Capstone Project: **Text Classification: Comparative Analysis of Different Deep Neural Network Architectures**

Description:

In this project, you are expected to develop and compare three different deep neural network (DNN) architectures for the task of text classification. The following three architectures should be explored:

1. **CNN: Convolutional Neural Networks**
2. **RNN: Recurrent Neural Networks**
3. **HAN: Hierarchical Attention Networks**

You are also expected to use Word vectors generated by Google's GloVe as an underlying data model:

- **https://nlp.stanford.edu/projects/glove/**
- *"GloVe is an unsupervised learning algorithm for obtaining vector representations for words (similar to Word2Vec). Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space."*

Why text classification?

- Text classification is one of the most important Natural Language Processing & Supervised Machine Learning tasks in different business problems.
- Example business applications include but not limited to:
    - Understanding audience sentiment from social media
    - Detection of spam & non-spam emails
    - Auto tagging of customer queries
    - Categorization of news articles into predefined topics

Submission Requirements:

Your submission must include:

- Jupyter notebooks (ipynb) and the corresponding html files for each of the DNN architectures: CNN, RNN, and HAN
    - You may choose Keras or PyTorch for implementation
    - Implementation must be Python-based
- Power point slides and the corresponding PDF files illustrating the final DNN architecture
- Data sets or the links pointing to where download the data sets from
- Report with the results supporting your comparative analysis: Include the following information and metrics at a minimum:
    - Description of the specific text classification problem of your choosing
    - Description of the data sets
    - Summary table of the data set sizes: train, validation, test
    - Architectures: visual graphs

- Architecture hyperparameters: Table
- Training and Validation Accuracy and Loss over Epochs: line graphs
- Time/Epoch (min.) bar graphs
- Hyperparameter Tuning: Choices, rationale, observed impact on the model performance
- If model training takes more than 10 minutes, then you should save your models and include the saved models with the submission and provide the code to load your models

Useful Resources:

- CNN: http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/
- RNN: http://colah.github.io/posts/2015-08-Understanding-LSTMs/
- HAN for Text Paragraphs and Documents: https://arxiv.org/pdf/1506.01057v2.pdf
- HAN for Text Classification: https://www.cs.cmu.edu/~diyiy/docs/naacl16.pdf
- https://towardsdatascience.com/understanding-neural-networks-from-neuron-to-rnn-cnn-and-deep-learning-cd88e90e0a90
- https://machinelearningmastery.com/cnn-long-short-term-memory-networks/