SI630 Project Report: Emoji Interpretation on Twitter

Xinyi Wang

1 Abstract

Emojis are ideograms that are naturally combined with plain text to visually complement or condense the meaning of a message. Even though Emoji now becomes an essential element in social media messages, their underlying semantics have received little attention from a Natural Language Processing standpoint. When we performing data cleaning, emojis are usually treated as disturbances to be eliminated from the plain text. Analyzers can only decode limited connotations. To better understand the usage of emoji on social media, Twitter, I explored emoji interpretation by Bidirectional Long Short-term Memory Networks (BLSTMs). I believe that the research results help analyzers gain a deeper understanding of the content of the tweets and help people, especially people who rarely set step into social media, to understand what people are communicating.

2 Introduction

The advent of social media has brought along a novel way of communication where meaning is composed by combining short text messages and visual enhancements, the so-called emojis. This visual language is available not only on Twitter, but also on other large online platforms such as Facebook, Whatsapp, or Instagram. Despite its status as language form, emojis have been so far scarcely studied from a Natural Language Processing (NLP) standpoint. Therefore, I analyzed the contexts and determine their meanings in a large amount of Tweet context. Solving this problem is pretty meaningful. If the problem is solved, people can use this tool to predict what information could contain in these tweets and it is really useful to some foreigners and some kids who do not know many words to read the contents of tweets. Also, my methodology will be useful and other people may use our methodology to do classification tasks like this. In the project, I proposed a Bi-directional Long Short-term Memory Networks (BLSTMs) model to solve the prediction problem. I believe the model is robust and effective. The final model demonstrates the power of neural networks and artificial intelligence to help humans better understand the social media context.

3 NLP Task Definitions

Given the fact that emoji providing an additional channel for emotion convey, it is important to interpreter the meaning behind those visual icons. Since by now there are hundreds of different emojis, I first narrow down the scope that I only focused on the 20 most frequently used emojis, (listed in figure **??**). I analyze the textual context of Tweets as well as the emoji involved and determine the most likely associated emoji.

To be specific, I approached this task by training a neural classifier that takes a large corpus of Emoji-embedded tweets as input and outputs a list of predictions of Emoji that most fit the tweets. The output of the model is a probability measure of how confident the network is that the context can match an Emoji. The model defines the Emoji prediction with the highest possible as the Emoji to be embedded in the Tweets.

4 Data

I extracted English tweets from the Twitter API and used them as the data set. Given the fact that the meaning of emoji will change with the time and region, I sampled the tweets mainly posted from Jan.1 2019 to Jan.1, 2021, in the US.

Besides, for simplicity, I only chose the tweets that contained only one emoji. Tweets with one more emojis are removed from the database. Also, I made a further restriction on the emoji contained.

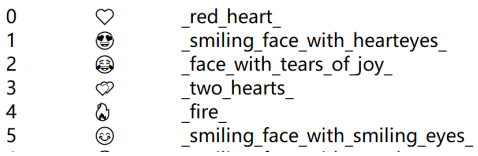

Figure 1: Mapping table between Emoji and word label

According to the usage frequency of emojis, only 20 most frequently-used emojis are calculated. In the end, I chose 40,000 satisfying data and split them into three sections: train(80%), dev(10%), and test(10%).

Data distribution: The emoji embedded in the tweets also follows the statistic result of Figure **??**, Joy and Hearts appears with high frequency. What I have not expected is that (1)the data is noisy since website links and unknown abbreviations are appearing in the tweets. (2)Most of the tweets are short and ambiguous, it is a bit hard to guess the meaning of each tweet, far from emoji. Therefore, further data cleaning was applied to those data to clear noise. In addition to the data noise, another problem of data preparation is emoji can't be processed by the model directly. So, a mapping between Emoji and label is conducted. Sample of mapping table between Emoji and word label is demonstrated in Figure **??** After data manipulation, the data can be processed by our model, sample of my data set is demonstrated:

1. Good Night Everyone $\_heart\_$ @ Trenton, New Jersey
2. 4 Miles, sunshine, and smiles. PalosVerdes I you. California Holidays @ Rolling Hills Estates,... $\_heart\_$
3. back at it $\_heart\_$ @ the University of North Carolina - Wilmington Campus Trask...
4. She only came for the nachos and cheese, that's why she's my big... $\_heart\_$
5. Happy birthday Madress can't wait for Led Zep tomorrow!!! $\_heart\_$ Here's a cool pic of you drinking...

