NLP Emotion Detection Research Report

Yuanxin Wang, Yiting Feng

Abstract

Emotion detection algorithms are widely adopted in various NLP applications including conversational AI and recommendation systems. Our project focuses on the fine-grained emotion detection dataset GoEmotions. In this report, we discusses the preliminary work on emotion detection research, presents our reproduction result, and proposes several modeling and non-modeling directions for future. This report is divided into four sections: literature review, baseline results reproduction, error analysis, and our proposals.

1 Related Work

1.1 Emotion Datasets

Early work on emotion detection such as Affective Text (Strapparava and Mihalcea, 2007) modeled emotions with six basic emotions: anger, disgust, fear, joy, sadness and surprise, following Ekman's model (Ekman, 1992). Following-up works introduced both manually-constructed (Bostan and Klinger, 2018) and weakly-labeled emotion datasets (Wang et al., 2012). The most recent benchmark GoEmotions (Demszky et al., 2020) includes the largest human-annotated dataset, with multiple annotations per example, which is the benchmark tested in our work.

1.2 Emotion Classification Models

Both feature-based and neural models have been adopted to computationally model emotion and automate emotion classification.

Lexicon-based Methods Using the categorical basic emotion model(Plutchik, 1984), the emotion classification task benefits from early manually-crafted lexicons of valence, arousal, and dominance (Bradley and Lang, 1999; Warriner et al., 2013). The reliability of those lexicons is further improved by fixing the consistency of annotations (Mohammad, 2018), which further assisted the emotion detection task.

Early Neural-based Methods Early neural-based methods found that LSTM-based neural network could capture sequence information and assist emotion classification (Hochreiter and Schmidhuber, 1997; Lin et al., 2020a). Neural-based methods benefits from the representation from large pretrained transformer-based models like BERT (Devlin et al., 2019) which reached state-of-the-art performance on various NLP tasks including emotion classification: (Demszky et al., 2020) found that BERT-based methods outperforms BiLSTM model. Recent boosts of classification models revealed multiple directions, from improving the decoding modules to novel pretraining methods: (Hofmann et al., 2020) revealed the importance of learning properties of events as latent variables (for instance that the uncertainty and the mental or physical effort associated with the encounter of a snake leads to fear). Most of the recent work in emotion recognition leverage the transformer architecture, attention mechanism, convolutions and recurrent neural networks to encode different levels of information in the document. (Adoma et al., 2020) finetunes BERT on the emotion task. (Cheng et al., 2020) uses CNN to encode local features between words, and BiLSTM to encode contextual information in the sentence. (Buechel et al., 2020) presented EMOCODER, a modular decoder architecture that generalizes emotion analysis over different tasks (sentence-level, word-level, label-to-label mapping), domains (natural languages and their registers), and label formats (e.g., polarity classes, basic emotions, and affective dimensions).

1.3 Recent Improvements

To better encode information on different levels, (Liao et al., 2021) proposes a graph neural network with three levels, in which tokens are nodes. As the level increases, tokens with longer distance are connected in the graph. Attention mechanism is applied on adjacent nodes to update node representations.

Another direction that gives performance boost is to leverage the semantic information in the emotion label embeddings and label correlations. (Gaonkar et al., 2020) computes attention between emotion labels and the input document to capture association between the labels and words in the document. For multi-label classification, labels that are positively correlated with each other are much more likely to co-occur. (Gaonkar et al., 2020) proposes a loss function that penalizes predictions with negatively correlated labels on top of the cross entropy loss.

To imitate how people classify sentences in daily life, (Lin et al., 2020b) proposes a framework that makes decisions by comparison. The framework randomly samples positive and negative instances and trains a neural network to compute the similarities between the current sentence embedding and the sampled embeddings.

