Covid-19 Impacts on Global Education

Teoman Berkay Ayaz
Computer Engineering
Istanbul Kultur University
Istanbul, Turkey
1800004169@stu.iku.edu.tr

Muhammed Safa Uslu
Computer Engineering
Istanbul Kultur University
Istanbul, Turkey
1700003646@stu.iku.edu.tr

İbrahim Ağcabay
Computer Engineering
Istanbul Kultur University
Istanbul, Turkey
1900002653@stu.iku.edu.tr

I. Introduction

Our project in accordance with the given project description aims to; develop a series of neural networks to analyze the sentiment in a given sample (tweet), minimize overfit and underfit in our neural network, build the most optimal neural networks with our data, and lastly analyze the impact of covid 19 on education.

II. Preferred Data Collection Method

Due to the last week limitation of twitter API, and the fact that GetOldTweets3 no longer working, we decided to use twint since it would be our best choice regarding data collection due to that it could collect more tweets that any other possible method we had available.

III. Data Collection Process& Logic

The logic behind the way we collected data is that we first decided on the keywords we were to use to collect the data. After settling on a total of 6 (8 with plurals) keyword pairs to use as our queries, we collected as much as twint allowed us to collect. After collecting a total 14002 rows of data and filtering out non-English tweets we were left with a grand total of 13388 tweets.

Keyword 1	**Keyword 2**
Education	Covid-19
Education	Coronavirus
Education	Online
Distance	Learning
Student(s)	Covid-19
Student(s)	Coronavirus

(Table 3.1 Keywords used as queries)

One other thing about said data was processed is that "ISO-8859-1" encoding was used when loading with pandas; hence some of the characters were unable to load properly.

Keyword Pair	**Rows Collected**
Education&Covid-19	2813
Education&Coronavirus	126
Education&Online	5804
Distance&Learning	1468
Student(s)&Covid-19	3348
Student(s)&Coronavirus	443

(Table 3.1 Number of rows collected from each keyword)

IV. Dataset Classes

Our dataset contains a total of 4 different labels. Our classes compose of three sentimental classes and an irrelevant (3) class. Our sentimental classes are negative (0), neutral (1), positive (2). Anything negative remark regarding education under covid or any overall negative sentiment in our dataset is labeled as a negative sentiment. Any statistical or factual tweet related to our subject was regarded as a neutral sentiment in our project. Positive sentiment class is composed of any tweet that regards education under covid positively or any tweet that contains an overall positive sentiment. Lastly, the irrelevant class is made up of any tweet that is irrelevant to our subject such as ads, bot tweets etc.

Class	Count	Percentage
0	439	19%
1	445	21%
2	329	15%
3	1018	45%

(Table 3.3 Class distributions in dataset)

V. Preprocessing

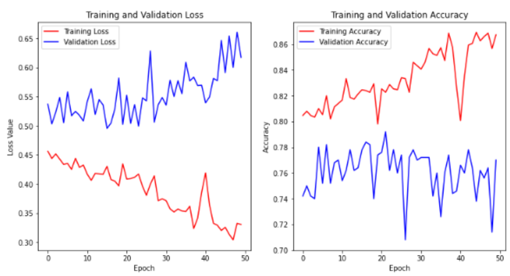
The initial preprocessing step we applied in our project was to convert every English character in our tweets to lowercase using pythons built in "to_lowercase" function.

Next preprocessing step to be applied was to remove non-alphabetic characters from our tweets. For this we used NLTK stopwords. Calling a function from NLTK again, we use "word_tokenizer", to tokenize our tweets. After this step is done, we can simply index our tokens and the training data is ready.

One additional preprocessing step we took was to shuffle our rows to distribute our 9032 unique tokens more evenly throughout training and validation data.

VI. Model Optimization

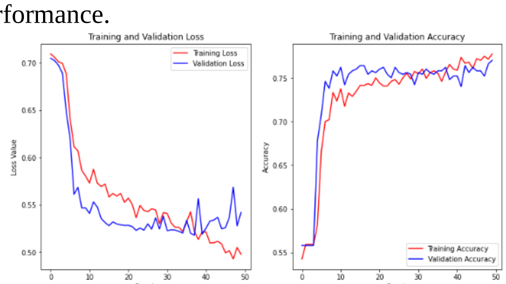
When we built our initial models, we were faced with intense amounts of overfit like in the following example.



(Image 6.1 Initial model performance)

The first step that came to mind was to use dropout layers. Generally, a dropout rate about 0.4 to 0.6 was optimal for our models. After dropouts, we tried adding regularizers, adjusting unit counts, different activation functions for our hidden layers (most notably tanh activation function), and changing the layer counts.

After all the methods above were applied to our neural networks the following plot was the performance.



