ECE 1786

Creative Applications of Natural Language Processing

EMOJIMOTION
Final Report

Project Name: Emojimotion

Group Members:
Ge Gao 1003803280,
Yuchen Yuan 1004549740

Word Count: 1995

<table>
<thead>
<tr>
<th>Name</th>
<th>Permission to post video</th>
<th>Permission to post report</th>
<th>Permission to post source code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yuchen Yuan</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Ge Gao</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Abstract—This paper would demonstrate how to predict emojis based on the input sentence and selected emotion using natural language processing. Training a dataset of tweets and emoji labels with baseline model and Bert model to predict emojis based on context meaning. Final emoji prediction is resulted by combining context scores with emotion scores.

Keywords—NLP, Bert, Emoji

I. INTRODUCTION

Emoji originated from Japan in 1997 and it became widely used in texting, social media posts and webpages. We use it everyday when messaging to our friends and posting on social media. Also, different emojis with the same sentence will convey different meanings or ideas. For example, “first day at school” means the school day starts. By adding a 😊, it means that I’m happy to go back to school and start the new term. By adding a 😞, it means that the unwillingness to go back to school. Our project Emojimotion is an application to predict the emoji based on both the sentence's meaning and user’s personal choice on eight different emotion types, including anger, anticipation, disgust, fear, joy, sadness, surprise and trust.

II. ILLUSTRATION

As shown in Figure 1, Emojimotion contains two parts. The first part takes sentences as input, passes a neural network model (baseline vs. BERT), and outputs a list of top 15 emojis with the highest probability by sorting the model output logits. The second part of Emojimotion combines the model outputs and the emotion score of the user’s choice of emotion and generates the final predictions. To combine the two numbers, the model output logits are passed through a softmax function and converted to context score. We extract the emotion score of each emoji in the top-15 list from the Emotag dataset and find the new emotion score by formula 1.

\[
\text{new emotion score}_i = \frac{\text{emotion score}_i}{\sum_{j=1}^{S} \text{emotion score}_j}
\]  

(1)

After that, equal weights are assigned to the two scores as in formula 2, and the application produces 3 emojis with highest possibility.

\[
\text{Final score}_j = 0.5 \times \text{context score}_j + 0.5 \times \text{new emotion score}_j
\]  

(2)

As the second part is pure mathematical calculations, the models and results discussed in the following sections are related to part 1 only.

III. BACKGROUND & RELATED WORK

As the rapid growth of emojis on social media, there are a lot of studies about emoji embeddings, sentiment analysis, and emoji predictions in recent years. We went through several papers and summarized 2 of which inspired us the most to create our EMOJIMOTION application.

In 2016, Eisner et al. [1] developed an emoji embedding model called Emoji2vec. It is a set of pre-trained embeddings of all Unicode emojis. Different from other pre-trained emoji vectors trained from a large number of tweets and Instagram posts, Emoji2vec is trained from the description of emojis. Besides training the word embeddings, this research also contains a simple sentiment analysis on tweets, which proves that the accuracy of the classification task can be improved by including the emoji in the sentences. The paper gave us our initial idea of exploring the emoji field.

In 2017, Francesco et al. [2] introduced Bi-LSTMs to predict the emojis. They compared the Bi-LSTM model with two baseline models: Bag of Words classifier and Skip-Gram Vector Average. The paper shows that the Bi-LSTM has the best performance among all the models in the experiment. The experiment result indicates that the frequent word is more likely to be mis-predicted as the model prefers to output the more frequent emoji which have similar meaning with some less frequent emoji. They also did a social experiment to let humans predict the emojis. Surprisingly, the Bi-LSTM achieved a higher accuracy.

IV. DATA AND DATA PROCESSING

Firstly, we collected realtime Twemoji using ranking from emojitracker API [3]. 54 emojis are randomly selected to be labels in which 20 emojis are from top 50 used emojis and the rest are from top 50-150. Raw tweet dataset containing ID, time, text and emoji was collected by implementing Twitter API [4]. The search requirement is collecting 5000 English tweets for each emoji, the text length is more than 10 and containing at least one target emoji.

