Intro to NLP: Assignment 1

In this assignment, we work with a dataset that contains English tweets. It has been collected for a shared task at SemEval 2018 that aims at detecting irony.

Task Organizers: Cynthia Van Hee, Els Lefever, and Veronique Hoste
Dataset: https://github.com/Cyvhee/SemEval2018-Task3/tree/master/
Paper: https://www.aclweb.org/anthology/S18-1005.pdf
Please use this file for your analyses on the training data: **datasets/train/SemEval2018-T3-train-taskB.txt**
Your analyses should be conducted using python.
You are allowed to use Python packages (e.g. pandas, sklearn).

All floating point numbers should be rounded to **two decimals**.
The format of the assignment should not be changed.
Total Points: 27 (Part A: 8, Part B: 9, Part C: 10)

A) Linguistic analysis using spaCy (total: 8 points)

Note that we are using the most recent spaCy version (3.0.5) and the model "**en_core_web_sm**". Results might vary for other versions. If you cannot use 3.0.5, clearly explain this to your TA and specify on your submission which version you are using instead.

1. **Tokenization (1 Point)**
   Process the dataset using the spaCy package and extract the following information:

   Number of tokens: 66518
   Number of types: 10790
   Number of words: 47420
   Average number of words per tweet:12.37
   Average word length: 4.42

   It should be noted that, we think that the number of tokens includes punctuation marks and other special symbols, such as emoji, URL, etc., so we did not perform any filtering operations. For the counting of word and type, we filter out punctuation marks, spaces, user id, URL links, digitals, @ and # symbols, etc., but we assume that the difference in capitalization represents different types and words. That is, we did not convert all words to lowercase. One reason for this is that we believe that capitalization is also, to a certain extent, represents some semantic information. For example, a word in all capitals may have an emphasis on meaning, This may be useful for judging whether there is irony or not.

   Words should NOT include punctuation tokens.

2. **POS-Tagging (3 Points)**
   Complete the table below for the ten most frequent POS-tags that the tagger assigns (the tagger in the model "**en_core_web_sm**" is trained on the PENN Treebank tagset).

Finegrained POS-tag	Universal POS-Tag	Occurrences	Relative Tag Frequency (%)	3 most frequent tokens with this tag	Example for an infrequent token with this tag
NN	NOUN	10027	15.07	#, day, time	background
NNP	PROPN	5213	7.84	Christmas, #, RT	CV
IN	ADP	4232	6.36	of, in, for	@BBCRadMac
.	PUNCT	3901	5.86	., !, ?	http://t.co/MXJXW5LOqz
DT	DET	3729	5.61	the, a, this	EVERY
PRP	PRON	3626	5.45	I, you, it	r
JJ	ADJ	3487	5.24	great, good, new	missing
RB	ADV	2915	4.38	so, just, now	myfairdaily
VB	VERB	2782	4.18	be, get, see	Want
NNS	NOUN	2737	4.11	#, people, hours	Hummingbirds

The Universal POS-Tag and Finegrained POS-tags need to combine together (not just considering a single one). This is because (as shown in the table above), for a certain Universal POS-Tag, such as NOUN, it can correspond to multiple Finegrained POS-tag, such as NN and NNS. In other words, only consider Universal POS-Tag or Finegrained POS-tag does not get the correct result.

3. **Normalization (1 Point)**
   Does it make sense to apply any filtering or normalization on the dataset? Justify your answer in 2-4 sentences.

   It depends on the specific application or task. If we want to do semantic analysis, or sentiment analysis, for example, to detect whether it is a ironic sentence, then it would be better not to do any filtering or normalization. The main reason is that some special symbols, such as emoji, clearly express whether or not they contain ironic information. In addition, capitalized words in the context can also indicate a strong tone to a certain extent. If our task is word embedding, then we need to filter emoji and other content, such as hyperlinks.