While emotion recognition is a classification problem at document level, it can potentially benefit from tasks at a more fine-grained level. One related task is emotion-cause pair extraction (ECPE). ECPE is a task at clause level that aims to extract the emotion clause and corresponding cause clause in a document. (Tang et al., 2020) jointly models clause-level emotion detection and emotion-cause pair extraction and shows performance advancement in both tasks. Another related task is aspect-level sentiment analysis that classifies the sentiment polarity for a specific target in the document. (Liao et al., 2020; Han et al., 2020) both encode the document and the aspect tokens separately, and then apply cross attention between the document and the aspect to incorporate aspect semantic information and pay attention to different positions of the document for different aspects.

2 Results Reproduction and Analysis

In this section, we will discuss the detailed steps on how to reproduce the metric numbers reported in the original paper. As discussed in the previous section, in this report our scope is limited only to the benchmark GoEmotions (Demszky et al., 2020) dataset. The precision, recall, F1 scores for each of the 28 emotion class of their original proposed BERT baseline is included in Table 2. The evaluation number we focus on reproducing is the macro-average F1 score across all 28 emotion classes. The reported average F1 score is 0.46 and our reproduced average F1 score is 0.47, which we believe falls in a reasonable range. Further analysis of the differences for some class-specific scores will also be discussed later.

The original code provided in (Demszky et al., 2020) is written in TensorFlow 1.1.5 and has various deprecated versioning issues for TensorFlow functions. Also, this model takes a very long time to train on the P100 Google Colab GPU (10h for only 3 epochs). Furthermore, even though we stick to the same configurations and hyperparameters reported in the official repository, the evaluation F1 score stays at around 20% for the first 5 epochs, which does not agree with the reported results. Due to the consideration of both time constraints and performance degradation issues, we decide to discard this implementation.

Our selected code is implemented in HuggingFace (PyTorch) that presents very similar metric numbers compared to the ones reported in the original paper. As shown in Table 3, we can see that besides the average F1 scores, most of the per class F1 scores are also similar within a reasonable range. It is interesting to see that for few cases, our reproduced precision scores are generally a little higher while the reproduced recall scores are lower. To further investigate this situation, we compared thoroughly the hyperparameter settings of both the original TensorFlow model and the HuggingFace model as shown in Table 1 below.

As we can observe from Table 1, most of the hyperparameters align well and only a few of them are different. For the config ”Do Lower Case”, since we are loading the ”bert-base-cased” model as the backbone as required in the original paper, we believe it would not be reasonable to do lower case operation on the input sentences; otherwise, there is no need to use the ”case-based” model. For the config ”Add Neutral Label”, the original code adds a neutral label to every test set instance while our new code does not. This might explain why the recall number for class ”neutral” is significantly higher in the original code, but could not influence the precision and recall for other classes.

Configuration	Original	Ours
Framework	TensorFlow	PyTorch
Adam Learning rate	5e-5	5e-5
Warmup proportion	0.1	0.1
Epochs	4	4
Max Seq Len	50	50
Batch Size	16	16
Eval Threshold	0.3	0.3
Eval Threshold	0.3	0.3
Do Lower Case	False	True
Add Neutral Label	True	False
Freeze Bert Layer	True	False

Table 1: Configuration Comparison between TF and PyTorch Code

In our code, we find that HuggingFace implicitly fine-tunes the pretrained BERT layers while the original TF code freezes these layers, which could explain why our results are a little bit better and why certain rows are different. We also believe that the differences in GPU compute engines, Python library versions, and deep learning framework will also affect the scores. Given that our F1 scores per class, as well as the average F1 score, are successfully reproduced, we decide to move forward to error analysis and model improvement brainstorming.

3 Error Analysis

In this section, we will discuss some potential issues of the baseline BERT models which could guide our directions in future improvements.

First, we visualize the F1 scores for each of the classes directly and observe which classes have significantly lower scores than others. In the following discussion, please check our Appendix 5 for mapping class indexes to class label names. As shown in Figure 1, we notice that for class index 16 (grief, 0.00), class index 21 (pride, 0.10), and class index 23 (relief, 0.13).

