ACADEMIC YEAR 2023-2024

Coursework Administrative Details

Module/Lecture Course:	Natural Language Analysis
Deadline for submission:	26th April 2024
Deadline for marks and feedback to be returned to students:	20th May 2024
Submission instructions:	Submit all files via Ultra
Submission file type(s) required:	PDF or Word for report; .zip for notebook.
Format:	Report as a Word or PDF document. Accompanying data analysis for individual report as a Jupyter notebook, compressed in a .zip file. Do not put your name on your report, just your username.
Contribution:	The report contributes 100% to the final mark for the module.

In accordance with University procedures, **submissions that are up to 5 working days late will be subject to a cap of the module pass mark**, and **later submissions will receive a mark of zero**.

This does not apply for resit submissions as a 2nd attempt. For such resits, late submissions receive an automatic mark of zero.

Content and skills covered by the assignment:

- Have a strong understanding of how to work with text and transform textual features to numeric features.
- Have a good understanding of advanced deep learning models for classifying and processing text.
- Select and implement appropriate feature extraction techniques from text.
- Train and test machine learning and deep learning classification models using real-world data.
- Effective written communication
- Planning, organising and time-management
- Problem solving and analysis

General Requirements

Students are expected to work on the coursework individually.

Students will work on the task of SPAM SMS detection using the dataset provided. The dataset has a total of 4,827 SMS legitimate (HAM) messages and a total of 747 SPAM messages. The body text is annotated by the following two classes: SPAM, HAM.

The data suffers from an imbalance problem where around 86.6% of the SMS are HAM.

Students are expected to:

1- Implement word embeddings using standard semantic based techniques and neural techniques.
2- Understand the challenges of the provided problem and suggest solutions to handle the imbalance nature of the data.
3- Implement natural language processing models and classifiers to predict the right category of a given test example.
4- Implement a text generation model to generate text for a chosen category and then use the generated data to test the implemented NLP classifier.

Individual Report [100%]

Each student should separately develop their own NLP models to classify SMS messages into one of the two categories. Write a report (max 1,500 words) on your findings, which will be assessed as follows:

1) Apply the following feature extraction techniques to generate one fixed-length vector for each document in the dataset. Explain how the two following techniques work, and discuss their advantages and disadvantages [**20%**]
   a) Term Frequency-Inverse Document Frequency (TF-IDF)
   b) Word2vec
2) Use both sets of features extracted from the step above to train a standard Machine Learning classifier (e.g., SVM, Naïve Bayes, or Random Forest), and discuss its performance on the testing set. [**15%**]
3) Train a Deep Learning model (e.g., LSTM, GRU, or CNN), using both sets of features extracted in step 1. Explain and justify the architecture of the deep learning model, the hyper-parameters used, and the loss function [**20%**]
4) Analyse, compare, and discuss the performance and training time results for both ML and DL models. [**10%**]
5) Build a text generation model using a Recurrent Neural Network (LSTM or GRU) to generate new “SPAM-like” e-mails using the training data of the **‘*SPAM’*** class, and explain how it works. Use it to generate 100 samples. [**15%**]

6) Use the 100 generated samples from step 5 to test the performance of the machine learning and deep learning models developed in steps 2 and 3. Report and discuss the results. [**10%**]
7) Academic English writing [**10%**], with good use of technical vocabulary, correct grammar, appropriate document structure and referencing where relevant.

The report should use diagrams, figures, and tables to demonstrate the results and analysis.
You should submit your 1,500-word report and also the associated Jupyter notebook used to produce your analysis and graphs.

The report word count should:

- *Include* all the text, including title, preface, introduction, in-text citations, quotations, footnotes and any other item not specifically excluded below.
- *Exclude* diagrams, tables (including tables/lists of contents and figures), equations, executive summary/abstract, acknowledgements, declaration, bibliography/list of references and appendices. However, it is not appropriate to use diagrams or tables merely as a way of circumventing the word limit. If a student uses a table or figure as a means of presenting his/her own words, then this is included in the word count.

Examiners will stop reading once the word limit has been reached, and work beyond this point will not be assessed. Checks of word counts will be carried out on submitted work. Checks may take place manually and/or with the aid of the word count provided via an electronic submission.

Students are strongly advised to use Arial font size 12 for their assignments.

PLAGIARISM and COLLUSION

Students suspected of plagiarism, either of published work or work from unpublished sources, including the work of other students, or of collusion will be dealt with according to Computer Science and University guidelines.

Please see https://durhamuniversity.sharepoint.com/teams/LTH/SitePages/6.2.4.aspx **and** https://durhamuniversity.sharepoint.com/teams/LTH/SitePages/6.2.4.1.aspx **for further information**