4. **Lemmatization** (1 Point)
   Provide an example for a lemma that occurs in more than two inflections in the dataset.
   Lemma: be
   Inflected Forms: are, 's
   Example sentences for each form:

1. We are rumored to have talked to Erv's agent... and the Angels asked about Ed Escobar...
2. that's hardly nothing ;)

5. **Named Entity Recognition** (2 Points)
   Number of named entities: 4743
   Number of different entity labels: 18

   Analyze the named entities in the first three sentences. Are they identified correctly? If not, what would be the correct decision?

   In the named entity recognition of the first three sentences, some of the entities were successfully recognized, but there are still several entity recognition errors.
   For the first sentence, ‘Sweet United Nations’ should be ‘United Nations’, ‘Sweet’ should not be the part of the entity; # and NoReligion should not be entities. For the second sentence, ‘Erv’ could be correctly identified or should be ‘Erv's agent’; The third sentence is correct.

B) Classification Task Analysis (total: 9 points)

1. Class Distributions (1 Point)

Analyze the number of instances for each of the four classification labels.

Class Label	Instances	Relative Label Frequency (%)	Example sentence with this label
0	1923	50.16	3 episodes left I'm dying over here
1	1390	36.25	Nothing makes me happier then getting on the highway and seeing break lights light up like a Christmas tree.
2	316	8.24	crushes are great until you realize they'll never be interested in you.| :p
3	205	5.35	It will be impossible for me to be late if I start to dress up right now. #studing #university #lazy

2. **Baselines (5 points)**
   Calculate two baselines and evaluate their performance on the test set:
   datasets/test_TaskB/SemEval2018-T3_gold_test_taskB_emoji.txt
   (Note that this file contains emojis. If this causes problems, you can get the tweets without emojis from SemEval2018-T3_input_test_taskB.txt)

   The first baseline is a random baseline that randomly assigns one of the four classification labels. Make sure to fix the random seed and average the results over 100 iterations!

   The second baseline is a majority baseline that always assigns the majority class. Calculate the results on the test dataset and fill them into the two tables below. Round the results to two decimals.

Random Baseline				
Class	Accuracy	Precision	Recall	F1
0	0.25	0.60	0.25	0.35
1	0.26	0.21	0.26	0.23
2	0.26	0.11	0.26	0.15
3	0.26	0.08	0.26	0.12
macro - average	0.26	0.25	0.26	0.22
weighted average	0.25	0.43	0.25	0.29

Majority Baseline				
Class	Accuracy	Precision	Recall	F1
0	1.00	0.60	1.00	0.75
1	0.00	0.00	0.00	0.00
2	0.00	0.00	0.00	0.00
3	0.00	0.00	0.00	0.00
macro - average	0.25	0.15	0.25	0.19
weighted average	0.60	0.36	0.60	0.45

3. **Understanding the task (2 points)**
   Read (at least) **section 3** of the paper to better understand the different classes. Provide **a new example** (a tweet that is not in the dataset) for each of the three irony classes in task B and explain why it belongs to this class. It might be helpful to examine some tweets and their labels in the training data.

Class	Example	Explanation
1	The students here are very honest, and they often cheat in exams.	This ironic statement has a strong polarity contrast: Honest and cheat form a strong contrast
2	I just drank a healthy, homemade, all fruit smoothie...in a @Budweiser beer glass. #irony	It is describing a irony situation where he/she drank smoothie in a beer glass.
3	I bet that lawyer was fun at parties. #sarcasm	This ironic statement contains neither a description of the situation nor a strong reversal: telling that the lawyer in the text is hilarious/unreliable.

4. **Classification as fine-tuning** (1 Point)
   As a preparation for the part C of the assignment, you need to work through this notebook by Chris McCormick and Nick Ryan. It explains how you can finetune a large pre-trained language model (BERT) for sentence classification tasks. Language models will be introduced next week but we would like you to already get familiar with some concepts. The notebook is very well documented, but there will be steps and terms that you do not completely understand. Describe which steps are not yet clear to you and why. [app. 5 sentences, depends on your level of understanding]

   Note: If you encounter this warning when running the notebook "*Some weights of the model checkpoint at bert-base-uncased were not used when initializing BertForSequenceClassification…*" you can ignore it for this assignment.

