Deep Learning (COSC 2779) – Assignment 2 – 2021

Jakrapun Sangchan (s3808216)

1 Problem Definition and Review

Nowadays, social media is full of rumours and false claims. To fighting against false rumours the system will be built to determine the veracity of rumours. Stance classification is one a part of rumour detection pipeline. Therefore, the better stance classification led to better rumours detection system. The dataset has 6,253 tweets which has 381 source tweet and 5,872 replies. The reply has 4 classes support, query, deny and comment. The dataset contains 37 topics.

Literature review:

Title	Performance	S	Q	D	C	Technique
BUT-FIT at SemEval-2019 Task 7: Determining the Rumour Stance with Pre-Trained Deep Bidirectional Transformers	61.67	49.11	64.45	41.29	91.84	Pre-train BERT. Stance depend on source, it self and previous post. The best model is ensembling of 100 trained model.
CLEARumor at SemEval-2019 Task 7: ConvoLving ELMo Against Rumors	44.6	34.6	15.4	42.2	86.1	ELMO Embeddings
BLCU NLP at SemEval-2019 Task 7: An Inference Chain-based GPT Model for Rumour Evaluation	61.87	91	60	48	48	Data Expansion form SemEval 2016 task6. Use Pretrained GPT
BASE Line	49.3	43.8	55	7.1	91.3	Branch LSTM

Table 1. Literature review

This task is subtask A of SemEval-2019 task 7 [2]. The best performance system in subtask A is BLCU_NLP [6] which is use pretrained GTP [1]. The BLCU [6] use data from SemEval-2016 task 6 to increase dataset and train on both task at the same time. This might not able to do with our dataset which have data only from subtask A. The BUT-FIT [4] use ensemble of pretrained-BERT [3] model training only on subtask A. In the paper, the best model using ensemble on bert-large-uncased which have 24 transformer layers, 1024 hidden size and 16 attention heads can achieve 61.67 macro f1. However, due to the limitation of google colab resources I will look forward to the single model which is can achieve 56.24 macro f1-score on Bert_{big} and 53.39 macro f1-score on $\text{Bert}_{\text{base}}$. In this case, I would set target performance around 50 to 55 macro f1-score.

2 Evaluation Framework

The task is twitter classification. For this problem, I use **macro average F1-score** as the main evaluation metric and accuracy as support metric in case that macro F1 are not different. The reason that macro F1-score is more suitable than accuracy is the class distribution are imbalance. The macro average F1-score will threat every classes equally but accuracy will bias on majority class.

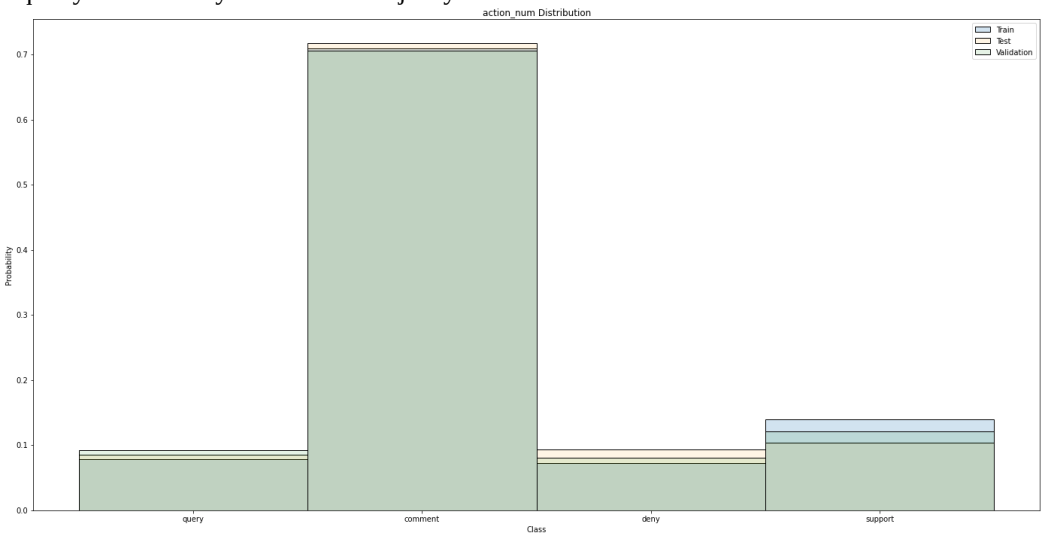


Fig 1. class distribution of train, validate and test set.