4.1 Related Work

Researchers have put efforts to help NLP to get a better understanding of the Emoji interpretation on social media platforms. In the paper Multimodal Emoji Prediction, the authors extended recent advances in emoji prediction by putting forward a multimodal approach that can predict emojis in Instagram posts. Their model has consisted of two synergistic modalities which improved accuracy in an emoji prediction task(2018, Francesco Barbieri, Miguel Ballesteros, Francesco Ronzano, Horacio Saggion)

In the paper Seq2Emoji: A hybrid sequence generation model for short text emoji prediction, the author proposes a hybrid sequence generation model, Seq2Emoji, to predict multiple emojis based on a short text. The results showed that the model can learn semantics between the text and emoji labels, as well as the correlation between predicted emojis. (2021, Dunlu Peng, Huimin Zhao)

In the paper Short Text Classification in Twitter to Improve Information Filtering, the authors proposed an approach to address the problem of using traditional classification methods on short text data such as tweets. They proposed to use a small set of domain-specific features extracted from the author's profile and text. They used a greedy strategy to select the feature set, which generally follows the definitions of classes, extracting 8 features which consisted of one nominal (author) and seven binary features (presence of shortening of words and slangs, time-event phrases, opinioned words, emphasis on words, currency and percentage signs). This approach provided a baseline to classify new tweets online with better accuracy. (2010, Bharath Sriram, David Fury, Engin Demir, Hakan Ferhatosmanoglu)

In the article Prediction of Emoji from News Headlines using Machine Learning Techniques, the author used Recurrent Neural Networks to solve the emoji prediction problem in extremely short context –news headlines. (2019, Chloé Lagrue Chloé Lagrue.)

5 Methodology

All the codes can be found on my GitHub `https://github.com/wwwangxinyi/SI630_Final_Project`. To solve the problem, I separate it into several tasks: data preprocessing, word2vect embedding, sentence vectoring, and bi-directional LSTM model.

The main aim for data pre-processing is to transform the given data into trainable vectors. Since our data is drawn from the Twitter API, the data can not be processed by the model directly. Firstly, I applied a filter to find the Tweets with target Emoji. Then, I mapped the Emojis to the