   I am not very clear about the meaning of Learning Rate Scheduler. In the tutorial, author uses a Linear Scheduler to update the learning rate. What are the specific benefits of doing this? In addition, author also used a parameter of num_warmup_steps and set the parameter to 0. If using Learning Rate Scheduler can improve the performance of the model, what is the specific meaning of num_warmup_steps? How to set an appropriate num_warmup_steps?

C) Irony classification by fine-tuning BERT (total: 10 points)

We now want to adjust the code for fine-tuning BERT (see exercise B4) to work with the irony detection dataset. These exercises can be tricky if you are not an experienced programmer.

Start early, ask questions in Piazza and tell your TA from the beginning that you might need more detailed explanations. If you still do not manage to solve exercises 1 and 2, provide a description of everything you tried and explain where you got stuck.

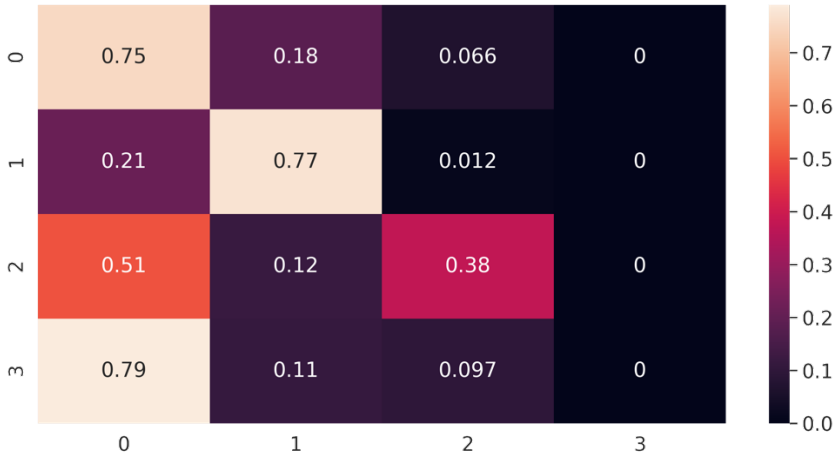
Note: If you run the notebook multiple times, you might encounter GPU restrictions. You can run the code on your own computer without a GPU but processing will take much longer. Keep this in mind when planning for the deadline.

1. **Classification and Evaluation (5 points)**
   Adjust the evaluation code for the test dataset so that you can output the **F1-measure per class** and the confusion matrix.
   Now adjust the notebook to work with the **irony dataset**. Make sure to set the maximum sentence length to a useful value (based on the sentence lengths in the training data).

   a) Provide the results for accuracy, precision, recall and F1-measure and plot a confusion matrix. (3 Points)

Finetuned BERT				
Class	Accuracy	Precision	Recall	F1
0	0.74	0.74	0.74	0.74
1	0.77	0.55	0.77	0.64
2	0.38	0.45	0.38	0.41
3	0.00	0.00	0.00	0.00
macro - average	0.48	0.43	0.48	0.45
weighted average	0.64	0.61	0.66	0.63

Confusion Matrix: Finetuned BERT				
	Predicted Class			
Gold Class	0	1	2	3
0	0.75	0.18	0.07	0
1	0.21	0.77	0.01	0
2	0.51	0.12	0.38	0
3	0.79	0.11	0.10	0