The twitter dataset was randomly split to train, validation and test by topic to prevent data leakage. Validation set was used to tune hyperparameter such as dropout rate and regularization. The test set was used to evaluated across difference models and make ultimate judgement.

3 Approach & Justifications

Data preprocessing have adopted from BUT-FIT [4]. The mention and URL were tokenized into $MENTION$ and URL using spacy library. Emoji has been converting to word by using emoji library. Then make all tweet into lower case before feed into TensorFlow preprocess for BERT. The source tweet and reply were feed into preprocessor. The preprocessor will add [CLS] token at the beginning [SEP] between 2 tweets and at the end. BUT-FIT assumed that the stance depends on itself, the source tweet and previous tweet. However, in our data all of target tweet replies directly to source tweet. Thus, the source tweet will be use with target tweet to make classification as discuss. The all of reddit data have been processed in the same way and was use as a training data.

The pretrained $BERT_{base}$ which have 12 layers, 768 hidden size and 12 attention heads was used as encoder in the solution. The experiment shows using $BERT_{large}$ and training is very time consuming it take around 14 mins per epoch and after 5 epochs the model seem to cannot learn much (0.2 macro f1-score). Thus, I decided to use $BERT_{base}$ model as an encoder and training on 3 different classifier structures. The first classifier is the same as BUT-FIT paper which have 1 layer of tanh with same size as hidden size of BERT encoder and 1 soft-max layer. The final architecture has 4-layer MLP adopt concept from [5]. The cross-entropy loss was use as a loss function and class weight was used to compensate imbalance class. Class weight was calculated from only training set distribution.

4 Experiments & Tuning

- Firstly, the $BERT_{large}$ was used to training with only twitter data. However, it takes too long to train around 14 mins per epoch.
- $BERT_{base}$ model was train on twitter data with 1 tanh layer and 1 soft-max layer. The result shows significantly overfitting.
- To mitigate overfitting, firstly, dropout technique was applied to the model.
- Dropout rate was varying from 0.3 to 0.7. The best dropout rate is 0.5.
- L2 regularization was combine with dropout to the model. The lambda was tuning start with 0.005, 0.01, 0.05 and 0.03 respectively. The best model of dropout + regularization is 0.5 dropout rate and lambda = 0.03. However, the performance is lower than use only dropout.
- Another $BERT_{base}$ model was train with 4 dense layers with tanh activation function and 1 soft-max layer. The result show significantly overfitting. Dropout rate have been varying from 0.3 to 0.1. The best model was 0.2 dropout rate.
- I also try to add reddit data into training set but the performance doesn't improve.

5 Ultimate Judgment, Analysis & Limitations

According to table2, the best model achieves 0.49 macro f1 score and 0.69 accuracy. The second-best model achieves 0.48 macro f1 score and 0.72 accuracy.

Model	Macro-f1	accuracy	f1-S	f1-Q	f1-D	f1-C
BERTbase + 1 layer	0.48	0.7	0.3	0.52	0.28	0.82
BERTbase + 1 layer +dropout	0.48	0.72	0.26	0.57	0.26	0.83
BERTbase + 1 layer +dropout + regularization	0.47	0.73	0.22	0.62	0.21	0.83
BERTbase + 4 layer	0.48	0.7	0.31	0.6	0.19	0.81
BERTbase + 4 layer + dropout	0.49	0.69	0.28	0.61	0.27	0.81
BERTbase + 4 layer + reddit	0.46	0.66	0.3	0.59	0.19	0.78

Table 2. Test set performance

The confusion matrix was use to determine the final model. Fig 2 shows $BERT_{base}$ + 4 layers + dropout have better f1 score on query, deny and support class but $BERT_{base}$ + 1 layer + dropout have better performance on comment class which is the majority class. This makes $BERT_{base}$ + 1 layer + dropout have higher accuracy.

