573 Project

Yu Ying Chiu, **Jiayi Yuan**, **Yian Wang**, **Chenxi Li**
Department of Linguistics
University of Washington
Seattle, WA, USA
{kellycyy, jiayiy9, wangyian, cl91}@uw.edu

Abstract

This paper presents a system for emotion and personality detection in open-domain dialogue - the FETA-Friends dataset which contains transcripts for all 10 seasons of the TV show Friends. We choose emotion detection as the primary task and personality detection as the secondary task. For the emotion detection task, the dataset includes utterances with annotated emotions, and the evaluation is based on the micro-F1 and W-F1 scores. For the personality detection task, the dataset includes short conversations with annotated binary personality traits, and the evaluation is based on accuracy. Our system uses BERT encoding and a classification layer for prediction, and includes a lexicon baseline model for comparison

1 Introduction

Large language models such as BERT have shown remarkable success in various natural language processing tasks. However, leveraging these models for downstream tasks such as sentiment analysis often requires fine-tuning on specific datasets. In this work, we operates on the FETA-Friends dataset, which contains transcripts for all 10 seasons of the TV show Friends, and presents a system that does emotion recognition and personality detection on this dataset.

2 Task description

We select the shared task (FETA challenge) by Albalak et al. (2022). It is a new benchmark for few-sample task transfer in open-domain dialogue. In this benchmark, we selected the FETA-Friends as our dataset from Chen and Choi (2016). It involves transcripts for all 10 seasons of the TV show (Friends).

The FETA-friends benchmark contains 7 tasks. We selected Emotion Recognition (Emory NLP) by Zahiri and Choi (2017) as our primary task and Personality Detection by Jiang et al. (2020) as our adaptation task.

For the primary task, the dataset contains utterances with one of the annotated emotions (eg. Neutral, Joyful, Powerful, Mad, Scared etc.). According to the dataset github [1], the dataset fields include the utterance id, speaker name, transcript, tokens and annotated emotion. It is an utterance-level classification task. The evaluation is to calculate the micro-F1 and W-F1 scores of the prediction (classification output).

For the adaptation task, the dataset contains short conversations with annotated binary Big Five personality traits (Agreeableness, Conscientiousness, Extroversion, Openness, and Neuroticism). According to the dataset github [2], each personality trait was annotated on a scale of -1, 0 , 1. The annotation of each trait in a short conversation will be summed up. The dataset contains scene id, character, AGR (Agreeableness), CON (Conscientiousness), EXT (Extroversion), OPN (Openness), NEU (Neuroticism), and text. It is a dialogue-level classification task. The evaluation is to calculate the accuracy of the prediction output.

For the key dimensions, the primary task has emotion as affect type, classification as recognition type, TV transcript as Genre, aspect-specific emotion as the target, text as modality, and English as language. For the adaptation task, it has personality type as affect type, aspect-specific personality detection as target, and all the other dimensions are the same as the primary task selected.

The FETA benchmark [3] provides access to data (train, test, dev), training and evaluation tool as a reference. The original author of dataset [4] also

[1] https://github.com/emorynlp/emotion-detection
[2] https://github.com/emorynlp/personality-detection
[3] https://github.com/alon-albalak/TLiDB/blob/master/FETA_README.md
[4] https://github.com/emorynlp

provides access to the same dataset.

3 System Overview

Our system is built for emotion detection on dialogue transcripts from the TV show 'Friends' provided by FETA challenge. The system firstly load the dataset from the provided FETA benchmark, and preprocesses the dataset by adding the required annotation specified by the command, and transformed the . During training, it encodes the original transcript using pretrained BERT model[5], and a classification layer is added in the end to make the predication. The parameters on the classification layer is trained uses Adam optimization algorithm to update the network weight each training epoch, and repeats for the number of epochs defined by the command, default to be 10 epochs. After training, evaluation is done against test set. In the end, it outputs the metrics and the prediction files. It also saves the trained model to the saved model path to allow future evaluation on the same model. The provided command ran evaluation on the models that are already trained as described above.

Apart from using the pretrained BERT model as encoder, we also implemented another baseline model to compare the results of our current and future model with. The baseline model involves creating lexicons using bag of words for each emotion label from the training data and using these lexicons to make label predictions for dev and test data.

4 Approach

4.1 Baseline - lexicon model

We implement a lexicon model as the baseline. It is a simple bag-of-words method that uses lexicons to detect emotion. We will first extract a lexicon dictionary from the training data. The dictionary has two keys, the first key is a lexicon, the second key is the emotional label, and the value is its frequency. For example, word_dict['like']['joyful'] = 50, word_dict['like']['peaceful'] = 20. To obtain the dictionary, we will iterate through the training data. For every utterance in the training data, we will separate it into tokens. Then we will add one frequency for the utterance's true label to every token in this utterance.

