ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed
Computing

Emoji for Natural Language Processing

Master's Thesis

Wang Yuqiao
yuqwang@student.ethz.ch

Distributed Computing Group
Computer Engineering and Networks Laboratory
ETH Zürich

Supervisors:
Wang Ye, Meng Zhao
Prof. Dr. Roger Wattenhofer

May 25, 2021

Abstract

Nowadays, emoji has become one new global language, which can be understood by anyone in the world no matter what the mother tongue is. Emoji can help people express their emotions more intuitively in communication. Beyond communication in daily life, some people are trying to use emoji in lifetime. In this thesis, we are trying to solving NLP task with emoji language. We trained model in order to predict the suitable emojis for a given text. Moreover, since each text could express several different emotion, our model also learned to predict different emojis given different sentiments for one single text.

Contents

Chapter 1

Introduction

Emojis are defined as “digital images" which have been added into our electronic daily communication texts. As a significant supplement in a visual message, it has been widely used in plenty of social media platforms and comjunication tools, such as twitter, facebook, whatsapp and many other forms of CMC (Computer-Mediated Communication)[1]. It is reported that almost half of the posts on Instagram contain emojis[2], and moreover, emojis are currently replacing emoticons on Twitter and becoming a popular way of representing things, feelings, concepts and so on[3]. In addition, studies in sociology and psychology show that emojis has become a new language in this new generation[4]. Therefore, emoji has become a valuable research direction for artificial intelligence in social media domain.

The intentions of using emojis in CMC have been discussed for so many years, and emojis are supposed to express sentiment, express humor, and strengthen the expression for people in daily life[5]. Since they are digital graphics instead of combinations of punctuation marks, emojis are obviously more various and vivid, and can express more complicated information compared to many other expression ways. In addition, in 2001, Walther et al.[6] studied sentiment impact on text expression in 2001, showing that positive emojis increase the positivity of positive verbal messages, but negative emojis do not increase the negativity of negative messages. And the result of studying the sentiment impacts of more types of emoticons in various social media environments has also been reported in 2008[5]. Therefore, we can see that sentiment expression is also a significant aspect in emoji research, and more detailed, the sentiments of emojis and verbal texts have some potential relationship and it is worthy for studying.

Since it is apparently that using glyph information in logographic languages helps to solve NLP tasks, there should be a mapping between natural language and emoji. It is intuitive that emoji should help us to solve NLP tasks in any language. In this paper, all tasks we have solved were based on natural language processing.We firstly trained bart model to predict suitable emojis for a given single text. And in the second part of our experiment, we mainly focused on how the sentiment of a given single text could influence the prediction of emojis from our trained model.

Chapter 2

Related Work

2.1 A Brief History of Visual Expression

Multiple forms of visual expression, such as emoticon, emoji, stickers, and memes, have developed for many years along with the development of Internet. Since 1982, U.S. professor Scott Fallman at Carnegie Mellon University used several symbols to make up the first visual expression of emotion ASCII characters ":-)". Inspired by this symbol, people started to use various keyboards arbitrarily symbols (such as English alphabets, punctuation marks, and numbers) to form the emoticon to vividly express facial expressions. Nowadays, the forms of visual expression also developed largely. Figure2.1 shows various kind of visual expression (a, emoticon; b, emoji; c, stickers; d, meme), and examples frequently used in China social media platforms.

With the appearance of more and more creative tools for people communication, visual expressions have evolved with various new forms, and more and more emojis have been released in these recent years. Nowadays, emoji could be classified into multiple types, such as people, animals, food, symbol and so on. The emergence of these new types of emoji are also related with the development of human culture.

2.2 Studies on Emoji

Previous work on emojis mainly focuses on three research directions: the meanings and sentiments of emojis, as well as the different usages of emojis among people. The first attempt of studying the usage of emojis with a word embedding approach is in 2015, which is carried by Dimson[2]. In this project, the authors transformed the emojis occurred in Instagram posts to vectors, and used the semantically closest words to the emojis in the vector space as their explanations. In 2016, Eisner et al.[7] also proposed an embedding model, which is

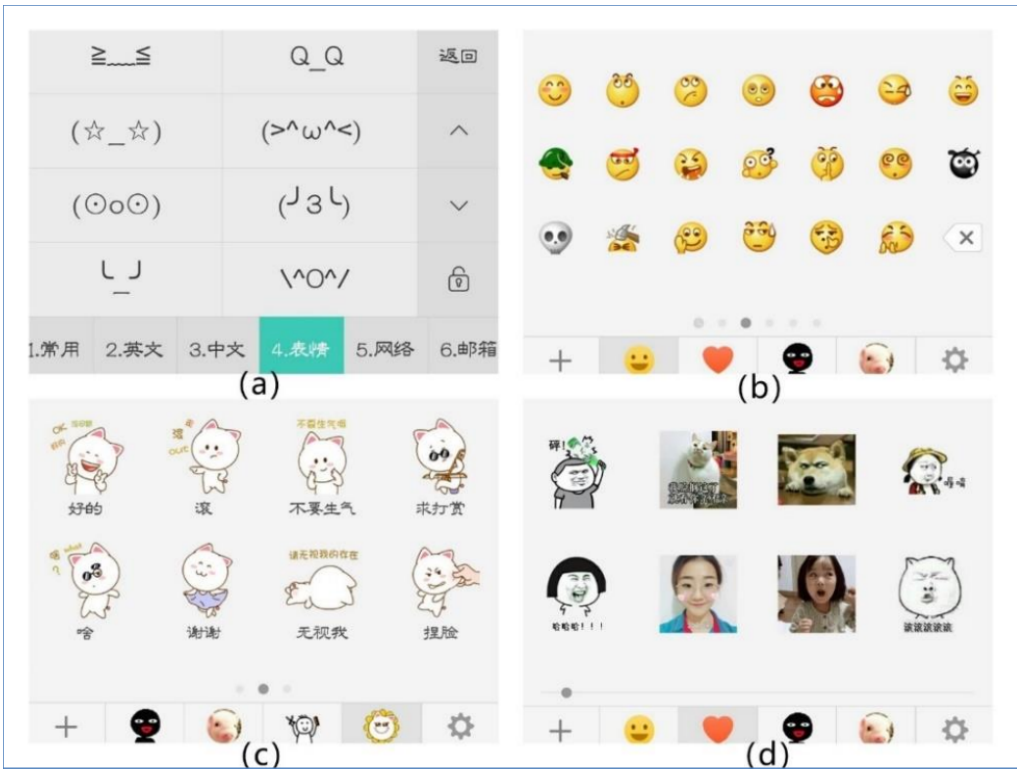


Figure 2.1: Four types of visual expression on Chinese social media: (a) emoticon; (b) emoji; (c) stickers; (d) meme.

called Emoji2vec for learning emoji representations. Instead of learning the emoji vectors from social media posts, they used the emoji descriptions, and obtained better performances on evaluation tasks. Similarly, Wijeratne et al.[8] built up a machine readable sense inventory for emoji by aggregating the descriptions of emojis from multiple online sources. In terms of the sentiments of emojis, Novak et al[9] set up a sentiment dictionary for plenty of emojis. In this paper, authors required participants to label the sentiment of texts that contain at least one emoji. The sentiment score of an emoji was then computed as the average scores of all the messages where this particular emoji occurred. They reported that most emojis are positive, and the emojis that are more used are more emotionally loaded. Tauch et al.[10] carried the research about the sentiment effects on mobile phone notifications of duplicate emojis. In this work, they found that when the number of emojis was high, the sentiment of the whole message was not related to the text content. According to Miller et al.[11], the sentiments and interpretations of emojis may vary from person to person. In this paper, the authors reported the most differently interpreted emojis. The different usages of emojis also are also related to the differences among people groups. It is suggested by Barbieri er al. [12]that, different countries have their specific preferences to emojis, and in 2016, Lu et al.[13] reported that such preferences imply the cultural and regional difference of the countries in many aspects.

2.3 Intentions of Using Emojis

By reviewing the previous work on the intentions of using nonverbal cues as their surrogates in communication, we summarized these intentions and their explanations from previous work as follows:

- Expressing sentiment[14][10]: same as facial expressions during face-to-face communication, emoji usage in order to express sentiments or emotions, such as anger, happiness, fear, and so on.

- Strengthening expression[6]: using emojis to strengthen their expression, for instance, adding an angry face emoji by the end of a text in order to express a more negative sentiment.

- Adjusting tone: using emojis to adjust tone, for instance, adding emojis in a sentence in an informal occasion in order to make other feel less serious.

- Expressing humor: using emojis to make communication more interesting.

- Expressing irony[15]: using emojis to make communication more sarcastic or ironic.

- Expressing intimacy: using emoji in order to shorten the distance between user and readers, making a closer relationship.

- Describing content[16][17]: using emojis to describe content in text, for example, using a heart emoji in order to express love.

Here, we can find that sentiment is a significant ingredient in emoji usage during daily communication, especially in social media platform. Therefore, sentiment studying during emoji research is valuable and it has become one of the cores in our project.