Figure 1: Flow chart of Emojimotion

Figure 2: Selected 54 emojis

Figure 3: Tweet Dataset Before Clean
In the data cleaning stage, duplicated tweets are first removed from the raw dataset as some tweets are scraped down for multiple emojis. Noises in the sentence like hashtags, urls, mentions and unicodes are removed, because we think emoji should relate to the information conveyed by Tweet text only. Tweets after cleaning less than four words would be deleted since lower word count tweets contain more noise and we want to decrease the bias to model training. Figure 4 is one of the cleaned texts. We sampled 2000 cleaned tweets for each emoji to be final dataset and it is splitted into train/test/validation in the proportion of 0.64/0.2/0.16.

As shown in Figure 6, the open source EmoTag1200 database contains trained emotion scores for 1200 emojis under eight emotions ranging from 0 to 1. The symbolic emojis (ie. ▶️) not shown in this dataset were assigned zeros for all the emotions. The new emotion dataset of 54 emojis is used to combine with the context score predicted from the model to make final emoji predictions based on the user’s selection of emotion.

V. ARCHITECTURE AND SOFTWARE

The final neural network model architecture is BERT. BERT has demonstrated its effectiveness in natural language processing tasks, such as text classification and question answering[6]. Pretrained BERT is a transformer based architecture which learns the contextual information using a multi-layer bidirectional transformer encode. As in Figure 7, BertTokenizer is used to tokenize input tweets, and two special tokens [CLS] and [SEP] are added to the original input tokens which represent start and end of sentence respectively. The embedding input enters the BERT Model and classifier head accordingly which is stacked on top of the model.

We fine-tuned the bert-base-uncased that consists of 12 layers and 110M parameters with a classifier head by freezing the first 6 layers of the BERT model, training 5 epochs, and setting the learning rate to be 5e-5. The performance of the model is evaluated by the top-8 accuracy.
### TABLE II. BASELINE ARCHITECTURE

<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimizer</td>
<td>Adam</td>
</tr>
<tr>
<td>Learning Rate</td>
<td>0.0001</td>
</tr>
<tr>
<td>Number of Epochs</td>
<td>50</td>
</tr>
<tr>
<td>Batch Size</td>
<td>5</td>
</tr>
</tbody>
</table>

### VII. QUANTITATIVE RESULTS

Both top 1 accuracy (conventional accuracy) and top 8 accuracy are used to evaluate the models. By using top 8 accuracy, a prediction will be considered as correct if the true label from our datasets exists in the top 8 possible predictions. As one sentence could have multiple meanings and the model contains 54 classes, using top 8 accuracy is more reasonable. Therefore, top-8 accuracy is used as major evaluation metrics to evaluate model performances. We also measured the result by human examination, which will be illustrated in the qualitative result section.

The following tables show the quantitative results of the models. For 54 classes, the highest accuracy for both top 1 and top 8 is achieved when freezing the first 6 layers. If we only used 30 classes, the top 1 accuracy does not have a significant increase compared with the 54 classes model. Also, the goal of Emojimotion is to provide users with a variety of selections and an appropriate model to predict emojis. Therefore, the final selected model is the BERT model with a 54-class classifier head.

