**Intermediate Report:**

Deepika Padmanabhan: A20456289

Dimple Mehra : A20457225

**Few Shot Learning**

**Introduction:**

Few-shot learning refers to understanding new concepts from only a few examples. We define a training objective that aims to extract as much information as possible from each training batch by effectively optimizing over all relative orderings of the batch points simultaneously.

This is a challenging setting that necessitates different approaches from the ones commonly employed when the labelled data of each new concept is abundant. Indeed, many recent success stories of machine learning methods rely on large datasets and suffer from overfitting in the face of insufficient data. It is however not realistic nor preferred to always expect many examples for learning a new class or concept, rendering few-shot learning an important problem to address

**1 Problem Statement**

Fake news and hoaxes have been there since before the advent of the Internet. The widely accepted definition of Internet fake news is: fictitious articles deliberately fabricated to deceive readers”. Social media and news outlets publish fake news to increase readership or as part of psychological warfare.

The purpose of the work is to come up with a solution that can be utilized by users to detect and filter out sites containing false and misleading information. We use carefully selected features to accurately identify fake posts.

Our project aims at classifying the given news articles as fake or true based on the content using few shot learning models with less data. To begin with, we have used pre-trained BERT models as our baseline.

**2 Implementation Details:**

**Datasets Used:**

**Smaller Dataset:**

<https://drive.google.com/file/d/1d10zW73C_Vd5JF5PKBXw9vmjV5MSS6C2/view?usp=sharing>

This is the first dataset containing id & content of fake news. We have trained our model for 322 samples.

**Larger Dataset :** <https://drive.google.com/file/d/1WnoSO6U0RSxxgFlG5Sijnq6C1j2GsKQu/view?usp=sharing>

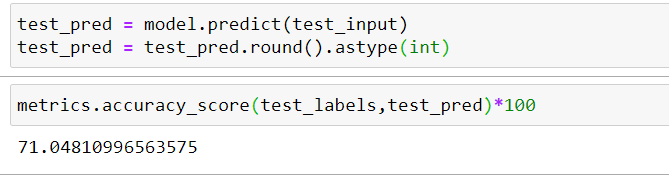
Trained our model for 500 samples. We decreased the labelled data for training and evaluated the performance of the BERT model.

Firstly we loaded the BERT from Tensorflow hub**[1]** using the following command:

module\_url ="<https://tfhub.dev/tensorflow/bert_multi_cased_L-12_H-768_A-12/1>"

Then loaded csv containing ~5800 samples. We used helper functions to encode the input text into tokens, masks and segment ids so as to feed to the BERT model. Now the encoded data was split into training and testing sets. In order to avoid memory allocation errors, we fixed the maximum length of each sentence to be 160.

Finally, we train the model and obtained accuracy of ~71%



**3 Results and Discussion**

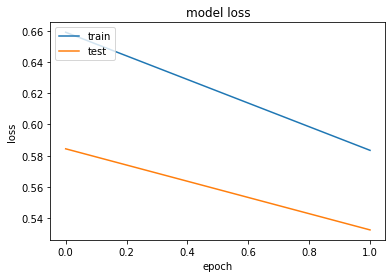
**Smaller Dataset:**

Training data size=332

Epochs=2

Bert model=<https://tfhub.dev/tensorflow/bert_multi_cased_L-12_H-768_A-12/1>

With the above parameters, the BERT model gave an accuracy of ~80% on the test dataset.



**Larger Dataset:**

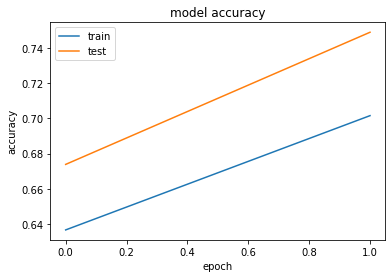
Training data size=4654

Epochs=2

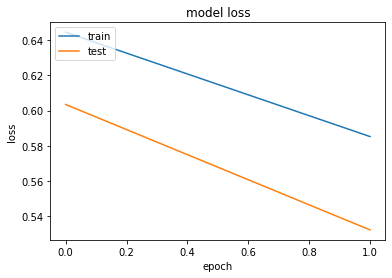
Bert model=<https://tfhub.dev/tensorflow/bert_multi_cased_L-12_H-768_A-12/1>

With the above parameters, the BERT model gave an accuracy of ~71% on the validation dataset.

**Training and validation accuracy**



**Training and validation loss**



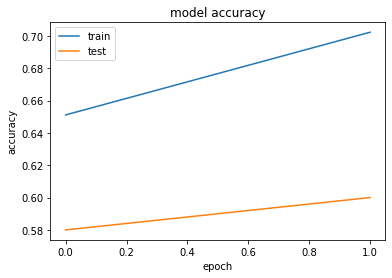
Training data size=500

Epochs=2

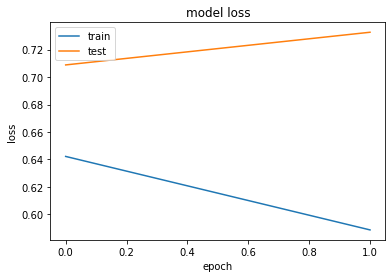
Bert model=<https://tfhub.dev/tensorflow/bert_multi_cased_L-12_H-768_A-12/1>

With the above parameters, the BERT model gave an accuracy of ~68% on the test dataset.

**Training and validation accuracy**



**Training and validation loss**



From the above results, we notice that when training data size is reduced from the large dataset, the accuracy also reduced. However, the goal here is to build an efficient model that gives good accuracy with less training data using few shot learning methods.

**4 Future work**

Many of the meta-learning and reinforcement-learning algorithms are combined with typical deep learning algorithms to produce remarkable results**[2][3]**. Prototypical networks are one of the most popular deep learning algorithms, and are frequently used for this task. Now, to move ahead in this project we will be using one of the few shot learning models and compare the results with the BERT model(baseline).

**5 References**

[1]<https://tfhub.dev/tensorflow/bert_multi_cased_L-12_H-768_A-12/1>

[2]<https://blog.floydhub.com/n-shot-learning/>

[3]<https://github.com/sudharsan13296/Hands-On-Meta-Learning-With-Python>