Before diving into some exemplar cases for wrong predictions for each of these three classes, we plot the distribution of the number of test set instances for each class. It is noticeable that the three poor performance classes are all minority classes: ”grief” class has only 6 instances, ”pride” class has only 16 instances, and ”relief” class only has 11 instances. This indicates that the results for these minority classes might not be statistically significant. However, it might be hasty to conclude that

Emotion	Precision	Recall	F1
admiration	0.53	0.83	0.65
amusement	0.70	0.94	0.80
anger	0.36	0.66	0.47
annoyance	0.24	0.63	0.34
approval	0.26	0.57	0.36
caring	0.30	0.56	0.39
confusion	0.24	0.76	0.37
curiosity	0.40	0.84	0.54
desire	0.43	0.59	0.49
disappointment	0.19	0.52	0.28
disapproval	0.29	0.61	0.39
disgust	0.34	0.66	0.45
embarrassment	0.39	0.49	0.43
excitement	0.26	0.52	0.34
fear	0.46	0.85	0.60
gratitude	0.79	0.95	0.86
grief	0.00	0.00	0.00
joy	0.39	0.73	0.51
love	0.68	0.92	0.78
nervousness	0.28	0.48	0.35
neutral	0.56	0.84	0.68
optimism	0.41	0.69	0.51
pride	0.67	0.25	0.36
realization	0.16	0.29	0.21
relief	0.50	0.09	0.15
remorse	0.53	0.88	0.66
sadness	0.38	0.71	0.49
surprise	0.40	0.66	0.50
macro-average	0.40	0.63	**0.46**
std	0.18	0.24	0.19

Table 2: Results Per Class in the Original Paper

Emotion	Precision	Recall	F1
admiration	0.60	0.74	0.66
amusement	0.74	0.89	0.81
anger	0.47	0.54	0.50
annoyance	0.34	0.40	0.36
approval	0.36	0.42	0.39
caring	0.42	0.47	0.44
confusion	0.41	0.52	0.46
curiosity	0.46	0.70	0.55
desire	0.57	0.46	0.51
disappointment	0.32	0.29	0.30
disapproval	0.35	0.46	0.40
disgust	0.46	0.44	0.45
embarrassment	0.61	0.41	0.49
excitement	0.40	0.48	0.43
fear	0.59	0.75	0.66
gratitude	0.90	0.91	0.91
grief	0.00	0.00	0.00
joy	0.53	0.68	0.60
love	0.73	0.85	0.79
nervousness	0.43	0.36	0.39
neutral	0.65	0.67	0.66
optimism	0.54	0.58	0.56
pride	0.50	0.05	0.10
realization	0.29	0.18	0.23
relief	0.50	0.07	0.13
remorse	0.56	0.83	0.67
sadness	0.50	0.55	0.52
surprise	0.56	0.52	0.54
macro-average	0.49	0.45	**0.47**
std	0.16	0.25	0.21

Table 3: Our Reproduced Results Per Class

the lack of test instances is the single source of reason for the poor performance of the three classes; we also notice that there are other minority classes that perform well, such as the ”nervousness” class. These findings inspire us to do some data sampling techniques and perform controlled experiments in Assignment 4 before we actually start implementing new modeling assumptions.

With the above issue in mind, we still need to observe a few wrongly predicted examples for all these classes and see if there are any hints on modeling improvements. Here we select some misclassification examples in Table 4 as a reference. Our current predictor seems to be able to capture the overall sentiment/emotion of the sentences but missing more fine-grained divisions. We see that in most of the cases we have a recall issue: our predictor does not predict the three aforementioned classes. Besides, since the ”neutral” class has over 1700 instances, it is frequently predicted as a false positive. Now we are more confident about the conclusion that due to the imbalanced nature of the dataset, the low recall in minority classes results in low F1 scores in them too.

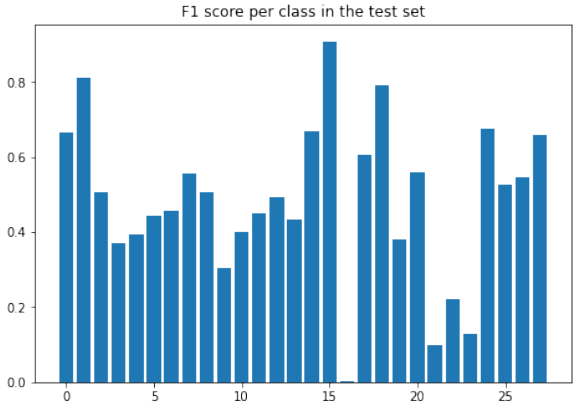

Figure 1: F1 Score for Each Class

4 Directions of Improvements

In this section, we will propose some ideas of improvements to explore.

