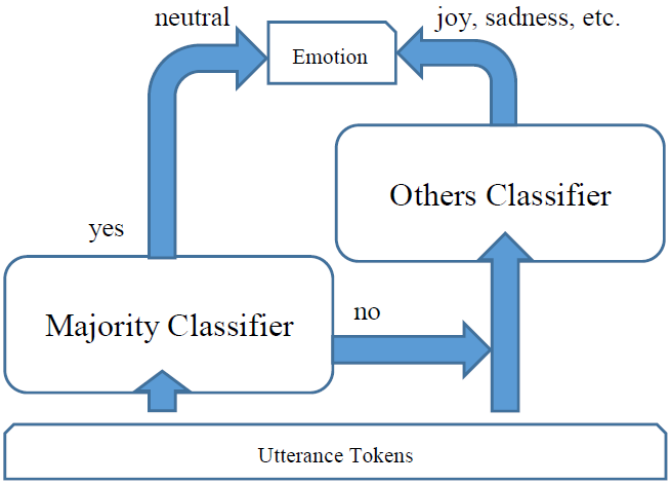

Figure 3: Overall Architecture

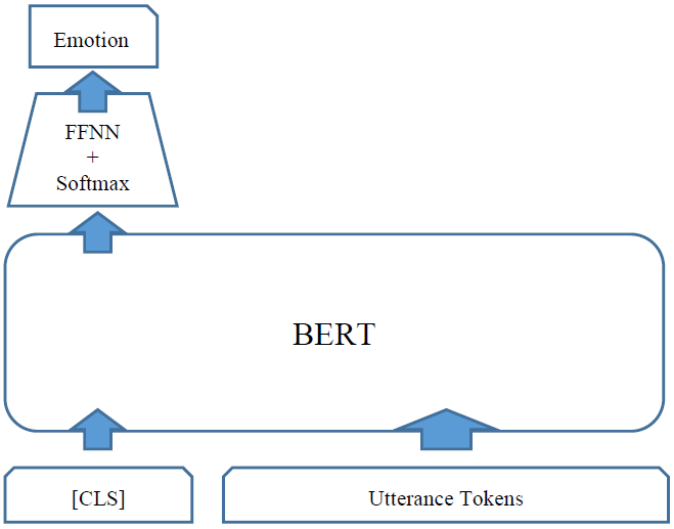

Figure 4: UtteranceClassifier

feed-forward neural-network (FFNN) layer with softmax output. The pre-trained BERT model takes a sequence of tokens representing the utterance as its input. The first token of every sequence is always the special classification embedding *[CLS]*. The final hidden state (i.e., output of Transformer) corresponding to this token is used as the aggregate sequence representation for the classification task. This hidden state (output of BERT model) is fed to the FFNN layer whose output is a set of probabilities for trained classes. The class with maximum probability is the predicted emotion. Figure 4 shows the architecture of the *UtteranceClassifier* block.

4 Training

Each classifier is trained separately and each dataset (*Friends* and *EmotionPush*) is trained separately. This gives us four *UtteranceClassifier* models to be trained:

- Friends Majority Classifier
- Friends Others Classifier
- EmotionPush Majority Classifier
- EmotionPush Others Classifier

4.1 Augmentation

The provided datasets include additional data that can be used for augmentation. Google Translate has been used to translate each utterance from English into three target languages (German, French, and Italian), then Google Translate is used to translate from the target language back into English.

We use the resultant utterances to augment the dataset by producing multiple examples for each utterance using their additional utterances with the same label of the original utterance. This helps in introducing some noise and improves the generalization of the model. Those additional utterances contain the same meaning of the original utterance but using different words or sentence structure.

4.2 Loss Weights

The datasets are class imbalanced as shown in Figure 1 and Figure 2. In order to overcome the imbalanced number of samples for each class, different loss weights have been used to give more penalty for rare labels.

Assume that $Supp_i$ is the number of samples for class i and L is the number of classes. Then $LossWeight_i$ which is the loss weight for class i, is calculated as follows.

$$Total = \sum_{i=1}^{L} Supp_i$$

$$Support = [Supp_1, Supp_2, ...] \in R^L$$

$$LossWeight_i = \frac{(Total - Supp_i)}{||Total \cdot \mathbf{1} - Support||_\infty}, \text{where} \mathbf{1} \in R^L$$

4.3 Max Sequence Length

Not all utterances have the same length. In order to normalize the classification task for the BERT model, a maximum sequence length has to be chosen. Shorter utterances are padded from the end with special tokens. Whereas, longer utterances are truncated from the end.

Analyzing the utterances lengths in the dataset showed that a max sequence length of 256 would be sufficient for most of the utterances.

5 Experiments

5.1 Experimental Setup

We are using a pre-trained $\text{BERT}_{\text{BASE}}$ model and we fine-tune it on the utterance datasets. See [Devlin *et al.*, 2018] for $\text{BERT}_{\text{BASE}}$ specifications. For fine-tuning, most model hyper-parameters are the same as in pre-training, with the exception for:

- Max sequence length: 256
- Learning rate: 1e-5

- Warm-up proportion: 0.1
- Batch size: 32

Four models have been trained with the above configurations. Each training was run for eight epochs. The output of each epoch is evaluated against the Development dataset and the best model is chosen.

The trained models are then assembled to construct the proposed system as in Figure 3. Separate systems are constructed for each dataset (*EmotionPush* and *Friends*).

The source code for our model is available as a Github repository[1].

5.2 Validation Results

The resulting systems with the trained models have been evaluated on the Development dataset with the following results. Table 3 and Table 4 show the validation results per label for the Friends and EmotionPush datasets, respectively, using the following abbreviations.