2.4 Sentiment Effects of Emoji

In terms of psycology, scientists have been focusing on the sentiment effects of emojis for many years. In 2001, Walther et al.[6] carried research on the sentiment impacts of emoticons. In this paper, they tried to study emojis and plain verbal messages together for the first time. They studied the impacts of positive and negative emoticons on positive and negative verbal messages. In addition, it is reported that positive emoticons increase the positive emotion of positive verbal messages, but negative emoticon do not increase the negativity of negative messages. With the same approach, Derks et al.[5] studied the sentiment influence of more types of emojis in various social contexts and reported similar results. By applying similar methods, the authors also studied the influences of emojis on identical perception[18] and the effects of emojis during task-based communication[19].

Chapter 3

Model

. In this project, we used Bart as our basic model. But before we discuss the details about model architecture, we also need to introduce the basic structure of Bart.

3.1 Sequence to Sequence Model

When we talked about transformer, it is actually an advanced version of sequence to sequence model(Seq2Seq)[20]. Seq2Seq is a variant of recurrent neural network which includes two parts: encoder system and decoder system. Seq2Seq is an significant RNN model in natural language processing, which can be used for machine translation, dialogue systems, and automatic summarization.

Seq2Seq could be considered as an N*M model. In this structure, encoder could be used for encoding given sequence information into any length of vector C. And after going through an encoder, we can obtain the final output sequence by feeding vector C into the following decoder.

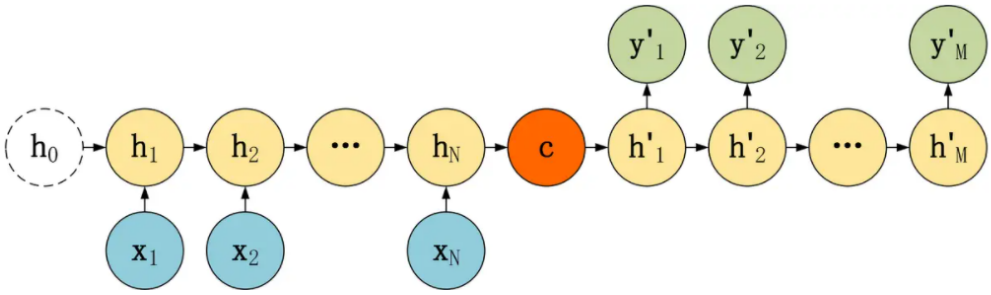


Figure 3.1: Seq2Seq Model Structure

3.1.1 Encoder

The structure of encoder in Seq2Seq model could be found in Figure 3.1. In this RNN cell, X means the given text which is fed into encoder, and we finally get the output vector c which contains the whole information in this input sequence. This vector would be the main input to the following RNN cell for decoding into the output sequence. Generally, the vector c could be computed in several different ways:

$$c = h_N \tag{3.1}$$

$$c = q(h_N) \tag{3.2}$$

$$c = q(h_1, h_2, ..., h_N) \tag{3.3}$$

According to the above equations, we can see that the vector c could not only be directly used as the final hidden state h_N but also be generated from the transformation with previous hidden states.

3.1.2 Decoder

In the simplest decoder system, vector c would be considered as the first hidden state and fed into this decoder RNN cell. Subsequently, it only accepts the hidden layer state $h^{'}$ of the previous neuron and does not accept other input x. Here are the computation of each hidden layer state and the output y:

$$h_1^{'} = \sigma(Wc + b) \tag{3.4}$$

$$h_t^{'} = \sigma(Wh_{t-1}^{'} + b) \tag{3.5}$$

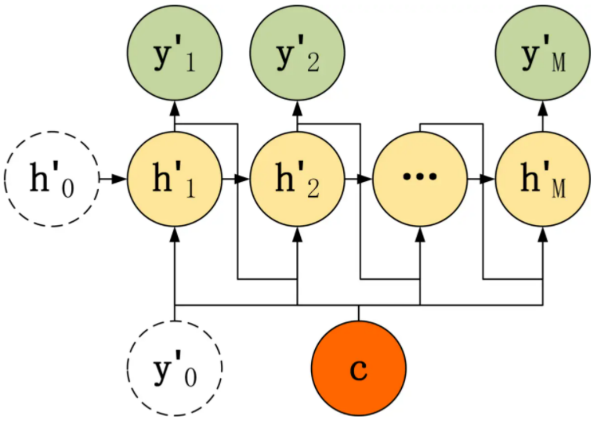


Figure 3.2: Decoder structure with target input

$$y_t^{'} = \sigma(Wh_t^{'} + b) \tag{3.6}$$

However, in general cases, we usually use a more complicated structure as the decoder system which has been shown in Figure 3.2.

In this case, the output $y^{'}$ of the previous neuron is added for generation. So now the input of each neuron includes: the hidden layer state $h^{'}$ of the previous neuron, the output $y^{'}$ of the previous neuron, and the current input c (the context vector encoded by the Encoder). For the input $y_0^{'}$ of the first neuron, it is usually the embedding vector of the starting flag of the sentence. The computation of hidden layer states and the output have been shown below:

$$h_t^{'} = \sigma(Uc + Wh_{t-1}^{'} + Vy_{t-1}^{'} + b) \tag{3.7}$$

$$y_t^{'} = \sigma(Vh_t^{'} + c) \tag{3.8}$$

3.1.3 Teacher Forcing

Teacher forcing is generally used during training stage. The input of the neuron of the second Decoder model includes the output $y^{'}$ of the previous neuron. If the output of the previous neuron is wrong, the output of the next neuron is also easy to be wrong, which results in the error being passed on all the time.

But when teacher forcing has been introduced, we would have a kind of orthodontics, which means that compared of feeding the output $y^{'}$ to the next neuron, we choose the correct target instead, and this would largely correct our prediction to the most precise result.

3.2 Transformer

Nowadays, transformer has become one of the most popular model structure for NLP task. Traditional CNN and RNN structures are abandoned in transformer, and the entire network architecture is entirely composed of attention mechanism. More precisely, transformer only consists of self-attention structure and feed forward Neural Network. A trainable neural network based on transformer[21] can be built by stacking transformer structures together. The details about transformer structure has been shown in Figure 3.3

3.2.1 Encoder and decoder structure

The encoder is composed of a stack of $N = 6$ identical layers. Each layer has two sub-layers. The first is a multi-head self-attention mechanism, and the second is a simple, position-wise fully connected feed-forward network.

The decoder is also composed of a stack of N = 6 identical layers. In addition to the two sub-layers in each encoder layer, the decoder inserts a third sub-layer, which performs multi-head attention over the output of the encoder stack.

3.2.2 Self-Attention

Self-Attention is the core of transformer. In self-attention, each word has 3 different vectors. They are query vector(Q), key vector(K) and value vector(V). Each of them is obtained by multiplying the embedding vector by three different weight matrices W^Q, W^K and W^V, and the dimensions of the three matrices are also the same.

So how does self-attention work? First of all, we convert the input word into an embedding vector and we can obtain three vectors q, k and v as the instruction above. For each embedding vector, we compute a score for itself and with other vectors, and the score of which is calculated as $score = q \cdot k$. Next, we feed these normalized scores into Softmax layer and multiply the output by vector v in order to obtain the value v for each input word. The final feature vector for this particular input word is computed as $z : z = \sum v$. There is the complete equation for self-attention mechanism:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{3.9}$$

3.2.3 Multi-Head Attention

Multi-Head Attention is equivalent to an integration of h different self-attentions. It concatenates h Zs and passes it into a fully-connected layer with final output Z. In transformer, it is multi-head attention that is used in both encoder and decoder.

3.3 Bart Model

For models based on transformer, Bart model[22] is particularly good at handling sequence generation tasks. It uses a standard neural machine translation architecture based on transformer, which can be regarded as a generalized form of pre-trained models such as BERT (bidirectional encoder) and GPT (left-to-right decoder).

3.3.1 Bart Architecture

Bart is a standard Seq2Seq model which combines bidirectional and auto-regressive transformers. The detail of structure has been shown in Figure 3.4.

In this architecture, we can see that inputs to the encoder need not be aligned with decoder outputs, allowing arbitrary noise transformations. Since Bart allows us to apply any type of document corruption, that means Bart training mainly consists of two steps: (1) Use arbitrary noise functions to destroy the text (2)

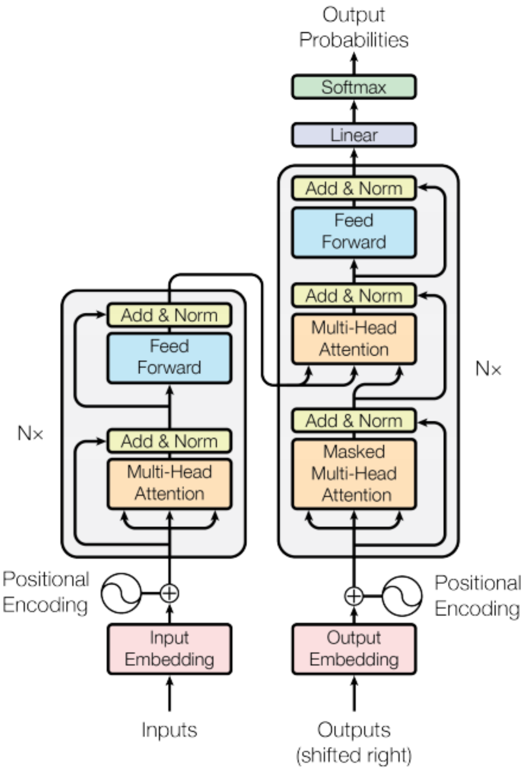


Figure 3.3: transformer

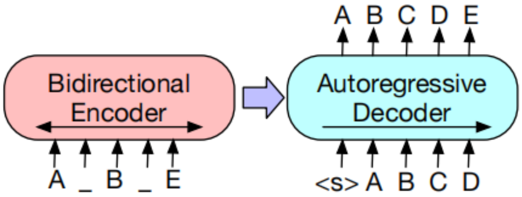


Figure 3.4: bart

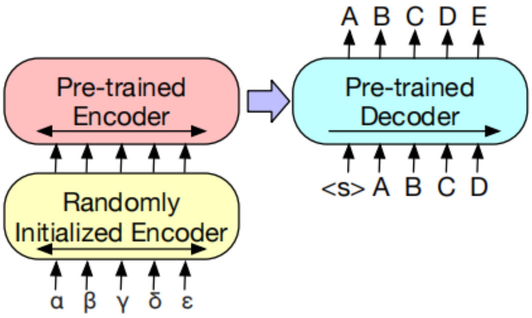


Figure 3.5: machine translation

Train model to reconstruct the original text. Here are several previously proposed and novel transformations for document corruption: token masking, token deletion, text infilling, sentence permutation and document rotation.[22]