### TABLE III. 54 CLASSES(EMOJIS) MODEL RESULTS

<table>
<thead>
<tr>
<th>Model</th>
<th>Top 1 Accuracy</th>
<th>Top 8 Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>3%</td>
<td>24%</td>
</tr>
<tr>
<td>BERT</td>
<td>23.9%</td>
<td>56.3%</td>
</tr>
<tr>
<td>Freeze all layers – except the classifier head</td>
<td>18.6%</td>
<td>50.08%</td>
</tr>
<tr>
<td>Freeze the first 6 layers</td>
<td>24.6%</td>
<td>56.73%</td>
</tr>
</tbody>
</table>

### TABLE IV. 30 CLASSES(EMOJIS) MODEL RESULTS

<table>
<thead>
<tr>
<th>BERT</th>
<th>Top 1 Accuracy</th>
<th>Top 8 Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not freeze</td>
<td>3%</td>
<td>24%</td>
</tr>
<tr>
<td>Freeze all layers – except the classifier head</td>
<td>22.7%</td>
<td>60%</td>
</tr>
</tbody>
</table>

Besides the BERT Model, we also fine-tuned the smallest version of GPT2 which has 124M parameters. According to Figure 9 and 10, the BERT model with 56.3% accuracy outperforms the GPT2 model with 55% accuracy. Therefore, our main architecture is the BERT model and the following sections are based on BERT and baseline.

### VIII. QUALITATIVE RESULTS

The following figures show results from the baseline model and the best BERT model. For the baseline model with “Let’s grab a cup of coffee after class” and joy as input, the output contains happy face emojis. With sadness as input, the output contains emojis with negative emotion. For the BERT model with the same input sentence and joy as emotion, it outputs a coffee emoji and 2 happy faces. With sadness as emotion, it outputs coffee, a skull and a flushed face. Results for both models make sense, while the BERT model results are more reasonable, as it reflects precisely the sentence context with key word coffee and other generated emojis are more appropriate. One of the reasons for Bert good performance is due to the BERT model selection. In the progress report we used bert-tiny and the training accuracy could only reach 23%. In comparison, bert-base-uncased performed much better as this model has richer parameters good for sentence classification.
As shown in Figure 14, for some input sentences, the top 3 predicted emojis without emotion selection would stay the same after choosing different emotions. It is because these kinds of sentences have strong emotional implication and the context meaning is obvious, therefore the high context score would outweigh the emotion score and lead to the same results.

IX. DISCUSSION AND LEARNING

In this project, according to the results listed in the previous sections, the baseline model top-8 accuracy (24%) is higher than we expected since the largest dimension of GloVe Embedding was used in baseline which increased word vector complexity, therefore words could be represented more accurately. Our fine-tuned Bert model reached outstanding top-8 accuracy (56%) which is very high among 54-class classification. One of the surprising findings is that our Bert model could accurately identify different K-pop groups as shown in Figure 15. Each group has their own fan color (e.g., BTS in heart, Super Junior in heart) and tweets posted by fans usually contain fan color of heart emoji.

The future improvements would include some manual work. Firstly, current emojis are selected by sample() method based on setting criteria, so similar emojis in one grouping might all be selected, for example, 🌠 🌠 🌠 🌠 🌠 from our emoji labels all represent red hearts and have similar meaning. The future emoji class should be selected more diversely, so the input sentence could match more appropriate emoji.

Secondly, some data in the cleaned dataset like shown in Figure 14 are similar and repetitive. Most importantly, some words have no actual meanings and do not exist in English. These bad data are mostly ads which were usually posted at the same time, so according to the API timeline scraping, these tweets are collected for the same emoji class. Since they could not be filtered in the data processing, they would become the bias in the training and influence the testing accuracy. Therefore, this type of tweets require human sense to identify and delete in the future.

Thirdly, although the purpose of Emojimotion is to predict emoji based on context and emotion, the current trained Tweet dataset only contains text and emoji labels, and we manually assigned weights to combine the context score and emotion score. Consequently, the future tweet dataset could be completed by adding an emotion label. We could get more reasonable and accurate weights of score combination through training to further increase accuracy.

X. INDIVIDUAL CONTRIBUTIONS

We contributed equally to this project. Carol was responsible for scraping twitter data and building the baseline model. Yuchen was responsible for cleaning the raw dataset and building a Gradio interface. Both of us got involved in fine-tuning Bert model, training and testing the baseline model and Bert model.

REFERENCES