Inspired by (Gaonkar et al., 2020; Liao et al., 2020; Han et al., 2020), we plan to leverage the rich semantic information in the pre-trained embeddings of the emotion labels. We may apply the attention mechanism between the emotion labels and the input documents. In our dataset, the emotion labels are imbalanced. For example, the admiration class is labeled much more frequently than grief and realization, and our current model is

Sentence	Prediction	Ground Truth
You’ll miss a begging old man asking for a spare coin. RIP	’neutral’	’grief’, ’sadness’
My condolences.	’sadness’	’grief’, ’sadness’
I am proud of you random internet stranger, you good today.	’admiration’	’admiration’, ’pride’
Yep. I did this in uni, got mad respect for holding my ”booze”.	’admiration’, ’approval’	’pride’
I am proud to be racist No one in real life will know this	’excitement’	’pride’
at least it wasn’t the evil [NAME].	’neutral’	’relief’
Glad I’m not the only one	’joy’	’joy’, ’relief’

Table 4: Error Case Analysis

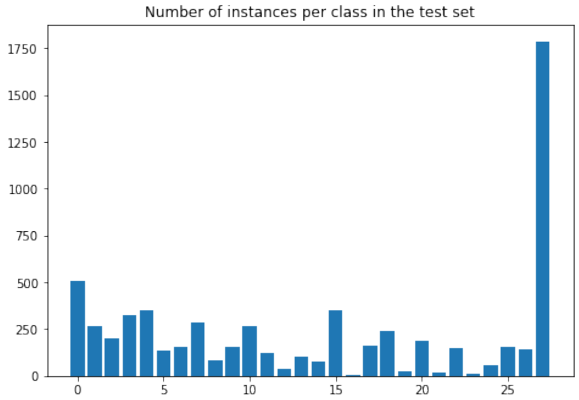

Figure 2: Number of Test Instances for Each Class

performing much better on the former class than the latter ones. This method leverages the information extracted in a much larger corpus, and we expect it to improve the performance for the less popular classes. Moreover, this method may address the within sentence emotion shift problem by paying attention to different parts of the sentence for different emotions.

In our multi-label classification dataset, more than 16% documents are labeled with more than one classes. Some emotion labels co-exist often, while others almost never co-exist. Similar to (Gaonkar et al., 2020), we can experiment with loss functions that penalize predictions which contain unusual label pairs and encourage positively correlated pairs.

Another idea is to encode information in each emotion class explicitly. While (Lin et al., 2020b) samples positive and negative examples and encode them in the framework for binary sentiment classification, we can sample examples from each emotion class for our model. We hypothesize that incorporating the same number of samples from each class into our model will boost the performance for the less popular classes.

We also get inspirations from NLP tasks not limited to emotion detection or general classification problems. As mentioned in (Khandelwal et al., 2020b) and (Khandelwal et al., 2020a), the authors take advantage of the power of training resources, or more generally, external knowledge, to boost the performance on the test set performance of language modeling tasks. We plan to experiment with the following directions: first, at inference time, mix the embeddings of a test sentence with the top-n most similar in the training sentences before the dense layer; second, run multiple inferences on the test set sentence as well as the retrieved similar training sentences and mix the final prediction logits before the softmax layer. We expect the mixing with training knowledge sources will also remedy the data imbalance problem discussed above.

References

Acheampong Francisca Adoma, Nunoo-Mensah Henry, Wenyu Chen, and Niyongabo Rubungo Andre. 2020. Recognizing emotions from texts using a bert-based approach. In *2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, pages 62–66. IEEE.

Laura Ana Maria Bostan and Roman Klinger. 2018. An analysis of annotated corpora for emotion classification in text. In *COLING*.