3.3.2 Fine-Tune of Bart

The representations produced by BART can be used in several ways for downstream applications. In this project, we used Bart model for machine translation task, which means we translate 'language' without emojis into 'language' with emojis. The basic way we do fine-tune for this task has been shown in Figure 3.5. For machine translation task, we do fine-tune in two steps. First of all, we freeze most of BART parameters and only update the randomly initialized source encoder, the BART positional embedding and the self-attention matrix of the first layer in encoder. Next we train all model parameters for a small number of iterations.[22]

3.3.3 Bart Tokenizer

There is 50265 tokens in the original BartTokenizer. However, this is not enough for our model training, since we have 1276 emojis needs to be added into tokenizer. But we did not directly add these original emojis into bart tokenizer, but firstly transform each emoji into a short English description and enclose it into '[]'. This is the way our emoji token shows during training process. So we added these 1276 new emoji tokens into bart tokenizer and now the length of our tokenizer should be 51541.

Chapter 4

Dataset for Experiment

4.1 Data Pre-process

The dataset we used for model training is tweet with emojis from twitter.

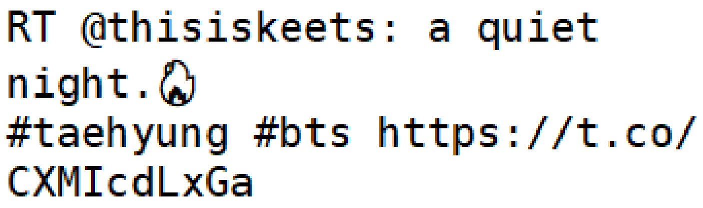
4.1.1 Twitter Crawler

Since our goal of this task is to translate text without emojis into text with emojis, our dataset needs to contain large amount of texts with various emojis. So we chose twitter as the research communication platform and did tweet crawling through Twitter API. In this case, we chose English as our research language and we collected 3,500,000 tweets as the initial dataset.

Next, since not each tweet contains emojis and this kind of data is useless for our task, we extracted 400,000 tweets from the initial dataset where each data contains at least one emoji.

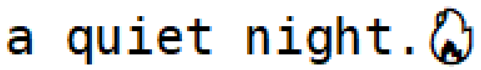
4.1.2 Data Refine

If you used twitter previously, you can find that some tweets contains tags with '#' and associated link with 'http'. Moreover, when people re-post one tweet, this re-post tweet would start with 'RT' and followed by the id name of the original tweet. In addition, some people usually mention their friends when they post one tweet with '@'. In this case, all these extra information is useless since we only concern about the original information users wish to express. So for each tweet in this dataset, we have to remove all these extra ingredients.



```
RT @thisiskeets: a quiet
night.🔥
#taehyung #bts https://t.co/
CXMIcdLxGa
```

Figure 4.1: tweets with useless information



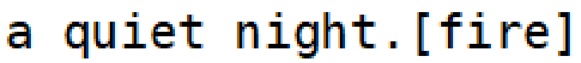
```
a quiet night.🔥
```

Figure 4.2: tweets after extraction

For instance, Figure 4.1 shows one of the tweets we collected. Here we can see that this tweet was re-posted from id named '@thisiskeets' and it added two tags called '#taehyung' and '#bts'. A link also followed by the end of this tweet. So we need to remove all these ingredients. The final version of this data should be like this in Figure 4.2.

Now we need to decide which is the training data and which is the target. Since this is a machine translation task, our target should be the text with emojis. Therefore, we filtered out all emojis in each text and these data after filtering should be used as our training data and the original data with emojis should be the target.

Finally, we still need to process the emoji expression. Since we need to make sure each emoji is readable during training process, we solve this problem like what we did for bart tokenizer: we transform each emoji into an English description, which could be find on emoji official website. In addition, we used '[]' to enclose each emoji description. Taking the example above, if our training data looks like in Figure 4.2, then its target data should be like in Figure 4.3.



```
a quiet night.[fire]
```

Figure 4.3: example of target

4.2 Sentiment Score

During the second part of our experiment, our goal is to give different prediction when we set the input text into different emotions. For example, when we type text "Time to work.", one possibility is that you wish to express you are energetic and ready for work which is a positive emotion. However, another possibility could be like "I haven't had enough rest yet.", which is definitely a negative emotion. In this case, the emojis our model predicts should absolutely be different. That's what our model needs to learn–for a single text, the emoji prediction should be adapted according to what kind of emotion users wish to express. Therefore, the sentiment score for each data is necessary during training process.

However, when we observed dataset in details, we can find that many data contains more than one sentence. This might result in mix sentiment, which means one text contains both positive and negative sentiment even though the sentiment of whole text is neutral, and it would influence our model training result. To eliminate this negative influence, we need to separate all data into single sentence and we also dropped those single sentences without emojis. Now the final dataset has been successfully generated. For sentiment analysis, we use pre-trained Bert model and the sentiment score is a value between 0 and 1, where 0 refers extremely negative and 1 refers extremely positive. Moreover, when we analyze data distribution for all sentiment scores, we find in Figure 4.5 that most scores are near 0 or 1 and Figure 4.4 shows that the ratio of value below 0.5 and above 0.5 is 47.3%:52.7%, which is almost 1:1. Therefore, in order to simplify our task, we did binarization for all sentiment score, which means all scores below 0.5 could be regarded as 0 and others could be regarded as 1.

After obtaining the sentiment score for each single text in training data, we still need to set sentiment score for each emoji, since we need to figure out which emojis are frequently used when users want to express negative emotion and which are for positive usage. Therefore, we did this job. For each emoji, we extracted all texts in training dataset which contain this particular emoji. Then we calculate the average of sentiment score of these extracted texts and this average score would be set for this emoji as its own sentiment score.

4.3 Token

Like we just said above, we have already added 1276 new emoji tokens into bart tokenizer for training. However, this is not enough.During the second part of our experiment, we need to add sentiment score as a special signal for each training data, we have to add special tokens $< v >$ and $< /v >$ in tokenizer so that the

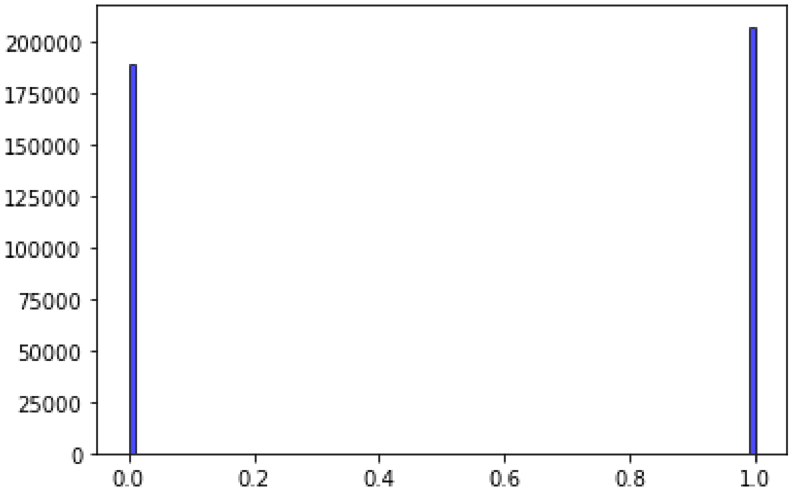


Figure 4.4: sentiment ratio after binarization

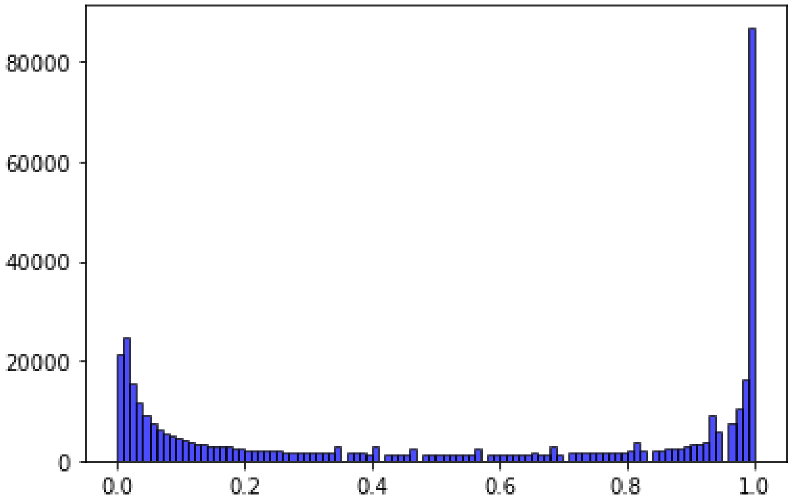


Figure 4.5: sentiment data distribution

model could make a difference between score signel and original sentence. Our training data would be fed into model as type like "$< v >$ + sentiment score + $< /v >$ + $< s >$ + text + $< /s >$". Therefore, our final BartTokenizer should be in length of 51543.