Confusion matrix for Finetuned BERT

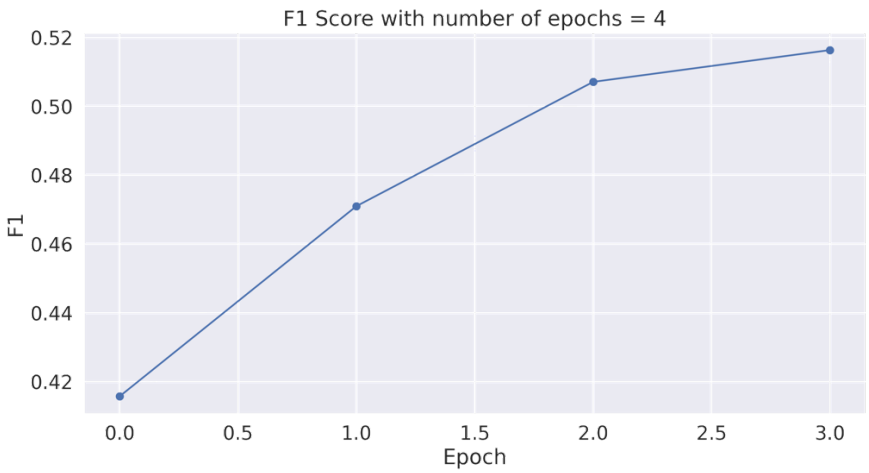
b) Compare your results to the results in section 7 of the paper in 2-4 sentences. (2 Points)

Overall, the accuracy of the model is not so good, it is because the model is difficult to identify class 2 and 3. The main reason may be that the number of samples of these two types are too small, and the model cannot well capture the semantic characteristics of these two types. A potential method is to apply data augmentation to increase the number of samples of these two types. The overall precision score of the model is not bad, ranking at sixth place. The recall of the model is probably in the 5th place, and the F1 score of the model is also in 5th place. The bottleneck of the overall performance of the model is that the sample is not balanced, so we need to solve this problem.

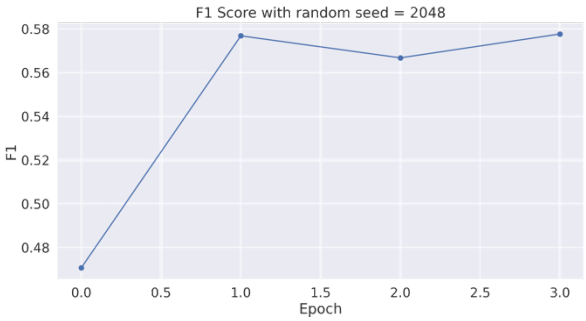
2. **Exploration (2 points)**
Vary a hyperparameter of your choice and plot the results for at least 5 different values. Examples for hyperparameters: number of epochs, random seed, learning rate, epsilon. Interpret the result (3-6 sentences).

Please use the macro-average F1-measure as evaluation metric and evaluate on the validation set of the irony dataset. If you didn't solve exercise 1, you can use accuracy and the CoLA dataset (specify this clearly).

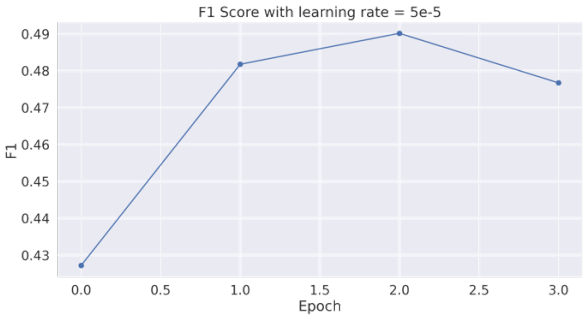
Your plot:



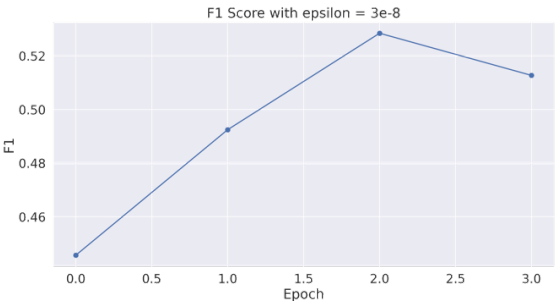
Control group (original parameters)



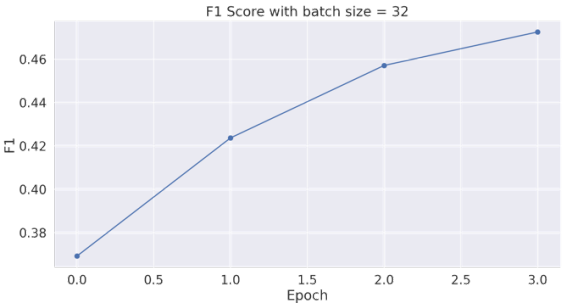
Experimental group (change the parameters of random seed)