(Image 6.2 Optimized model performance)

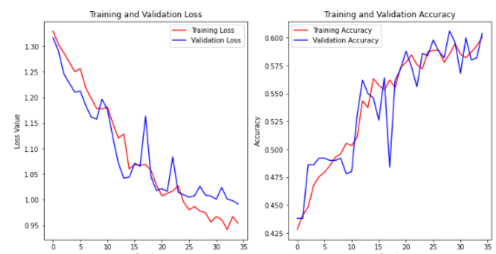
Other methods we tried to optimize our models include learning rate adjustments, other optimizers. With regards to learning rate, too high or too low rates were causing our models to flatline. And we found the default learning rate of "Adam" optimizer which is 0.001 to be optimal for our case.

With regards to alternative optimizer, throughout the optimization of our models we found the "Adam" optimizer to be ideal for our model compared to other available optimizers, more notably "SGD" optimizer which caused our models to flatline.

VII. All-Classes Classifier

The All-Classes Classifier as the name suggests, makes classification between all our four classes.

Its architecture consists of five total layers starting with; an embedding input layer with "trainable" parameter set to false, an LSTM layer with 64 units, a dense layer with 16 units, "relu" activation and 0.1 as an l2 bias regularizer, a dropout layer with a dropout rate of 0.5, and lastly, an output layer with "softmax" activation since it is a multiclass classification model.

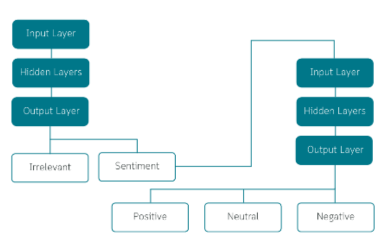


(Image 7.1 All-Classes Classifier Model Performance)

The model yields about (± 2%) 60% accuracy and continues training until epoch 50 since overfit is rather minimal hence, dismissible.

VIII. Hierarchical Classifier

The hierarchical classifier consists of two separate neural networks. The prior is a relevance classifier which classifies whether the given sample is within the irrelevant class or not.



(Image 8.1 Hierarchical Classifier Flowchart)

If the classifier classifies the sample as relevant, we move it to the second neural network in our hierarchy, the sentiment multiclass classifier. The sentiment multiclass classifier classifies which class a given sample falls into.

One thing that should be noted, is that in the event that the initial classifier makes a false classification the second classifier has no chance of making a correct classification.

The architecture of the relevance binary classifier, consists of a total of seven layers starting with; an embedding layer with "trainable" parameter set to false as its input layer, an LSTM layer with 32 units, a flatten that comes after the LSTM layer, two back to back dense layers with 8 units, tanh activation function, l2 regularizers with 0.001 regularization rate as kernel and bias regularizers, a dropout layer with a dropout rate of 0.5 , and an output layer with sigmoid activation since it is a binary classification model.

Its architecture of sentiment multiclass classifier is composed of a total of 6 layers. The initial layer is the embedding input layer with trainable set to false followed by an LSTM layer with 64 units. The next layer is a dense layer with 16 units, "relu" activation and an l2 regularizer with a regularization factor of 0.2 as its bias regularizer. Next, we have another dense layer with 8 units and, "relu" activation and an l2 regularizer with a regularization factor of 0.2 as its bias regularizer. Next, we have a dropout layer with a dropout rate of 0.55 followed by the output layer with "softmax" activation since it is a multiclass classification model.

The relevance classifier and sentiment classifiers each have respectively about 75% and 59% accuracy each. Due to combined probability, they yield an accuracy of 44.25%. Which when compared to the all-classes classifier, is considerably lower and makes the latter, irrelevant for any potential use.

IX. Information Gathered

When we look at the ratio of tweets who are happy about the situation of education under covid conditions (positive sentiments – Class 2) and the tweets that are negative regarding the situation (negative sentiments – Class 0) which is roughly (rounded down to two decimals) 0.75, we can see that significantly higher amount of people are unsatisfied with the situation of education in covid conditions.

When we thoroughly inspect the tweets, we can see that most of the negative tweets have certain subjects in common mostly regarding inequity, poverty and that some students are falling behind compared to their peers.

With that said however, despite being about 25% lower in numbers, some people are satisfied or happy about the fact that education can now be online. One of the more common things that the positive tweets was that students with social anxiety performed better in distance learning than they did with face-to-face learning.

According to the information on the tweets it seems that, distance learning gave students with certain disabilities and/or high levels of social anxiety a better chance at learning and having higher levels of academic success.

One other notable find was that about 45% of the tweets were in the irrelevant class (Class 3), which mostly comprise of advertisements. When we inspect the said tweets, we can see that a significant amount educational institutions such as universities, academies, business schools etc. are offering programs that would normally would have been a 4-year undergraduate program or a 2-year master's program pre-covid.

This shows us that the education industry has adapted to this new era of online learning and there are lots of more thing that can be learned online ranging from flying a commercial aircraft to developing a successful neural network model to analyze tweets off of twitter.