1 Part I

You have been given daily data on US technological stocks (Amazon, Intel and Facebook) from 13/02/2015 - 12/02/2020, obtained from Yahoo Finance. There is one .csv file for every stock available in Keats for you to download.

Given the above dataset, students need to use a neural network of their own choice, to forecast whether the stock price of Facebook, will go up or down in the next day of trading, given data from the previous 7 days of trading. As predictors, students need to use information on the volume and stock returns of all stocks. Further, students need to create the predictive variable which will be 1 if the stock return is positive and 0 otherwise.

Hint: This exercise is similar to the one we covered during our LSTM tutorial. Your report should contain the following information:

In no more than two pages, students need to discuss the main components of the Neural Network they are going to use and the reason why. Specifically, the Architecture of their choice, the activation function to be used and the Cross-Validation scheme to be employed.

In a .py or .ipynb notebook please follow the following steps:

1. Import the necessary libraries.
2. Read the different datasets, using pandas.
3. Create the train-validation-test split. Specifically, use as

   train_start_date="2015-04-28"
   train_end_date="2017-12-31"
   val_start_date="2018-01-03"
   val_end_date="2018-12-31"
   test_start_date="2019-01-02"
   test_end_date="2020-01-31"

4. Get the returns for every stock and standardise both the returns and the volume.
5. Create the forecasting variable, i.e. what we called label in the tutorial. Create a new dataframe that contains all the predictors and the predictive variable.
6. Create the training, validation and testing generators using the above dates.
7. Create the neural network (in Keras) and choose the relevant hyper-parameters and
8. Follow the Cross-Validation scheme that you defined above, and forecast

using the optimal parameters from CV. For this you can use the random search function from the tutorial but you can also use your own written one. The code should be well documented.

9. Comment on your findings.

For further information please see below the Disclaimer.

2 Part II

From KEATs download the Huff_news.csv file which contains news articles from Huffington Post (HuffPost) (data set obtained from Kaggle public data repository). This data set has about 25,000 articles and 27 different categories. Students are encouraged to use the text of their choice, though.

Using the above dataset evaluate three text classifiers of your choice on a news categorization problem. In no more than a page, please discuss feature representation, feature weighting, and evaluation of the algorithms.

In a .py file or ipynb notebook please proceed as instructed below:

1. Import all necessary libraries.

2. Read the dataset on a dataframe. The text of each article is contained in column 'short description' and the headline is the 'headline' column. The different categories (i.e. classes) are in column 'category'.

3. Merge the 'headline' and the 'short description' columns and create a new column with the resulting text. Name that column 'text_all' (hint: this column is your feature space).

4. Split dataset for training and testing. 25% of the dataset should be used for testing based on our evaluation strategy and the remaining will be used for training the classifier.

5. Prepare the features using tf-idf feature weighting scheme.

6. Train the models.

7. Show the top three predictions according to their probabilities for all models.

8. Provide the accuracy of your classifiers using an evaluation metric.
9. Which algorithm will you choose to classify the articles? Briefly explain your answer.

Note : provide the code on .py or ipynb notebook and your answers of the questions on a word file. The deadline for the completion of the pro ject work is 10:00am 12/04/2023 You can work in a team of up to five students but that needs to be clear in the beginning of the document submitted.

Disclaimer: Copying code without mentioning the original source is forbidden and it counts as plagiarism. However, you can reuse codes provided that you state the original source in your code documentations. In the case of code-reuse, you must be able to verify its output and explain the algorithms that is implemented by the reused code. You would not get any credit, if you fail to answer these questions.

The codes must be modularized. It must be accompanied by proper documentation, i.e. for most of methods the input, output and functionality must be clearly stated in the source code. Your report must state your general findings, justify your approach, and report an evaluation.

Obtaining assistance and teamwork is acceptable and encouraged. However, please be informed that plagiarism won't be tolerated.