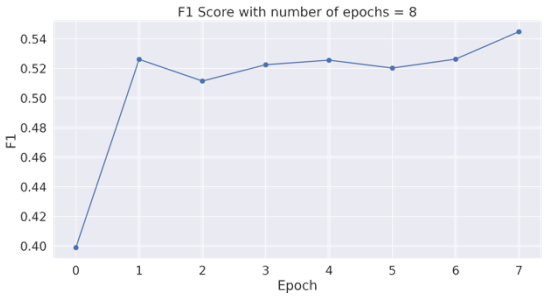
Experimental group (change the parameters of learning rate)



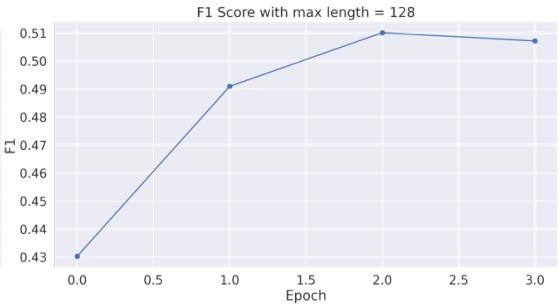
Experimental group (change the parameters of epsilon)



Experimental group (change the parameters of batch size)



Experimental group (change the parameters of epochs)



Experimental group (change the parameters of max length)

Your interpretation:
First, the original hyperparameters of the model are: number of epochs=4, random seed=1024, learning rate=2e-5, epsilon = 1e-8, batch size = 16, max length(sentence) = 64. Only one of these hyperparameters is changed at a time, and the rest remain unchanged. The purpose is to see the impact of the change of each hyperparameter on the model performance (macro-average F1 score), and the results are shown in the figure above.

From this, we know that when the random seed is changed to 2048, the performance of the model will be improved by a little, but the effect is not obvious. The performance improvement brought by the change of the random seed cannot be used as the basis for our selection of the model. The increase in learning rate (from 2e-5 to 5e-5) reduces the performance of the model, the possible reason is that the adjustment range of the model parameters is too large by large learning rate and it is difficult to fit. The change of epsilon has no obvious effect on the performance of the model. A larger batch size will have a bad influence on the model, the possible reason is that our data volume is too small, so the model is under-fitting by large batch size. The increase of epochs has no obvious impact on the performance of the model, but it will bring the risk of overfitting. Increasing the length of the maximum sentence to 128 has no obvious impact on model performance, but it will increase the time and space complexity of model calculation and reduce the speed of reasoning.

3. **Analysis (3 points)**

Save the fine-tuned irony classification model. Compare the BERT representations of the last layer before fine-tuning and after fine-tuning. You can find some useful code snippets in bert_example.zip.

a) Extract the representations for the first 20 tweets of the training set using the **untuned BERT** model. Use the vector for the **CLS token in the last layer** as the representation for the tweet. Calculate the pairwise cosine similarity between tweets. Provide 3 examples for pairs of tweets with high similarity and 3 examples for pairs with low similarity. Use the following format:

3 examples for pairs of tweets with high similarity:
S1: Sweet United Nations video. Just in time for Christmas. #imagine #NoReligion http://t.co/fej2v3OUBR (Label:1)
S6: You're never too old for Footie Pajamas. http://t.co/ElzGqsX2yQ (Label:0)
Similarity: 0.95

S4: 3 episodes left I'm dying over here (Label:0)
S8: 4:30 an opening my first beer now gonna be a long night/day (Label:0)
Similarity: 0.91

S10: @samcguigan544 You are not allowed to open that until Christmas day! (Label:0)
S13: @TargetZonePT :pouting_face: no he bloody isn't I was upstairs (Label:0)
Similarity: 0.92