Chapter 5

Experiment

Now let's see how we carried experiment for this task. Our experiment contains two parts. The first one is about training model for emoji prediction. The second one contributes to give different predictions when we locate one single text with different emotions.

5.1 Basic Model Train

In the first part of our experiment, our task is to simply return one prediction for a single given text. Now we have 400,000 tweets with emojis, we splited it into two subsets, one with 80% data was used as training dataset and the rest as validation dataset. All parameters were set as table 5.1.

parameter	value
criterion	CrossEntropyLoss
optimizer	Adam
learning rate	1e-4
max length	50
batch size	32
epoch	10

Table 5.1: Training Parameter

Now we use tensorboard to draw each curve during training process, which is shown in figure 5.1.

In this figure, blue curve represents training loss and red one represents the validation loss. We can see that in epoch No.2 the validation loss decreased to optimal value as 15.0.

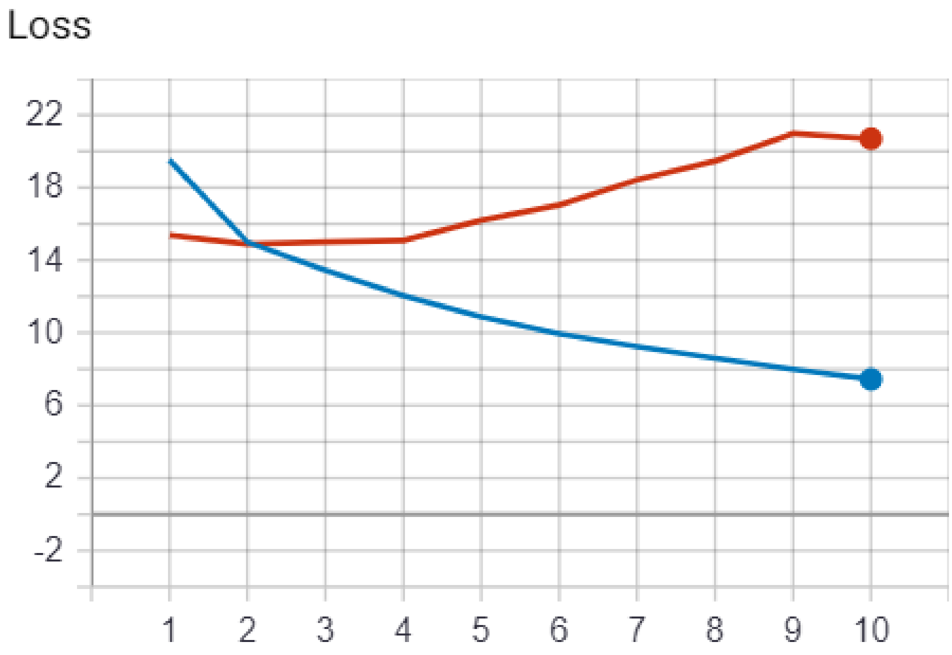


Figure 5.1: Training curves in Experiment 1

During translation task of Natrual Language Processing, bleu score is a significant index to evaluate how well our model performed[23]. Here tensorboard also summarized bleu score for each epoch during training process, which is shown in figure 5.2.

In this figure, we can see that bleu score come to optimal value as 69.76 in epoch 10.

Now let's see some examples about our first part of experiment in different types. Figure 5.3 shows us four texts and their single predictions. In the first sentence, it said "did a week of work in 4 hours ", and our model returned emoji 'Pleading Face' at the end as its final prediction. In the second example, the emoji prediction is apparently positive. Moreover, in the third text, we can see that more than one emojis have been predicted, which means our model would not only return single one emoji but several different emojis as well. In the last instance, this is a long text and we can see that the predicted emoji was inserted in the middle of the sentence, which means our model did not just simply add emoji at the end of text but insert into a more suitable position.

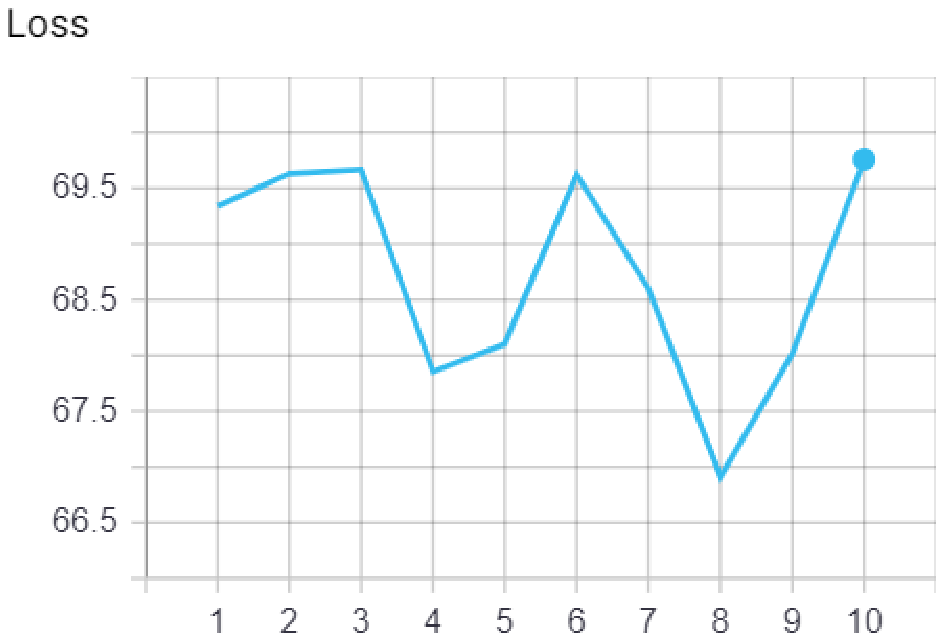
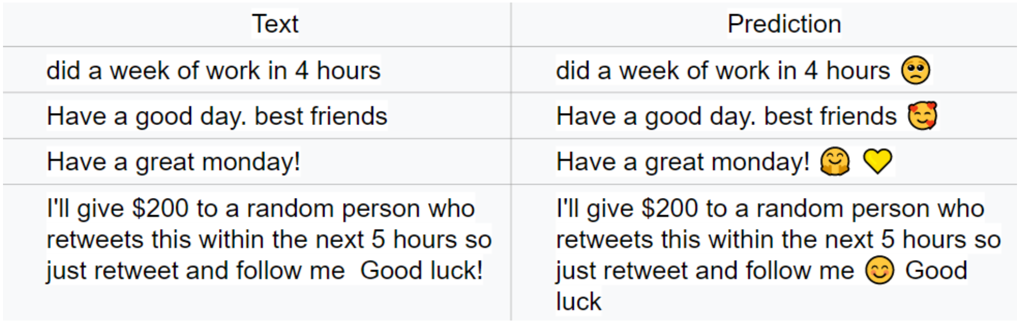


Figure 5.2: Bleu Score in Experiment 1



Text	Prediction
did a week of work in 4 hours	did a week of work in 4 hours 🥺
Have a good day. best friends	Have a good day. best friends 🥰
Have a great monday!	Have a great monday! 🤗 💛
I'll give $200 to a random person who retweets this within the next 5 hours so just retweet and follow me Good luck!	I'll give $200 to a random person who retweets this within the next 5 hours so just retweet and follow me 😊 Good luck

Figure 5.3: Examples in Experiment 1

5.2 Model Train with Sentiment Analysis

From the previous part, we can obtain one suitable emoji prediction given a single text. However, let's think about a question: what does "suitable" mean for an emoji? As we all know, the use of emoji largely depends on what kind of emotion we wish to express. For instance, when you are given a text "A new week for school", you might want to express “Another energetic week!” or you may just want to say "I don't want to go to school...". So without given a sentiment user wants to express, we can not say which emoji is really suitable for your intention. Therefore, we got our motivation for the next part of experiment. We need to train a model which could return different emojis for different sentiment when a single text was given and our model should also return more than one prediction for either sentiment, that means we can give users more options.

Now we have the sentiment score for each text, we added this score in front of its corresponding text as a new signal and fed these new sequences into model for a new train. In this part, we also have two indexes summarized on tensorboard to evaluate our model's performance.

In figure 5.4, we can see that in epoch 3 the validation loss decreased to optimal value as 6.989. And in figure 5.5, we can see that bleu score come to optimal value as 66.54 at epoch 9.

In addition, we added a new metric this time called 'hit rate', which means we can calculate how many test data has the same predicted emojis as its targets. The result shows that we got the highest hit rate at epoch 9 as 26.12%, which could meet our expectation.