- p for precision
- r for recall
- f for f1-score

Table 5 shows a summary of the validation results of both datasets in terms of micro-f1 and macro-f1.

Label	p	r	f
neutral	82.4	100.0	90.4
joy	67.8	64.1	65.9
sadness	39.5	35.4	37.4
anger	39.8	52.2	45.2
disgust	21.1	12.9	16.0
fear	50.0	23.3	31.8
surprise	57.4	63.0	60.1
non-neutral	46.4	24.3	31.9

Table 3: Friends validation results by label

Label	p	r	f
neutral	91.5	100.0	95.6
joy	86.8	80.4	83.5
sadness	55.0	44.9	49.4
anger	14.3	08.3	10.5
disgust	50.0	27.3	35.3
fear	100.0	25.0	40.0
surprise	64.1	51.0	56.8
non-neutral	46.5	27.7	34.7

Table 4: EmotionPush validation results by label

5.3 Test Results

The task requires the utterances to be labeled with one of the following four labels: *neutral*, *joy*, *sadness* and *anger*. So, the softmax layers of the resulting systems are altered to output the most probable label of those four labels, dropping the other probabilities.

[1]https://github.com/MeenaAlfons/EmotionX-2019

Dataset	micro-f1	macro-f1
Friends	69.2	47.3
EmotionPush	86.5	50.7

Table 5: Validation results

Label	p	r	f
neutral	84.8	84.2	84.5
joy	78.6	66.9	72.3
sadness	53.9	62.8	58.0
anger	50.5	73.0	59.7

Table 6: Friends test results by label

Label	p	r	f
neutral	93.7	86.7	90.1
joy	72.7	79.0	75.8
sadness	42.2	69.1	52.4
anger	21.9	51.9	30.8

Table 7: EmotionPush test results by label

Dataset	micro-f1	macro-f1
Friends	77.0	68.6
EmotionPush	84.1	62.2

Table 8: Test results

The final evaluation of EmotionX-2019, includes computing the precision, recall, and F1-scores for each of the four labels, and then summarizing using micro F1-score. Table 6 and Table 7 show the test results per label for the Friends and EmotionPush datasets, respectively, using the previous abbreviations. Table 8 shows a summary of the test results of both datasets in terms of micro-f1 and macro-f1.

6 Conclusion and Future Work

In this work, we propose a BERT-based approach for emotion detection on *EmotionLines* dataset. We show that using the state-of-art BERT produces promising results in this task. We also show that using cascade classification improves performance on datasets suffering from the dominance of a majority class. In addition, proper preprocessing helps the network detect important features in the input and provide better predictions. Using proper classification loss weights during training helps overcome the class imbalance in the dataset. The results show that our model detects emotions successfully.

In the future, we plan to experiment with modeling emotion patterns spanning multiple utterances. This could be done by using an additional neural-network layer to capture those patterns across the output of multiple utterances. We would also experiment using two utterances with two-sequence BERT-based classification.

References

[AlBalooshi *et al.*, 2018] Hessa AlBalooshi, Shahram Rahmanian, and Rahul Venkatesh Kumar. EmotionX-

SmartDubai_NLP: Detecting user emotions in social media text. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*, pages 45–49, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[Bird, 2006] Steven Bird. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, COLING-ACL '06, pages 69–72, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[Devlin *et al.*, 2018] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[Gaind *et al.*, 2019] Bharat Gaind, Varun Syal, and Sneha Padgalwar. Emotion Detection and Analysis on Social Media. *arXiv e-prints*, page arXiv:1901.08458, Jan 2019.

[Gupta *et al.*, 2017] Umang Gupta, Ankush Chatterjee, Radhakrishnan Srikanth, and Puneet Agrawal. A Sentiment-and-Semantics-Based Approach for Emotion Detection in Textual Conversations. *arXiv e-prints*, page arXiv:1707.06996, Jul 2017.

[Hsu and Ku, 2018] Chao-Chun Hsu and Lun-Wei Ku. SocialNLP 2018 EmotionX challenge overview: Recognizing emotions in dialogues. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*, pages 27–31, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[Hsu *et al.*, 2018] Chao-Chun Hsu, Sheng-Yeh Chen, Chuan-Chun Kuo, Ting-Hao Huang, and Lun-Wei Ku. EmotionLines: An emotion corpus of multi-party conversations. In *Proceedings of the 11th Language Resources and Evaluation Conference*, Miyazaki, Japan, May 2018. European Language Resource Association.

[Khosla, 2018] Sopan Khosla. EmotionX-AR: CNN-DCNN autoencoder based emotion classifier. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*, pages 37–44, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[Kim *et al.*, 2018] Yanghoon Kim, Hwanhee Lee, and Kyomin Jung. AttnConvnet at SemEval-2018 task 1: Attention-based convolutional neural networks for multi-label emotion classification. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 141–145, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[Kim, 2014] Yoon Kim. 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 2014. Association for Computational Linguistics.

[Liu *et al.*, 2016] Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. Recurrent Neural Network for Text Classification with Multi-Task Learning. *arXiv e-prints*, page arXiv:1605.05101, May 2016.

[Luo *et al.*, 2018] Linkai Luo, Haiqin Yang, and Francis Y. L. Chin. EmotionX-DLC: Self-attentive BiLSTM for detecting sequential emotions in dialogues. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*, pages 32–36, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[Saxena *et al.*, 2018] Rohit Saxena, Savita Bhat, and Niranjan Pedanekar. EmotionX-area66: Predicting emotions in dialogues using hierarchical attention network with sequence labeling. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*, pages 50–55, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[Torres, 2018] Johnny Torres. EmotionX-JTML: Detecting emotions with attention. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*, pages 56–60, Melbourne, Australia, July 2018. Association for Computational Linguistics.