

Assignment 5

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
USA	❤️	😍	😂	💕	🔥	😊	😎	✨	💙	😜	📷	🇺🇸	☀️	💜	😊	🏆	😄	🎄	📷	😜
ESP	❤️	😍	😂	💕	😊	😜	💪	😊	👉	🇪🇸	😎	💙	💜	😜	💕	✨	🎵	💕	😊	👉

ENGLISH

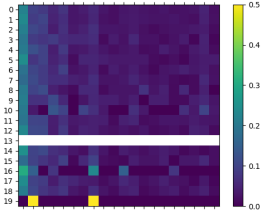
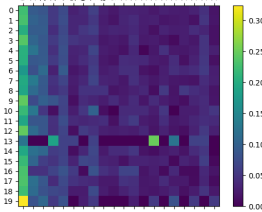
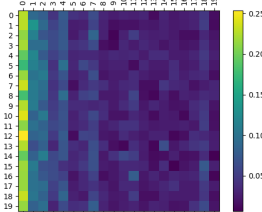
For the first question I use BERT to do the training and prediction. Since the dataset is not large enough, I didn't get good result. But I do observe something. For example, in both train and test dataset, emoji zero (the red heart) appear frequently. So, in the confusion matrix the first column always has the brightest color.

For English dataset I choose $2e-5$ as my learning rate and change epoch and batch size to see the performance. And the chart listed below shows the change in epochs and batch size.

1.

Learning rate : $2e-5$

Batch size: 32

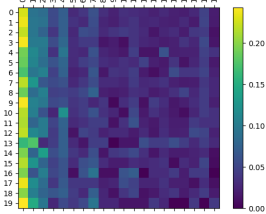
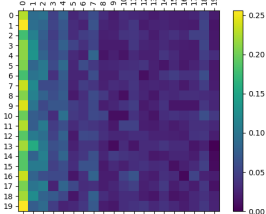
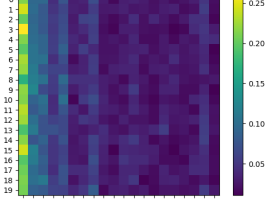
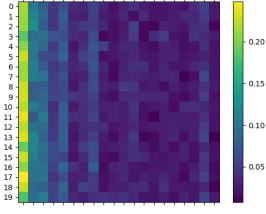
epochs	Confusion matrix	Macro_F-score Micro_F-score
3		4.419 12.54
6		5.126 11.38
9		4.42 6.4

Since when 9 epochs the F-score already dropped, so I stopped and take 6 as the epochs. The performance is bad and the brightest color always in the first column.

2.

Learning rate : 2e-5

epochs: 6

Batch size	Confusion matrix	Macro_F-score Micro_F-score
24		4.717 10.75
16		4.567 8.74
8		4.52 6.12
4		4.473 5.39

I take epochs equals to 6 and learning rate as 2e-5 and change the batch size. I tried from 24 to 4 and the result doesn't change much. The Macro F-score always around 4 and micro F-score is decreasing.

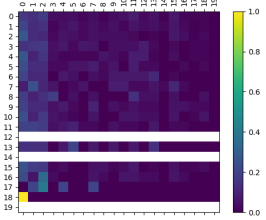
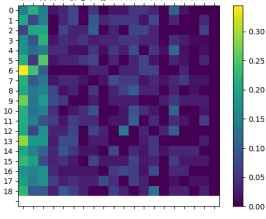
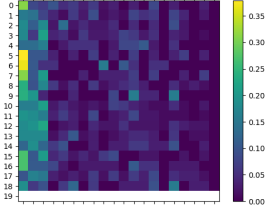
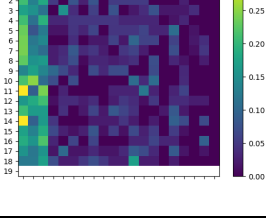
SPANISH

For Spanish dataset I do the same like I do for the English dataset. There are only 29000 text for training. So I modify the batch size for a really small number. And increased the epochs step by step. The result is listed below. (for Spanish please just ignore the last row and last column, because the labels are only 19)

1.

Learning rate : 2e-5

Batch size: 4

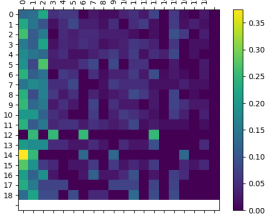
epochs	Confusion matrix	Macro_F-score Micro_F-score
5		4.366 8.5
8		3.97 4.5
10		5.074 6.7
15		2.903 3.8

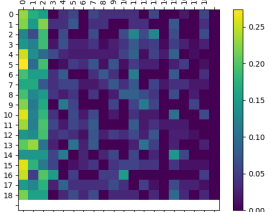
The best performance is epoch = 10 and the different between Spanish and English is t he brightest color is not the first column but the first three. And I found that the label 0-2 are same for the English dataset.

2.