Negative	Positive
Face with Steam From Nose	Face with Tears of Joy
Pensive Face	Beaming Face with Smiling Eyes
Face with Rolling Eyes	Sparkles
Skull	Smiling Face with Smiling Eyes
Neutral Face	Cat with Tears of Joy

Table 5.2: Emojis with Different Sentiment

Like we discussed just now, we wish to return more than one emoji prediction for each text and it is users to decide which emoji should be used given all prediction results. So during the test process, we chose to return 5 sequences and printed all results together. Here are some examples of our prediction result in figure 5.6.

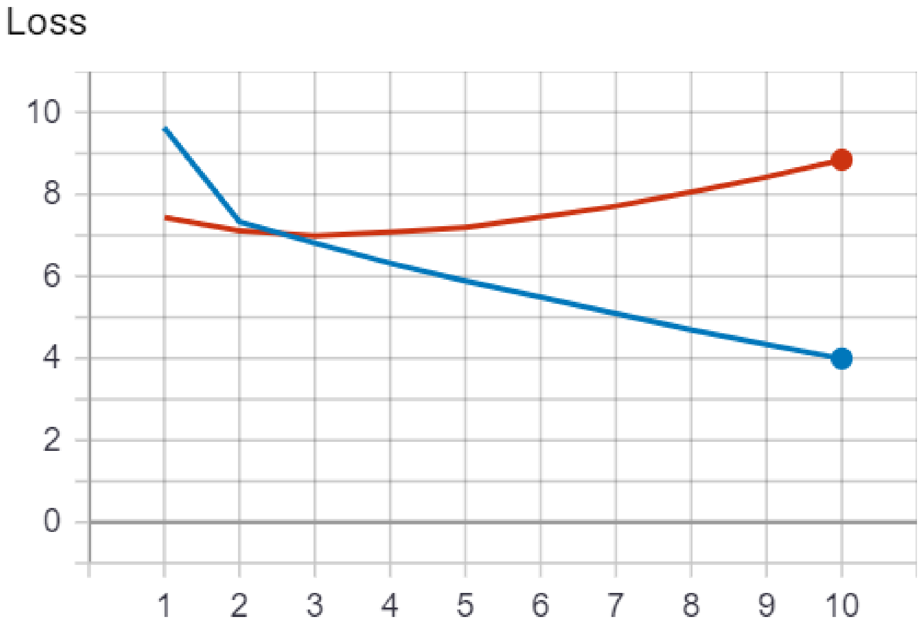


Figure 5.4: Training curve in Experiment 2

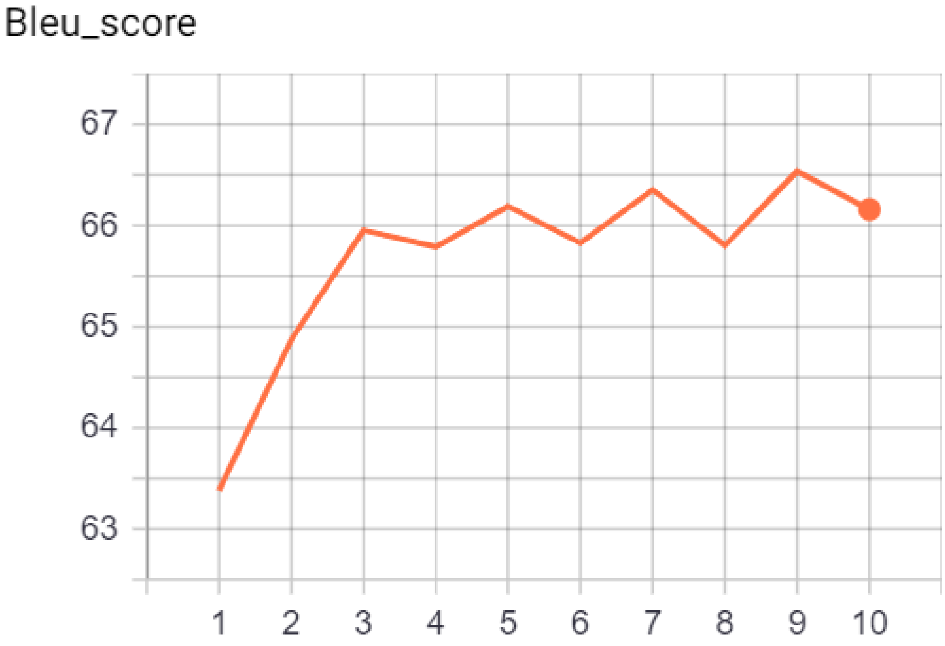
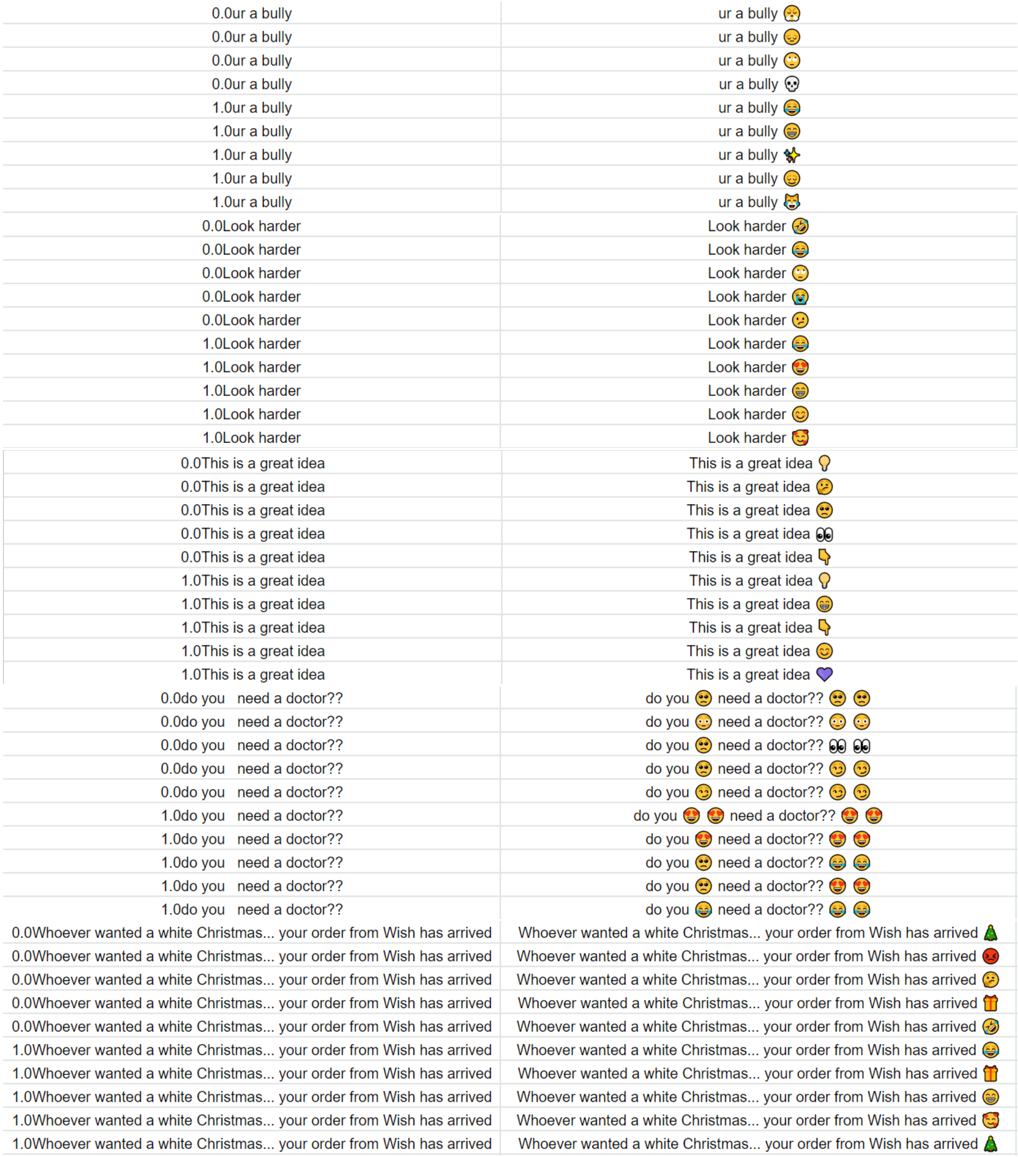