3 examples for pairs of tweets with low similarity:
S3: Hey there! Nice to see you Minnesota/ND Winter Weather (Label:1)
S14: Cold or warmth both suffuse one's cheeks with pink (colour/tone) ... Do you understand the underlying difference & its texture? (Label:0)
Similarity: 0.76

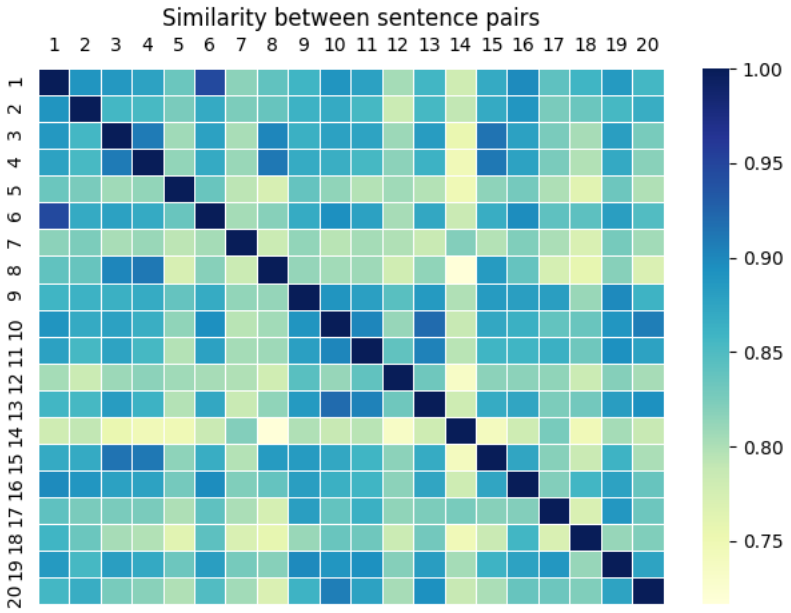
S8: 4:30 an opening my first beer now gonna be a long night/day(Label:0)
S14: Cold or warmth both suffuse one's cheeks with pink (colour/tone) ... Do you understand the underlying difference & its texture? (Label:0)
Similarity: 0.72

S12: But instead, I'm scrolling through Facebook, Instagram, and Twitter for hours on end, accomplishing nothing. (Label:0)
S14: Cold or warmth both suffuse one's cheeks with pink (colour/tone) ... Do you understand the underlying difference & its texture? (Label:0)
Similarity: 0.73

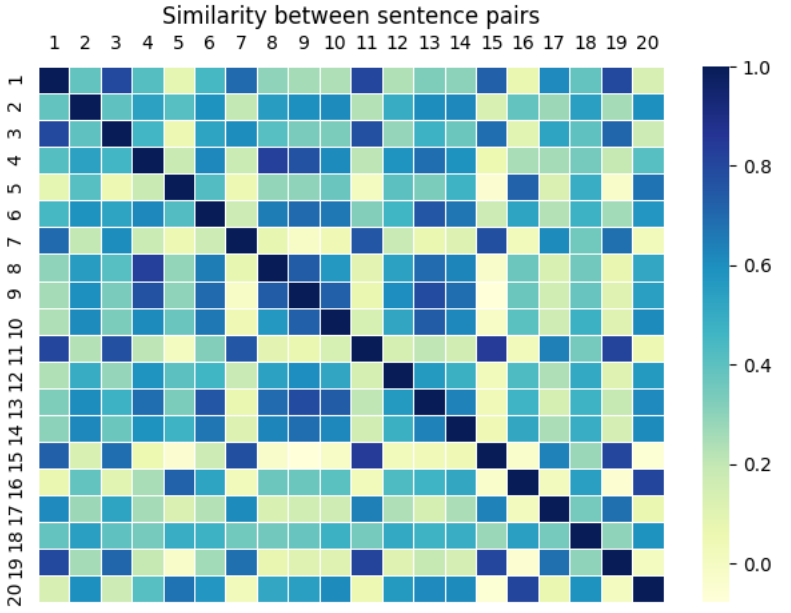
Are the similarity values intuitive for these pairs of tweets or would you expect something else? Justify your answer in 3-5 sentences. (1 Point)

