Hello everyone. We are Cheng Lin, Lau Cho Han and Qiao Shuyu. Today, we’ll be presenting our project. The topic of our project is Tweet Text Extraction Based on Sentiment.\\

We’ll cover the following sections in our presentation. Introduction, Data description, Exploratory data analysis, methods&models, conclusion and future works.\\

Firstly, we’ll introduce our project.\\

Twitter is one of the most popular SNS that people use nowadays. But it’s difficult to evaluate whether the sentiment of a sentence will have an impact or not. Evaluating the sentiment of sentences is therefore essential.

However, we do not know which word is used for sentiment description. In general, the sentiment of a sentence can be classified as positive, negative, and neutral. Some words or phrases are extracted from a sentence to reflect its sentiment.

In this project, we extracted part of the tweets, which are used for evaluating the sentiment. We used several models to train our data, including Natural Language ToolKit (nltk), SpaCy, and A Robustly Optimized BERT Pretraining Approach (in short RoBERTa).\\

There are plenty of previous studies that addressed the techniques of text extraction task.\\

Next, I’ll talk about data description.\\

The dataset consists of 27,481 rows and 4 columns.

**textID** is the unique ID for each piece of text

**text** is the content of the tweet

**Sentiment** is the general sentiment of the tweet, including 3 types: neutral, negative and positive

**selected\_text** is a subtext of text that supports the tweet's sentiment

Our goal is to extract key text from text as selected-text that can express its sentiment

In this project, we removed the missing values, and divided the original data into two datasets, in the ratio of 8:2 as the new training data and the new test data separately. \\

In our subsequent experiments, the evaluation index is Jaccard similarity. It is basically a formula for measuring how much overlap there is between A and B. It can be easily visualized using venn diagrams as shown on the right hand side. We’ll call it Jaccard score.

In our model evaluation, we calculate the Jaccard similarity score between prediction and ground truth. The higher the score, the higher the similarity, the more accurate our model predicts.\\

Next, I’ll talk about Exploratory Data Analysis.\\

In the exploration of data, we make sure that there are no missing values, and count the numbers of tweets posted that imply different sentiments. Based on the jaccard scores for each target tweets, we check for the similarity between selected and the original texts. We also find the most common words in the tweets.   \\

So first, it's clear to see from this funnel chart of sentiment distribution, neutral tweets takes up the most position.

In addition, we explored the distribution of meta-features. The distribution of the number of words includes both words counted from the selected text and the original text. The histogram is right-skewed, meaning that most tweets have less than 25 words.\\

To compare the Jaccard scores for these three sentiments together, we contribute the kernel distribution of Jaccard scores across all three sentiments.

This plot presents two obvious trends. Neutral tweets concentrate in one region around the peak with a score of about 1.0. Positive tweets and negative tweets have similar distributions, their patterns overlap a lot, and concentrate in two regions around two peaks.

This implies the selected text is almost same as the original text when sentiment is neutral.\\

These three tree graphs display the most common words in positive, negative and neutral sentiments.

Positive words like 'good', 'happy', 'love', 'thanks' and 'great', appear frequently in positive tweets.

Negative words such as 'miss', 'sorry', 'bad' and 'hate', strongly expressed unsatisfied negative sentiment in tweets posted.

Other words in neutral sentiment actually do not express any special sentiments, like 'work', 'going' and 'get' etc.\\

There are variations of words, for example, people write “thanks” as “thnx” and they are considered as unique words.

These three DoNut plots represent unique words in the order of positive, negative and neutral sentiments, indicating that unique words play a significant role in determining the sentiments.\\

Then let's talk about the methods and models we used.\\

After EDA, we noticed that in training dataset, the average Jaccard score of neutral sentiment between text and selected text is around ninety-seven point six percent (97.6%), which implies the selected text are almost same as the original. So we consider to keep all the words as the selected text extracted in neutral tweets.(右边是变化的过程\\\

For positive and negative tweets, We use nltk to do NLP.

Firstly, we tokenized the text, attached a part-of-speech tag to each word, converting the P.O.S tags to simplify Wordnet tags.

According to the tag, we found the set of synonyms sininimens for each word in the nltk corpus, and calculated the sentiment scores individually in all synonym sets. After we got the average sentiment scores of all words in each text, we extracted the corresponding sentiment scores (pos or neg) against the sentiment of the text. At this point, we set a threshold sraishoud to filter out words with scores less than it.

We chose 0.06, the threshold with the highest score on the training set to predict on testing set, and the final Jaccard score was 52.6%. \\

After consulting relevant information, we decided to use NER to solve this problem. In our project, we use spacy.

First, we converted the training data into the format required by SpaCy

Second, Creating an empty SpaCy NER model and trained it by adding our custom entities present in the training dataset.

Third, We give the model feedback on its prediction, in the form of an error gradient of the loss function, which calculates the difference between the training example and the expected output.

Finally, saving the updated model.\\\

Then, we select the model that performs best on the validation set to predict the testing set. The Jaccard score is 57.8%, which is higher than nltk.

Next, we consider creating models separately on tweet text of each sentiment, the experimental result shows that this has a higher accuracy rate with a Jaccard score of 63.2%.\\

Finally, we use the Roberta Model in HuggingFace transformers.

All we gonna to do is converting data into the format we want. Like this (grph), we need five embeddings.

We input the data into the RoBERTa model and apply a drop layer, a Convolutinal layer, a Flatten layer and a softmax layer to get the corresponding probability. After data post-processing, the predicted answer can be obtained.  \\

We did the five-fold cross-validation on training set. Each fold, we set checkpoints to find the best model weights and save it.The final five-fold cross-validation average Jaccard score is 70.5%.Then we load the best model to predict the testing set, and the Jaccard score on the testing set is around 71%. \\

Here are the conclusions.\\

Extracting some words or phrases from sentences to reflect their emotions (neutral, positive or negative) is needed. In this project, we used a variety of models to train our data and predict the word or phrase from the tweet that are examples of the provided sentiment, including the nltk, SpaCy and RoBERTa. We found that RoBERTa achieved the best rather than other results.\\

For future work, the result of prediction based on nltk was not very good, and the words filtered by the threshold siraishoud were not continuous, which would cause the extracted text to be inconsistent. So we need to adjust the algorithm to make the sentence continuous.

The parameters of the pre-trained RoBERTa model are a lot and complex, and we are not very good at tuning parameters. We still need to continue to learn and be familiar with BERT.\\

That's it. Thank you for listening.

Do u guys have any questions.