Figure 5.5: Bleu Score in Experiment 2



0.0ur a bully	ur a bully 😫
0.0ur a bully	ur a bully 😔
0.0ur a bully	ur a bully 🙄
0.0ur a bully	ur a bully 💀
1.0ur a bully	ur a bully 😂
1.0ur a bully	ur a bully 😁
1.0ur a bully	ur a bully ✨
1.0ur a bully	ur a bully 😔
1.0ur a bully	ur a bully 😹
0.0Look harder	Look harder 🤣
0.0Look harder	Look harder 😂
0.0Look harder	Look harder 🙄
0.0Look harder	Look harder 😭
0.0Look harder	Look harder 😕
1.0Look harder	Look harder 😂
1.0Look harder	Look harder 😍
1.0Look harder	Look harder 😁
1.0Look harder	Look harder 😊
1.0Look harder	Look harder 🥰
0.0This is a great idea	This is a great idea 💡
0.0This is a great idea	This is a great idea 🤔
0.0This is a great idea	This is a great idea 😟
0.0This is a great idea	This is a great idea 👀
0.0This is a great idea	This is a great idea 👇
1.0This is a great idea	This is a great idea 💡
1.0This is a great idea	This is a great idea 😁
1.0This is a great idea	This is a great idea 👇
1.0This is a great idea	This is a great idea 😊
1.0This is a great idea	This is a great idea 💜
0.0do you need a doctor??	do you 😟 need a doctor?? 😟 😟
0.0do you need a doctor??	do you 😳 need a doctor?? 😳 😳
0.0do you need a doctor??	do you 😟 need a doctor?? 👀 👀
0.0do you need a doctor??	do you 😟 need a doctor?? 😏 😏
0.0do you need a doctor??	do you 😏 need a doctor?? 😏 😏
1.0do you need a doctor??	do you 😍 😍 need a doctor?? 😍 😍
1.0do you need a doctor??	do you 😍 need a doctor?? 😍 😍
1.0do you need a doctor??	do you 😟 need a doctor?? 😂 😂
1.0do you need a doctor??	do you 😟 need a doctor?? 😍 😍
1.0do you need a doctor??	do you 😂 need a doctor?? 😂 😂
0.0Whoever wanted a white Christmas... your order from Wish has arrived	Whoever wanted a white Christmas... your order from Wish has arrived 🎄
0.0Whoever wanted a white Christmas... your order from Wish has arrived	Whoever wanted a white Christmas... your order from Wish has arrived 😡
0.0Whoever wanted a white Christmas... your order from Wish has arrived	Whoever wanted a white Christmas... your order from Wish has arrived 🤔
0.0Whoever wanted a white Christmas... your order from Wish has arrived	Whoever wanted a white Christmas... your order from Wish has arrived 🎁
0.0Whoever wanted a white Christmas... your order from Wish has arrived	Whoever wanted a white Christmas... your order from Wish has arrived 🤣
1.0Whoever wanted a white Christmas... your order from Wish has arrived	Whoever wanted a white Christmas... your order from Wish has arrived 😂
1.0Whoever wanted a white Christmas... your order from Wish has arrived	Whoever wanted a white Christmas... your order from Wish has arrived 🎁
1.0Whoever wanted a white Christmas... your order from Wish has arrived	Whoever wanted a white Christmas... your order from Wish has arrived 😁
1.0Whoever wanted a white Christmas... your order from Wish has arrived	Whoever wanted a white Christmas... your order from Wish has arrived 🥰
1.0Whoever wanted a white Christmas... your order from Wish has arrived	Whoever wanted a white Christmas... your order from Wish has arrived 🎄

Figure 5.6: Examples in the Second Part of Experiment

Negative	Positive
0.15005359056806	0.76894356318118
0.360831136610327	0.804027504911591
0.328744896891029	0.825450072682545
0.118490694894997	0.865902499654743
0.20361601377529	0.683168316831683

Table 5.3: Sentiment Score for each emoji

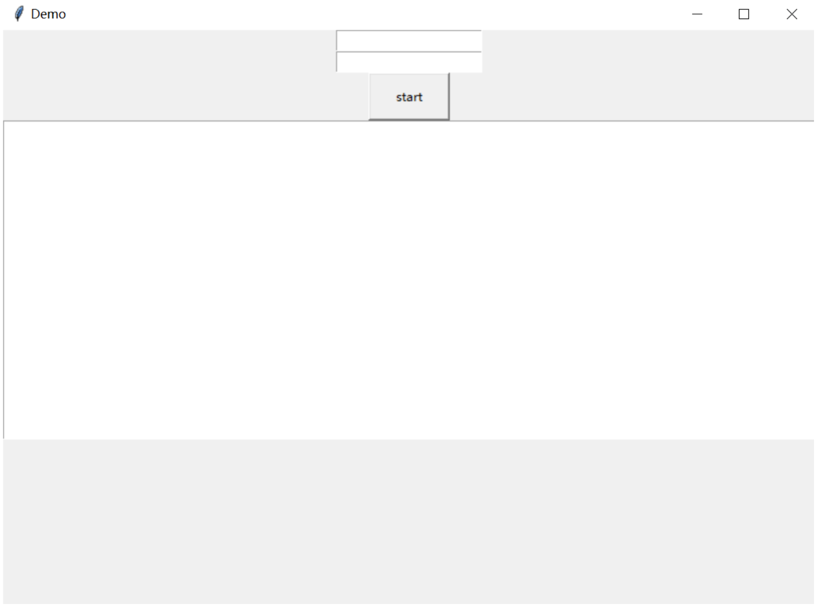
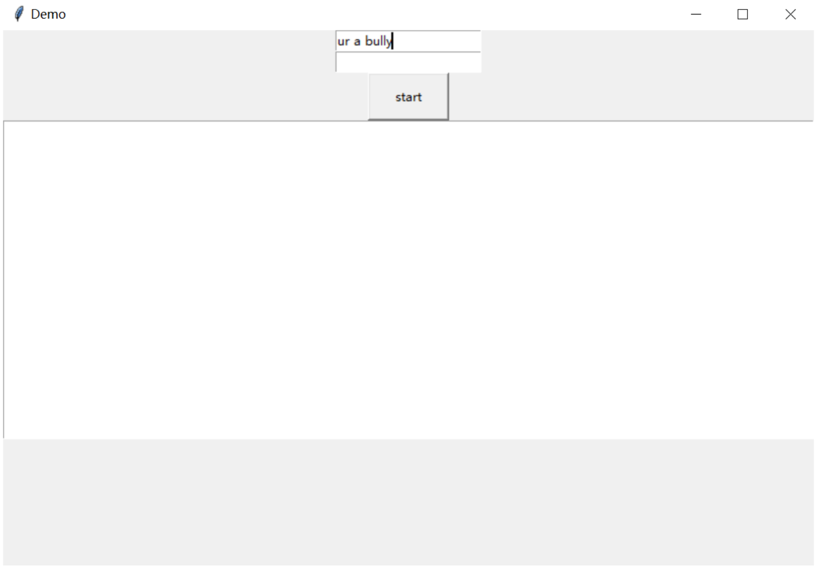


Figure 5.7: demo

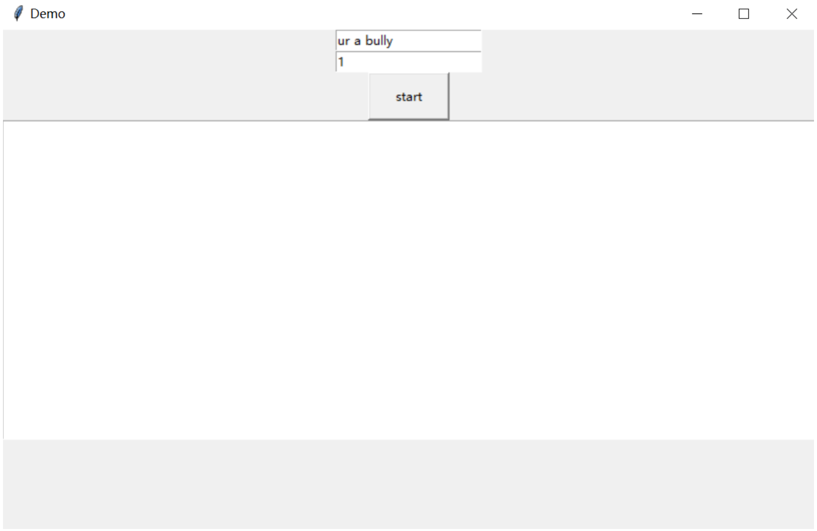
Let's have a look at one simple example. When a user entered a sentence "ur a bully", one of the possibilities of what he wishes to express is that "You are such a bastard!", which is really angry. However, he can also express a pampering attitude. So in this case, we added 0/1 in front of the text and return 5 predictions for each sentiment type. The table 5.2 shows the list of five prediction results for both negative and positive emotions. Remember the sentiment score for each emoji we calculated previously? The sentiment score of each emoji could be checked below in table 5.3(The complete version could be checked by the end of this chapter). As we checked sentiment score for each emoji we predicted in this table, we can find that the scores of most predicted emojis for negative sentiment are all below 0.5 and those of positive emojis are mostly above 0.5. Moreover, for all test data set to negative sentiment, we also computed the average score of all predicted emojis and we did the same job for all test data set to positive sentiment. And since the negative average score is 0.41, which is lower than 0.5, and positive one is 0.64, which is higher than 0.5, the result greatly proved that the emoji predictions of negative and positive sentiment are different and matching.

Finally, in order to make our result more clearly and readable, we made an interface as a demo which looks like figure 5.7.