	precision	recall	f1-score	support
0	0.59	0.63	0.61	122
1	0.79	0.82	0.81	1026
2	0.32	0.24	0.27	133
3	0.29	0.27	0.28	148
accuracy			0.69	1429
macro avg	0.50	0.49	0.49	1429
weighted avg	0.68	0.69	0.69	1429

	precision	recall	f1-score	support
0	0.63	0.52	0.57	122
1	0.78	0.88	0.83	1026
2	0.37	0.20	0.26	133
3	0.32	0.22	0.26	148
accuracy			0.72	1429
macro avg	0.53	0.45	0.48	1429
weighted avg	0.68	0.72	0.69	1429

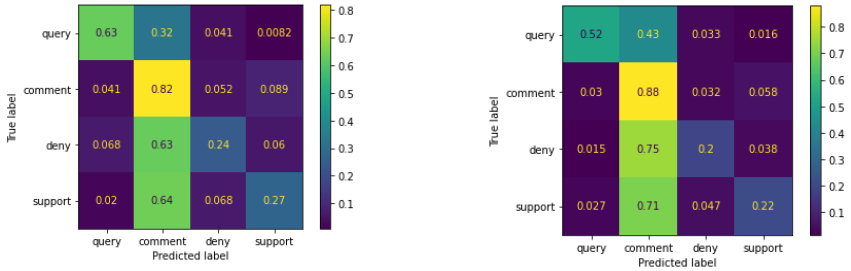
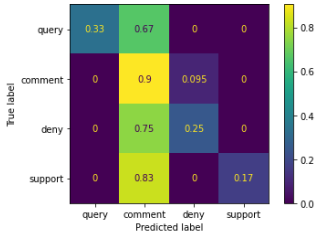


Fig 2. Confusion matrix. Left: $BERT_{base}$ + 4 layers + dropout. Right: $BERT_{base}$ + 1 layer + dropout

To conclude, $BERT_{base}$ + 4 layers + dropout will be final model due to higher macro f1-score and have more balance performance on each class.

6 Real world data:

	precision	recall	f1-score	support
0	1.00	0.33	0.50	3
1	0.66	0.90	0.76	21
2	0.33	0.25	0.29	4
3	1.00	0.17	0.29	6
accuracy			0.65	34
macro avg	0.75	0.41	0.46	34
weighted avg	0.71	0.65	0.60	34



The data have been collected from 2 tweet2 about blacklivesmatter and ImmigrantsAreEssential topics. There consists of 34 replies classifies as 21 comments, 3 query, 4 deny and 3 support. The result shows that model performance has been decrease from 0.49 to 0.46 on real world data.

7 References:

[1] A. Radford, K. Narasimhan, S. Tim et al., *Improving language understanding by generative pre-training*, 2018.

[2] G. Gorrell et al., "RumourEval 2019: Determining rumour veracity and support for rumours", *Proc. 13th Int. Workshop Semantic Eval.*, pp. 845-854, 2019.

[3] J. Devlin, M.-W. Chang et al., *Bert: Pre-training of deep bidirectional transformers for language understanding*, 2018.

[4] M. Fajcik, P. Smrz and L. Burget, "BUT-FIT at SemEval-2019 task 7: Determining the rumour stance with pre-trained deep bidirectional transformers", *Proc. 13th Int. Workshop Semantic Eval.*, pp. 1097-1104, 2019, [online] Available: https://www.aclweb.org/anthology/S19-2192.

[5] R. Anggrainingsih, G. Hassan and A. Datta, "BERT based classification system for detecting rumours on Twitter", *IEEE transactions on Computational and Social Systems (still underreview process)*, 2021. Available: https://arxiv.org/abs/2109.02975. [Accessed 17 October 2021].

[6] R. Yang, W. Xie, C. Liu and D. Yu, "BLCU_NLP at SemEval-2019 task 7: An inference chain-based GPT model for rumour evaluation", *Proc. 13th Int. Workshop Semantic Eval.*, pp. 1090-1096, 2019, [online] Available: https://www.aclweb.org/anthology/S19-2191.