M. Bradley and P. Lang. 1999. Affective norms for english words (anew): Instruction manual and affective ratings.

Sven Buechel, Luise Modersohn, and U. Hahn. 2020. Towards a unified framework for emotion analysis. *ArXiv*, abs/2012.00190.

Yan Cheng, Leibo Yao, Guoxiong Xiang, Guanghe Zhang, Tianwei Tang, and Linhui Zhong. 2020. Text sentiment orientation analysis based on multichannel cnn and bidirectional gru with attention mechanism. *IEEE Access*, 8:134964–134975.

Dorottya Demszky, Dana Movshovitz-Attias, J. Ko, Alan S. Cowen, Gaurav Nemade, and S. Ravi. 2020. Goemotions: A dataset of fine-grained emotions. In *ACL*.

J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

P. Ekman. 1992. An argument for basic emotions. *Cognition Emotion*, 6:169–200.

Radhika Gaonkar, Heeyoung Kwon, Mohaddeseh Bastan, Niranjan Balasubramanian, and Nathanael Chambers. 2020. Modeling label semantics for predicting emotional reactions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4687–4692.

Yue Han, Meiling Liu, and Weipeng Jing. 2020. Aspect-level drug reviews sentiment analysis based on double bigru and knowledge transfer. *IEEE Access*, 8:21314–21325.

S. Hochreiter and J. Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9:1735–1780.

Jana Hofmann, Enrica Troiano, K. Sassenberg, and Roman Klinger. 2020. Appraisal theories for emotion classification in text. In *COLING*.

Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2020a. Nearest neighbor machine translation.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2020b. Generalization through memorization: Nearest neighbor language models.

Wenxiong Liao, Bi Zeng, Jianqi Liu, Pengfei Wei, Xiaochun Cheng, and Weiwen Zhang. 2021. Multi-level graph neural network for text sentiment analysis. *Computers & Electrical Engineering*, 92:107096.

Wenxiong Liao, Bi Zeng, Xiuwen Yin, and Pengfei Wei. 2020. An improved aspect-category sentiment analysis model for text sentiment analysis based on roberta. *Applied Intelligence*, pages 1–12.

Yuan Lin, J. Li, L. Yang, K. Xu, and Hongfei Lin. 2020a. Sentiment analysis with comparison enhanced deep neural network. *IEEE Access*, 8:78378–78384.

Yuan Lin, Jiaping Li, Liang Yang, Kan Xu, and Hongfei Lin. 2020b. Sentiment analysis with comparison enhanced deep neural network. *IEEE Access*, 8:78378–78384.

Saif M. Mohammad. 2018. Obtaining reliable human ratings of valence, arousal, and dominance for 20, 000 english words. In *ACL*.

R. Plutchik. 1984. Emotions : a general psychoevolutionary theory.

C. Strapparava and Rada Mihalcea. 2007. Semeval-2007 task 14: Affective text. In *SemEval@ACL*.

Hao Tang, Donghong Ji, and Qiji Zhou. 2020. Joint multi-level attentional model for emotion detection and emotion-cause pair extraction. *Neurocomputing*, 409:329–340.

W. Wang, L. Chen, K. Thirunarayan, and A. Sheth. 2012. Harnessing twitter ”big data” for automatic emotion identification. *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing*, pages 587–592.

Amy Beth Warriner, V. Kuperman, and M. Brysbaert. 2013. Norms of valence, arousal, and dominance for 13,915 english lemmas. *Behavior Research Methods*, 45:1191–1207.

5 Appendices

Label Name	Label Index
admiration	0
amusement	1
anger	2
annoyance	3
approval	4
caring	5
confusion	6
curiosity	7
desire	8
disappointment	9
disapproval	10
disgust	11
embarrassment	12
excitement	13
fear	14
gratitude	15
grief	16
joy	17
love	18
nervousness	19
optimism	20
pride	21
realization	22
relief	23
remorse	24
sadness	25
surprise	26
neutral	27

Table 5: Mapping between Label Name to Label Index