Coursework 3 – Deep Learning

Crowdfunding has been a popular way for many, from people to big companies, to fund risky, uncertain, or otherwise obscure projects. The risk in a crowdfunded project comes from the inability of the project controllers to deliver what they promise; and given the huge rise in low quality "projects" requesting funding, the ability of the platforms being able to detect when things start to go awry is becoming a necessity.

In this project, you are given a dataset of comments arising from two well-known crowdfunding platforms (Kickstarter and IndieGoGo), along with the sentiment of the comment. These comments have been tagged by real people, so there is also a confidence score associated to it. The variables are:

- ID: Comment ID (not predictive)
- Text: Comment in text format. Scraped from the web.
- Sentiment: Either positive (1) or negative (0).
- Confidence: How confident was the scorer on the sentiment score.

Starting from this sample, your task is to create a model that infers the sentiment from the text comments and design the way the company will put it into production, using what you know about Deep Learning and Big Data technologies. For this purpose, write a report that answers the following questions:

1. Preprocess the text so that it can be used in Deep Learning. Explain your decisions and describe the resulting datasets.
2. Study the distribution of the dataset and extract basic statistics from the document. At the very minimum, discuss the words that are repeated the most, the length of the phrases, and the distribution of the comments.
3. Train several Neural Networks to predict whether a comment is speaking in a positive or negative way about a project. Try at least two different embeddings (e.g. fastText, GloVe, BERT, etc.) as input layers, with one of them being training your own embedding; and two different architectures, one of them including more than one hidden layer (i.e. not Kim's model). Discuss and justify your choice of the architecture, layers, and other parameters for each of the architectures. Comment on how you think it is best to use the confidence score.
4. Calculate your accuracy, AUROC, and your confusion matrix over the test set for each of your models. Generate a plot with all the ROC curves of your models. Which embedding works best? Which architectures?

In terms of software, use Python, and Excel as needed. Carefully report the various steps of your methodology and discuss your results in a rigorous way! Attach as appendices your Google Colab or Jupyter Notebook printed to PDF.

Conditions of the coursework

Software: You must use Python to run the numerical calculations over your portfolio. A copy of your jupyter notebook must be attached to the coursework as an appendix in readable format, and a link to the notebook must also be included. Instructions how to export to PDF can be found here: https://stackoverflow.com/questions/52588552/google-co-laboratory-notebook-pdf-download

Word Limit: 2000 words +/-10% either side of the word count is deemed to be acceptable. Any text that exceeds an additional 10% will not attract any marks. The relevant word count *includes* items such as cover page, executive summary, title page, table of contents, tables, figures, in-text citations and section headings, if used. The relevant word count *excludes* your list of references and any appendices at the end of your coursework submission.

You should always include the word count (from Microsoft Word, not Turnitin), at the end of your coursework submission, before your list of references.

Title/Cover Page: You must include a title/ cover page that includes: your Student ID, Course Code, Assignment Title, Word Count. This assignment will be marked anonymously, please ensure that your name does not appear on any part of your assignment.

Submission Deadline: December 12th, 23:59.

Turnitin Submission: The assignment MUST be submitted electronically via OWL. All required papers may be subject to submission for textual similarity review to the commercial plagiarism detection software under license to the University for the detection of plagiarism. All papers submitted for such checking will be included as source documents in the reference database for the purpose of detecting plagiarism of papers subsequently submitted to the system. Use of the service is subject to the licensing agreement, currently between The University of Western Ontario and Turnitin.com (http://www.turnitin.com).

Late Submission: Late submissions are possible up to a week after deadline. There is a 10% penalty per day of late submission subtracted directly from the final mark. Submissions after the 7 days are not accepted and will be considered a non-submission.