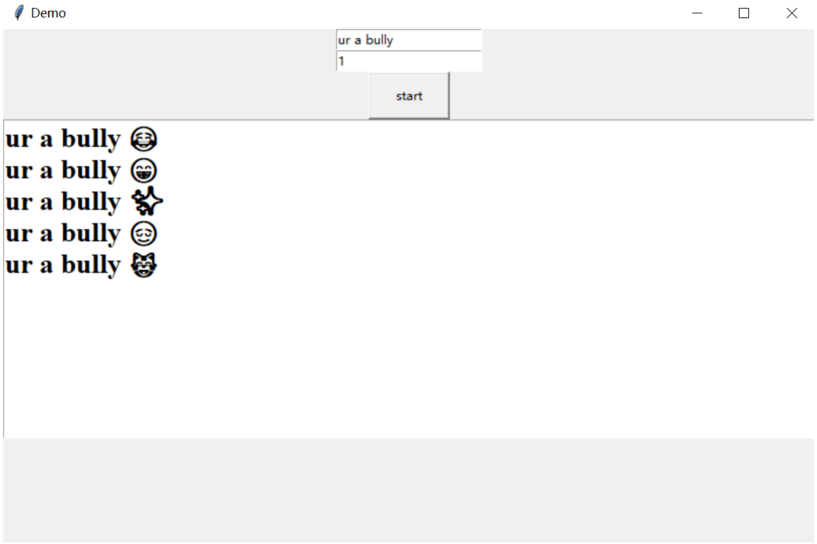
ur a bully

Figure 5.8: type into text



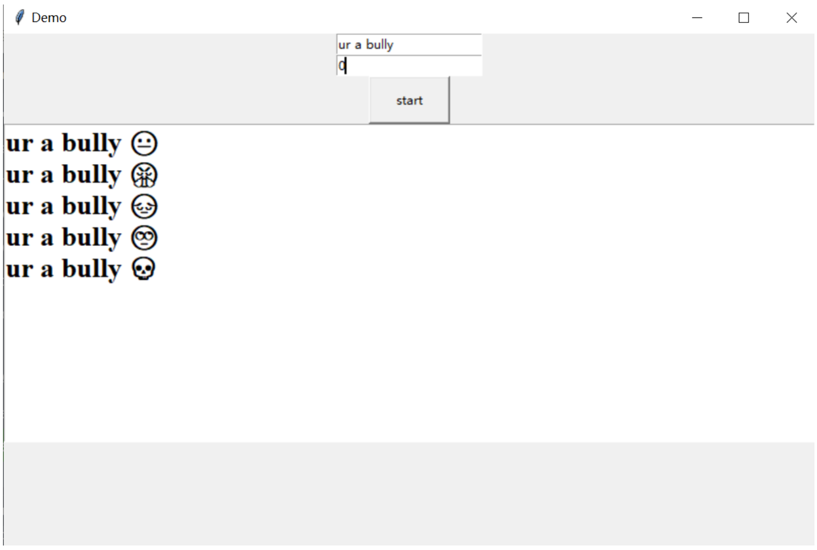
ur a bully
1

Figure 5.9: set sentiment to positive or negative



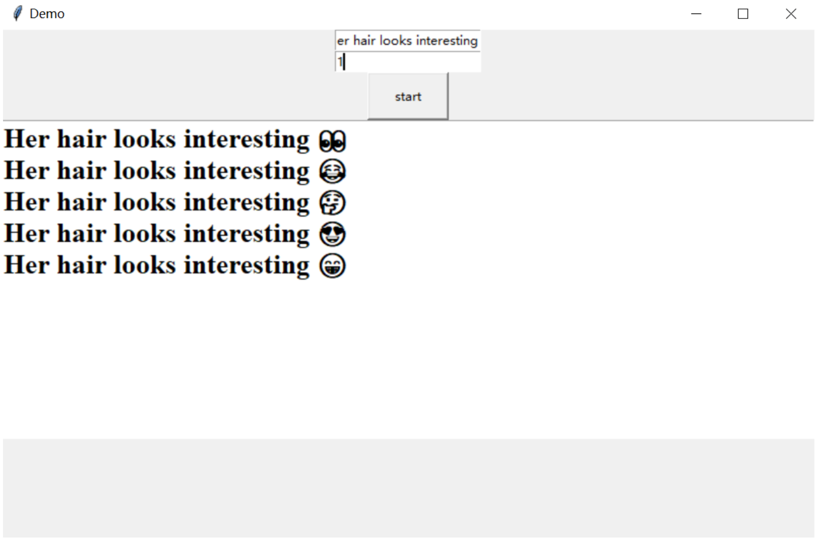
```
ur a bully 😂
ur a bully 😁
ur a bully ✨
ur a bully 😌
ur a bully 😹
```

Figure 5.10: positive example of demo



```
ur a bully 😐
ur a bully 😫
ur a bully 😔
ur a bully 🙄
ur a bully 💀
```

Figure 5.11: negative example of demo

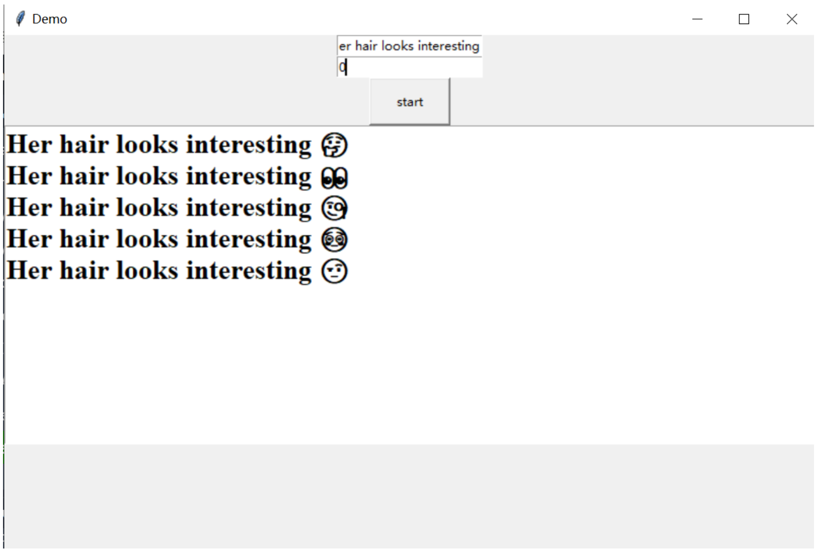


er hair looks interesting

1

Her hair looks interesting 👀
Her hair looks interesting 😂
Her hair looks interesting 🤤
Her hair looks interesting 😍
Her hair looks interesting 😁

Figure 5.12: second positive example of demo



er hair looks interesting

0

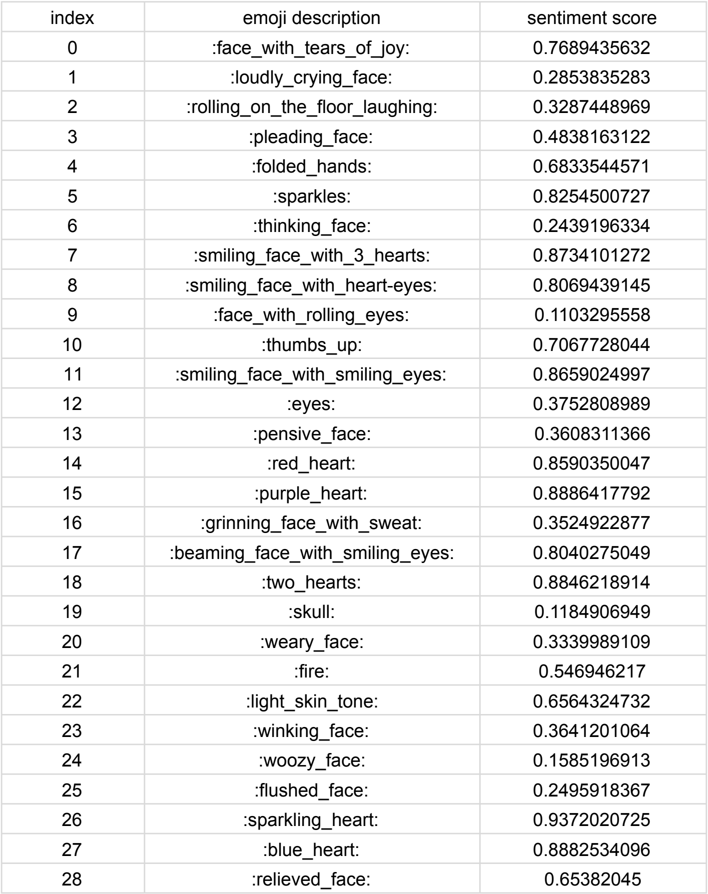
Her hair looks interesting 🤤
Her hair looks interesting 👀
Her hair looks interesting 😋
Her hair looks interesting 😳
Her hair looks interesting 🤨

Figure 5.13: second negative example of demo

In this demo, we type our input text in the upper text box as figure 5.8, and set our sentiment score in the text box below as figure 5.9. And the five predictions of our model would be shown on the screen after pressing 'start', which is shown in figure. 5.10

In this case, we set sentiment score to 1, which is positive, but we can also set it to 0, which is negative emotion we wish to express. This time, we can see in figure 5.11 that the results are totally different from that previously.