After we obtain the dictionary, we will use it to make prediction for dev and test set. For every utterance in dev and test set, we will separate it into tokens, and for each token, we will compute its possibilities based on frequency. The possibility for an utterance is computed as the sum of the possibilities of its tokens. The utterance will be predicted as the label with the highest possibility.

4.2 Baseline - BERT model

We implemented another model based on BERT. Adam has been used as its optimizer and 3e-5 as the learning rate of our model, We input the original dataset without any prepossessing to test for the base BERT model performance. The task is a multi-class classification task. Our model contains linear neural network layers without restriction of maximum length and with the number of possible classes. The input data is tokenized and passed through those layers. A classification layer is added at the end for outputting the classified emotion types.

We trained the model for the train data (8629 examples) provided for 10 epochs. Then we evaluated the model with dev data (2065 examples) for 10 epochs. For testing, we used the test data (110 examples) for comparison.

5 Results

This presents the results of the model.

5.1 Baseline - lexicon model

The model attained 0.3534 as F1-micro and 0.2589 as F1-weighted on dev set. It attained 0.3215 as F1-micro and 0.2219 as F1-weighted on test set.

5.2 Baseline - BERT model

For training on the 10th epoch, the model attained 0.9735 as F1-micro and 0.9734 as F1-weighted. For evaluation on the 10th epoch, the model attained 0.3705 as F1-micro and 0.3664 as F1-weighted. For testing, the model attained 0.4189 as F1-micro and 0.3913 as F1-weighted.

6 Discussion

As shown in the results section, the F1 score for both the baseline model and the BERT model have large improvement space. We believe the low score of the BERT model is largely due to different domains: BERT is known to be trained from formal texts like books, but our task is to classify emotions in a more informal setting - daily dialogues. That

[5] https://huggingface.co/bert-base-uncased

said, here is a summary of what we plan to do for D3 deliverable.

Building a new lexicon: the lexicon we used in baseline is a simple bag of words model; we plan to build a lexicon for each emotion label using a more sophisticated metric. The process will be divieded into seeds selection and lexicon expansion. For seeds selection, we are considering manually selecting seeds for each lable, for instance, selecting 'anger' for label 'mad'. We might also adopt a more data-driven method in this part considering the transcription in our data is daily dialogue that is short and that might not contain many such words. Secondly, we will expand the lexicon using the selected seeds either by similarity, such as the cosine similarity between the vector of the seed and the vectors of the target words, or by co-occurrence of seeds and target words in the training data.

Pre-processing: in the current two models, we did not engage in much pre-processing of the original transcript. For D3, we plan to try out different pre-processing techniques including removing stop words and punctuation; applying stemming or lemmatization on tokens; removing numbers; and removing words based on POS tag.

Machine Learning models for classification: Jain et al. (2017) did a comparative analysis of different models and concluded Naive Bayes performed the best in their study of emotion detection in the multi-lingual texts. After we create a new lexicon, we will obtain the vector for each input based on the lexicon, we plan to try on different machine learning models to see which one can yield the best results.

Fine tuning BERT: as mentioned earlier, only adding a linear classification layer in the end does not achieve very high F1 score. We plan to fine tuning BERT by create a new module of neural network that operates on the results of BERT. Specifically, we plan to pass the transcritp into BERT and obtain the [CLS] BERT embedding. We then use this embedding as the input of our own model, testing different hyper-parameters to achieve a decent accuracy in the training data.

7 Ethical Considerations

This part discusses ethical considerations.

8 Conclusions

This part presents the conclusion.

References

Alon Albalak, Yi-Lin Tuan, Pegah Jandaghi, Connor Pryor, Luke Yoffe, Deepak Ramachandran, Lise Getoor, Jay Pujara, and William Yang Wang. 2022. FETA: A benchmark for few-sample task transfer in open-domain dialogue. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10936–10953, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yu-Hsin Chen and Jinho D Choi. 2016. Character identification on multiparty conversation: Identifying mentions of characters in tv shows. In *Proceedings of the 17th annual meeting of the special interest group on discourse and dialogue*, pages 90–100.

Vinay Kumar Jain, Shishir Kumar, and Steven Lawrence Fernandes. 2017. Extraction of emotions from multilingual text using intelligent text processing and computational linguistics. *Journal of computational science*, 21:316–326.

Hang Jiang, Xianzhe Zhang, and Jinho D Choi. 2020. Automatic text-based personality recognition on monologues and multiparty dialogues using attentive networks and contextual embeddings (student abstract). In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 13821–13822.

Sayyed M Zahiri and Jinho D Choi. 2017. Emotion detection on tv show transcripts with sequence-based convolutional neural networks. *arXiv preprint arXiv:1708.04299*.

Appendix A: Work Split

work split in detail

Appendix B: Packages Used in the Systems

Link to the code repository on github:
`https://github.com/kellycyy/LING573-project`

Off-the-shell tools used in code:

- `sample package` how the package is used for what