Learning rate : 2e-5

Epoch: 4

Batch size	Confusion matrix	Macro_F-score Micro_F-score
5		4.049 5.4

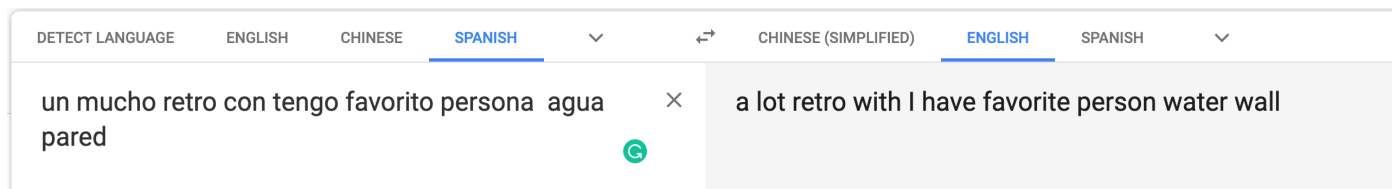
3		3.477 4.4
---	---	--------------

Changing the batch size seems doesn't help. So I stopped and move on to the last question.

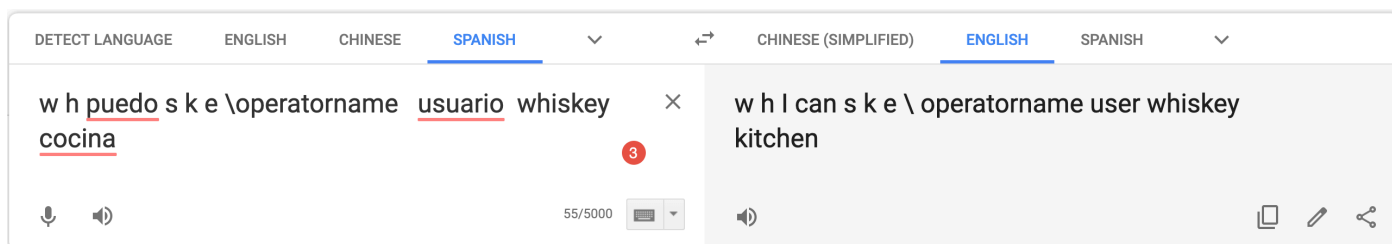
ENGLISH TO SPANISH

For this question, I use MUSE to do the translation. I download the `wiki.multi.en.vec` and `wiki.multi.es.vec` file and try to find the closest neighbor of each word in the English text. Unfortunately, the text in English are 90k which I ran a lot of time and finally gave up. Finally, I choose 10k text from the English dataset and do the translation. The translation part in `test.py` refer to the `demo.ipynb`.

I remove the illegal character like `@` or `...` and try to improve the accuracy of translation but the result of translation is still terrible. For example, the first sentence in English data is `'A little throwback with my favourite person @ Water Wall'` and after MUSE it becomes



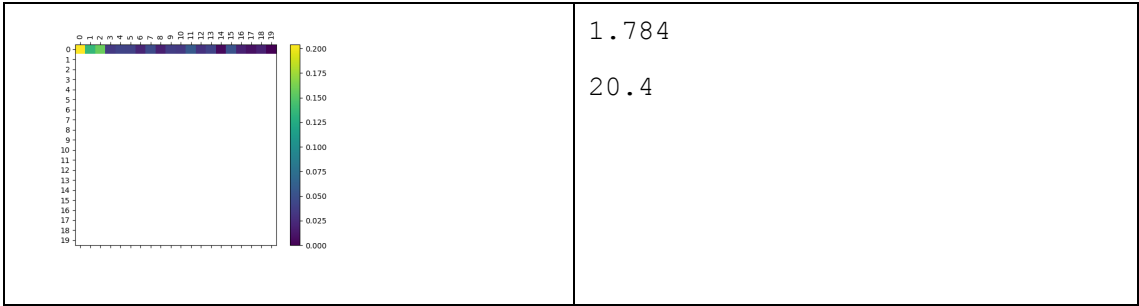
Which seems really nice but for another sentence like `'W H I S K E Y | : @user @ Whiskey Kitchen'` will becomes



So as you can see, text in Twitter always is very casual. Sometimes even people like the same things will know the meaning of some short names.

I used the originally Spanish data and 10k English data as an input with `train_batch_size=4`, `learning_rate=2e-5` and `num_train_epochs=10.0`. and the result is listed below.

Confusion	Macro_F-score
matrix	Micro_F-score



As you can see the performance is really bad which only because the prediction of this are all zero. I think the mainly reason is because the translation. (I've already translated the first 10k English dataset which can be seen in extend_data_test.csv). Since muse is word to word translation and there are many illegal character and network language can be seen in Tweet so the translation performance is really bad. Based on these facts, I only got zero from prediction which is the most frequently emoji people like to use. The high micro F-score proves this and low macro F-score proves that.