Moreover, we can also see another example. If we type 'Her hair looks interesting', it can mean that her hair is beautiful, or her hair looks pretty strange, which is a negative meaning. So when you set sentiment score to 1, which is positive, then the predicted emojis are also looks positive in figure 5.12. However, when we changed sentiment score to 0, which is negative, at least 3 predicted emojis also changed and obviously it looks negative as well in figure 5.13.



index	emoji description	sentiment score
0	:face_with_tears_of_joy:	0.7689435632
1	:loudly_crying_face:	0.2853835283
2	:rolling_on_the_floor_laughing:	0.3287448969
3	:pleading_face:	0.4838163122
4	:folded_hands:	0.6833544571
5	:sparkles:	0.8254500727
6	:thinking_face:	0.2439196334
7	:smiling_face_with_3_hearts:	0.8734101272
8	:smiling_face_with_heart-eyes:	0.8069439145
9	:face_with_rolling_eyes:	0.1103295558
10	:thumbs_up:	0.7067728044
11	:smiling_face_with_smiling_eyes:	0.8659024997
12	:eyes:	0.3752808989
13	:pensive_face:	0.3608311366
14	:red_heart:	0.8590350047
15	:purple_heart:	0.8886417792
16	:grinning_face_with_sweat:	0.3524922877
17	:beaming_face_with_smiling_eyes:	0.8040275049
18	:two_hearts:	0.8846218914
19	:skull:	0.1184906949
20	:weary_face:	0.3339989109
21	:fire:	0.546946217
22	:light_skin_tone:	0.6564324732
23	:winking_face:	0.3641201064
24	:woozy_face:	0.1585196913
25	:flushed_face:	0.2495918367
26	:sparkling_heart:	0.9372020725
27	:blue_heart:	0.8882534096
28	:relieved_face:	0.65382045

29	:growing_heart:	0.9209166479
30	:medium_light_skin_tone:	0.5992141454
31	:medium_skin_tone:	0.5426020408
32	:medium_dark_skin_tone:	0.558988764
33	:smiling_face_with_sunglasses:	0.5653939886
34	:broken_heart:	0.4560829242
35	:clapping_hands:	0.7765598651
36	:upside-down_face:	0.2906131938
37	:hundred_points:	0.5247465713
38	:face_blowing_a_kiss:	0.7926980198
39	:grinning_squinting_face:	0.3608824498
40	:raising_hands:	0.7237936772
41	:hugging_face:	0.867592279
42	:slightly_smiling_face:	0.5392651297
43	:star-struck:	0.6830780452
44	:regional_indicator_symbol_letter_u:	0.637829912
45	:grimacing_face:	0.1108639229
46	:smirking_face:	0.262012012
47	:clown_face:	0.1749611198
48	:black_heart:	0.7657657658
49	:green_heart:	0.8943217666
50	:flexed_biceps:	0.681136543
51	:grinning_face:	0.5434326479
52	:unamused_face:	0.1465770685
53	:regional_indicator_symbol_letter_s:	0.6397463002
54	:neutral_face:	0.2036160138
55	:OK_hand:	0.6414359862
56	:backhand_index_pointing_right:	0.5048118985
57	:backhand_index_pointing_down:	0.5143868969
58	:yellow_heart:	0.884494382

59	:crying_face:	0.4675385334
60	:white_heart:	0.8799608035
61	:grinning_face_with_big_eyes:	0.6058502727
62	:pouting_face:	0.08345827086
63	:revolving_hearts:	0.9052896725
64	:zany_face:	0.290238837
65	:partying_face:	0.5676100629
66	:grinning_face_with_smiling_eyes:	0.6358320043
67	:face_savoring_food:	0.3915598291
68	:face_with_steam_from_nose:	0.1500535906
69	:smiling_face_with_horns:	0.3502227171
70	:party_popper:	0.8245717661
71	:tired_face:	0.2504451039
72	:raised_fist:	0.6201318155
73	:jack-o-lantern:	0.6056511057
74	:backhand_index_pointing_left:	0.4193548387
75	:crossed_fingers:	0.4416826004
76	:beating_heart:	0.9182754183
77	:winking_face_with_tongue:	0.1592271819
78	:face_with_hand_over_mouth:	0.171043771
79	:face_with_raised_eyebrow:	0.1445448228
80	:musical_notes:	0.7883672039
81	:victory_hand:	0.6261549396
82	:see-no-evil_monkey:	0.3193456615
83	:sneezing_face:	0.2635135135
84	:nauseated_face:	0.01439393939
85	:face_screaming_in_fear:	0.2741935484
86	:disappointed_face:	0.14375
87	:face_with_symbols_on_mouth:	0.1572327044
88	:face_vomiting:	0.05411954766

89	:confused_face:	0.1731890092
90	:smiling_face_with_halo:	0.7447873228
91	:sleepy_face:	0.4383445946
92	:drooling_face:	0.1597892888
93	:face_with_monocle:	0.1524229075
94	:face_with_medical_mask:	0.3081032947
95	:exploding_head:	0.1120144535
96	:expressionless_face:	0.1306715064
97	:orange_heart:	0.8652416357
98	:pile_of_poo:	0.08108108108
99	:oncoming_fist:	0.5867298578
100	:cat_face_with_tears_of_joy:	0.6831683168

Chapter 6

Conclusion

In this paper, we carried experiments about natural language processing task with emoji language. We firstly trained a model in order to simply give an emoji prediction with a single input text. In addition, we tried to train a model to give different emoji predictions according to the emotion users wish to express with a single text. Finally, we constructed a demo of our model for visualization. After analyzing our result, we can conclude that our trained model can successfully return suitable emoji given a single sentence and moreover it can predict different and suitable emojis depending on different sentiment or emotion users wish to express.

In the future, our experiments should not only focus on twitter environment. There are plenty of social communication platforms around the world and the use of emoji is totally different due to the variety of culture among countries. Therefore, it is also worth to do the research on other social media environments, such as Weibo in China. Moreover, we can also focus on some other fields about emoji, since the habits of using emoji also depend on gender and age and so on.

Bibliography

[1] E. Dresner and S. C. Herring, "Functions of the nonverbal in cmc: Emoticons and illocutionary force," *Communication theory*, vol. 20, no. 3, pp. 249–268, 2010.

[2] T. Dimson, "Emojineering part 1: Machine learning for emoji trends," *Instagram Engineering Blog*, vol. 30, 2015.

[3] U. Pavalanathan and J. Eisenstein, "Emoticons vs. emojis on twitter: A causal inference approach," *arXiv preprint arXiv:1510.08480*, 2015.

[4] H. Alshenqeeti, "Are emojis creating a new or old visual language for new generations? a socio-semiotic study," *Advances in Language and Literary Studies*, vol. 7, no. 6, 2016.

[5] D. Derks, A. E. Bos, and J. Von Grumbkow, "Emoticons in computer-mediated communication: Social motives and social context," *Cyberpsychology & behavior*, vol. 11, no. 1, pp. 99–101, 2008.

[6] J. B. Walther and K. P. D'addario, "The impacts of emoticons on message interpretation in computer-mediated communication," *Social science computer review*, vol. 19, no. 3, pp. 324–347, 2001.

[7] B. Eisner, T. Rocktäschel, I. Augenstein, M. Bošnjak, and S. Riedel, "emoji2vec: Learning emoji representations from their description," *arXiv preprint arXiv:1609.08359*, 2016.

[8] S. Wijeratne, L. Balasuriya, A. Sheth, and D. Doran, "Emojinet: Building a machine readable sense inventory for emoji," in *International conference on social informatics.* Springer, 2016, pp. 527–541.

[9] P. Kralj Novak, J. Smailović, B. Sluban, and I. Mozetič, "Sentiment of emojis," *PloS one*, vol. 10, no. 12, p. e0144296, 2015.

[10] C. Tauch and E. Kanjo, "The roles of emojis in mobile phone notifications," in *Proceedings of the 2016 acm international joint conference on pervasive and ubiquitous computing: Adjunct*, 2016, pp. 1560–1565.

[11] H. Miller, J. Thebault-Spieker, S. Chang, I. Johnson, L. Terveen, and B. Hecht, ""blissfully happy" or "ready tofight": Varying interpretations of emoji," in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 10, no. 1, 2016.

[12] F. Barbieri, G. Kruszewski, F. Ronzano, and H. Saggion, "How cosmopolitan are emojis? exploring emojis usage and meaning over different languages with distributional semantics," in *Proceedings of the 24th ACM international conference on Multimedia*, 2016, pp. 531–535.

[13] X. Lu, W. Ai, X. Liu, Q. Li, N. Wang, G. Huang, and Q. Mei, "Learning from the ubiquitous language: an empirical analysis of emoji usage of smartphone users," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2016, pp. 770–780.

[14] S.-K. Lo, "The nonverbal communication functions of emoticons in computer-mediated communication," *Cyberpsychology & behavior*, vol. 11, no. 5, pp. 595–597, 2008.

[15] R. Filik, A. Ţurcan, D. Thompson, N. Harvey, H. Davies, and A. Turner, "Sarcasm and emoticons: Comprehension and emotional impact," *Quarterly Journal of Experimental Psychology*, vol. 69, no. 11, pp. 2130–2146, 2016.

[16] H. Cramer, P. de Juan, and J. Tetreault, "Sender-intended functions of emojis in us messaging," in *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services*, 2016, pp. 504–509.

[17] R. Kelly and L. Watts, "Characterising the inventive appropriation of emoji as relationally meaningful in mediated close personal relationships," *Experiences of technology appropriation: Unanticipated users, usage, circumstances, and design*, vol. 2, 2015.

[18] T. Ganster, S. C. Eimler, and N. C. Krämer, "Same same but different!? the differential influence of smilies and emoticons on person perception," *Cyberpsychology, Behavior, and Social Networking*, vol. 15, no. 4, pp. 226–230, 2012.

[19] T. T. Luor, L.-l. Wu, H.-P. Lu, and Y.-H. Tao, "The effect of emoticons in simplex and complex task-oriented communication: An empirical study of instant messaging," *Computers in Human Behavior*, vol. 26, no. 5, pp. 889–895, 2010.

[20] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *arXiv preprint arXiv:1409.3215*, 2014.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[22] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence

pre-training for natural language generation, translation, and comprehension," *arXiv preprint arXiv:1910.13461*, 2019.

[23] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

Appendix A

First Appendix Chapter Title