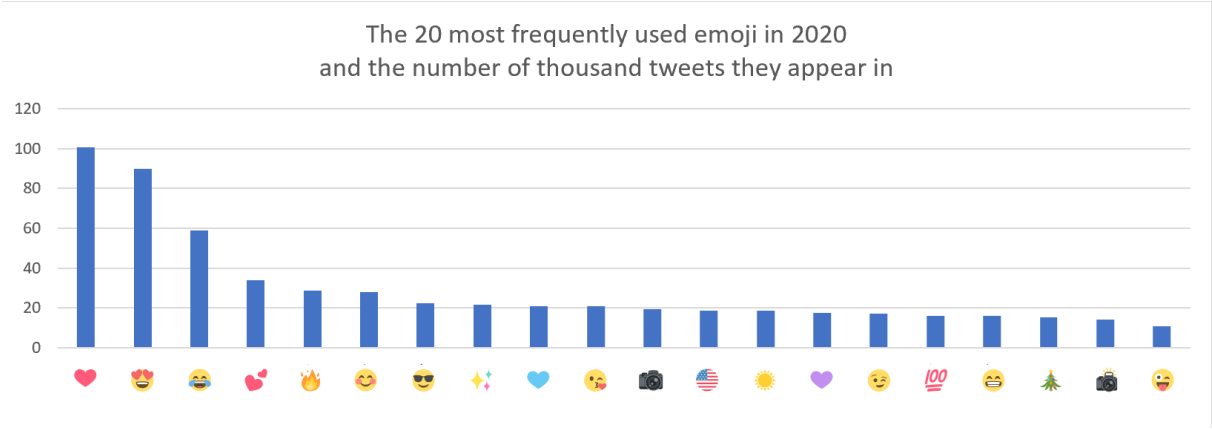

Figure 2: The 20 most frequently used emoji in 2020 and the number of thousand tweets they appear in

word label that can be recognized by the model through regular expression. Secondly, common data cleaning is applied to the data set. Frequent words(stop words), number, other symbol, URL links, and mentioned user name are excluded from the plain text since they don't provide too much meaning to the content.

In the word2vect process, the aim is to find a vector to represent words. For this part, I turned to google pre-trained model for help. With the help of the pre-trained Google word2vec model, the word is converted to a 300-dimension vector. This part is not included in the code since the model is quite large, to know more knowledge about the pre-trained wor2vec model, the official website can be referred to.

Finally, we applied a bi-directional LSTM model to solve the problem. The model structure is an embedding layer with output dimension 1, two bi-directional LSTM layers with output dimension 1, and one 1-dim vector to scalar dense layer. The embedding layer serves as the first layer of the whole model, which doesn't change the dimension of the input. The most important layer is the bi-directional LSTM layer, and we use the softmax function as activation. The dropping rate is set to follow suggestions from the website on the topic of applying LSTM to regression problems.

6 Evaluation

To validate the effect of my bi-directional LSTM model, a bag of words classifier is applied as a baseline algorithm, since it has been successfully employed in several classification tasks, like sentiment analysis and topic modeling. I represent each message with a vector of the most informative tokens selected by term frequency-inverse document frequency (TF-IDF). Then an L2-regularized logistic regression classifier is applied to make the predictions.

This validation experiment is a classification task, wherein each tweet the unique emoji is removed and used as a label for the entire tweet. I analyze the Bi-directional LSTM and Bag of Word model. For the evaluation of models, I performed the evaluation metrics: the overall precision(P), recall(R), and F1-score(F1). Table 1 demonstrates the prediction result of Bag of Words and bi-directional LSTM model. From the data. no matter how many emojis are involved in the classification tasks, the performance of the bi-directional LSTM model outperforms the baselines. The pre-trained vectors allow initializing the system with unsupervised pre-trained semantic knowledge, which improves the performance of the model. Also, we need to notice that with more emojis involved in the classification tasks, the performance of the model decreases.

In addition to the comparison with the baseline model, I invited several people to predict the fit Emoji. Since only three people(my friends and classmates) are involved in the test, only 300 tweets are selected to conduct the test. Also, the data set only contains the 5 most frequent emojis. The participants were asked to predict the emoji after they read a piece of a tweet without emoji. They could only be selected from the 5 emojis mentioned above. The prediction result can be seen in table 2. The result of the comparison between manual prediction and model prediction shows that the performance of the model surpasses the performance of humans in Emoji prediction.

7 Discussion

In Table 1, we can see that with more emoji involved in the classification task, the performance of the algorithm decreases. This may indicate that if we applied classification tasks on the top 20 emojis, the evaluation result will not be that satisfying, even though the algorithm is improved. This means that if an emoji is frequent, it is more likely to be on top of the possible choices even if it is not that fit. This result may be related to the frequency distribution of emoji in the original data set. If all the Emoji share the same frequency, the accuracy of Emoji prediction is likely to increase.