For sentences with high similarity score, they tend to have the same pattern. For example, there are hyperlinks, digitals (or corresponding to the same position, for example, at the beginning), and @ signs. and many more. In addition, the similarity score between sentences pair is very high, and the difference is not obvious, most of the scores are above 0.7. It should be noted that the BERT model that has not been fine-tuned does not perform well for semantic distinction, and the model fluctuates greatly. It is hard to distinguish sentences with dissimilar semantics. Also, the first pair with high similarity are labeled differently in the sense of irony level, we were expecting that the similarity can somewhat identifies tweets with the same irony level.

b) Extract representations for the first 20 tweets using **the model that has been tuned to the irony detection task**. Use the vector for the CLS token in the last layer as the representation for the tweet. Compare the pairwise similarities to the ones you obtained from the untuned model and also take the label of the tweets into account (3-5 sentences). Provide an example of a pair of tweets for which the similarity changes remarkably. (2 Points)



BERT without fine tuning



BERT with fine tuning

S1: Sweet United Nations video. Just in time for Christmas. #imagine #NoReligion http://t.co/fej2v3OUBR (Label:1)
S6: You're never too old for Footie Pajamas. http://t.co/ElzGqsX2yQ (Label:0)
Similarity: 0.45

S4: 3 episodes left I'm dying over here (Label:0)

S8: 4:30 an opening my first beer now gonna be a long night/day (Label:0)
Similarity: 0.46

S10: @samcguigan544 You are not allowed to open that until Christmas day! (Label:0)
S13: @TargetZonePT :pouting_face: no he bloody isn't I was upstairs (Label:0)
Similarity: 0.74

S3: Hey there! Nice to see you Minnesota/ND Winter Weather (Label:1)
S14: Cold or warmth both suffuse one's cheeks with pink (colour/tone) ... Do you understand the underlying difference & its texture? (Label:0)
Similarity: 0.37

S8: 4:30 an opening my first beer now gonna be a long night/day(Label:0)
S14: Cold or warmth both suffuse one's cheeks with pink (colour/tone) ... Do you understand the underlying difference & its texture? (Label:0)
Similarity: 0.63

S12: But instead, I'm scrolling through Facebook, Instagram, and Twitter for hours on end, accomplishing nothing. (Label:0)
S14: Cold or warmth both suffuse one's cheeks with pink (colour/tone) ... Do you understand the underlying difference & its texture? (Label:0)
Similarity: 0.48

The figure above shows the comparison of pairwise similarities of the model before and after fine-tuning. After comparing before and after fine tuning, it is obvious that after fine-tuning the BERT model, the model tends to give higher similarity scores for two sentences with the same label, while for two sentences with different labels, the model tends to give a lower similarity score. Because the same label has a higher probability of containing the same semantics, but this is not an absolute result. The model shows such a trend, but not every result follow this pattern. In addition, the fine-tuning model can give more differentiated similarity scores, which means that for different sentences, the model can distinguish the differences between sentences to a certain extent. Moreover, whether before or after fine-tuning, for the same sentence (the diagonal of the confusion matrix), the model tends to assign a value of 1. This it is correct because the similarity between them should be 1 for the same sentence.

The pairwise similarities between sentence (1,6), (9,15) has changed remarkably.

S1: Sweet United Nations video. Just in time for Christmas. #imagine #NoReligion http://t.co/fej2v3OUBR
S6: You're never too old for Footie Pajamas. http://t.co/ElzGqsX2yQ

ORD: 0.95
NEW: 0.45

S9: @Adam_Klug do you think you would support a guy who knocked out your daughter? Rice doesn't deserve support.
S15: Just great when you're mobile bill arrives by text
ORD: 0.88
NEW: -0.08

One explanation for the significant change of the similarity score between these two sentences pairs is that they express different, or even opposite semantics, such as positive and negative meaning.