On the other hand, there are some emojis sharing overlapped semantics that causes difficulties for the model in classifying. It leads to the model make decisions with bias. The model prefers to choose the one with a higher frequency. For example, there are four hearts are involved in the data set, $\_red\_heart\_$, $\_two\_hearts\_$, $\_blue\_heart\_$, and $\_purple\_heart\_$. Among the four hearts, the model will prefer to $\_red\_heart\_$, instead of the other three hearts. The only context under certain additional circumstances, such as Partisan competition, other hearts will be preferred.

Also, need to be figured out that the model is not that robust. Since the model is built based on Tweets that only contained single Emoji, the performance of Emoji prediction is limited to a single meaning. If the model can be trained on Tweets with multiple Emojis, the prediction will be more effective and closer to human knowledge.

Last but not least, even though the performance of the model on Emoji prediction has surpassed the manual prediction, the prediction result is still not that satisfactory. One result of this problem may have resulted from the shortness of Tweet text. The motion or meaning is conveyed limited, which provides few clues for the model to determine. Also, some Tweets in the data set don't have any concrete meaning, leading to the confusion of Emoji determination.

8 Conclusion

In this project, my goal was to use neural network methods to predict the Emoji used in the Tweets that provides a tool for people to understand the context used in social media. While my method did not produce a satisfactory result, I believe that the results are still significant. My method shows that the convolutional neural networks can predict the context for Emojis on social media without requiring extensive prepossessing and feature engineering. I also demonstrated that social media context data, specifically from Twitter, can be utilized to make a neural network classifier.

9 Other Things I Tried

In my early experiments, I tried using Naive Bayes to approach this task. Even though Naive Bayes can drive Emoji prediction results, their performance is much lower than the one of Bi LSTM. Especially when the model is making predictions for those Emojis with relatively low frequency. I guessed that there may be too many categories for the Naive Bayes model to classify.

10 What I Would Have Done Differently or Next

As is stated before, the model is not that robust, it can not annotate the Tweets with multiple Emojis embedded. I would like to alter the model and take the Tweets with multiple Emojis as the input. In addition, I would have also liked to experiment with different methods of generating word embeddings, including trying different numbers of dimensions and using pre-trained word2vec, GLOVE, or FastText vectors.

11 References

[1] Short Text Classification in Twitter to Improve Information Filtering, Bharath Sriram, David Fuhry, Engin Demir, Hakan Ferhatosmanoglu. Proceeding SIGIR '10 Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval: 841-842

Sida Wang and Christopher D. Manning, 2012, Baselines and Bigrams: Simple, Good Sentiment and Topic Classification, Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics:9094

Peng, D., Zhao, H. (2021). Seq2Emoji: A hybrid sequence generation model for short text emoji prediction. Knowledge-Based Systems, 214, 106727. https://doi.org/10.1016/j.knosys.2020.106727

Prediction of Emoji from News Headlines using Machine Learning Techniques. (2019). International Journal of Recent Technology and Engineering, 8(4), 10321–10324. https://doi.org/10.35940/ijrte.d4549.118419

	5 Emojis			10 Emojis			20 Emojis		
	P	R	F1	P	R	F1	P	R	F1
BOW	0.52	0.58	0.56	0.45	0.46	0.45	0.30	0.34	0.28
BLSTM + P	0.64	0.63	0.63	0.46	0.47	0.47	0.41	0.38	0.33

Table 1: Results of 5,10,20 emojis. Precision, Recall, F-measure. BOW represents Bag of Word, BLSTM + P represent the model with a pretrained embedding

	Human			Bi-LSTM		
	P	R	F1	P	R	F1
_red_heart_	0.70	0.59	0.64	0.73	0.82	0.76
_smiling_face_with_hearteyes_	0.54	0.51	0.52	0.61	0.62	0.61
_face_with_tears_of_joy_	0.50	0.45	0.47	0.62	0.45	0.54
_two_hearts_	0.19	0.26	0.26	0.52	0.30	0.39
fire	0.24	0.26	0.25	0.63	0.51	0.56

Table 2: Precision, Recall and F-Measure of human evaluation and B-LSTM prediction for the 5 most frequent emojis