**Capstone Project: Classifying clinically actionable genetic mutations**

**Part 3 Check-in – Progress Report (Cheong Yu Chye)**

1. Do you have data fully in hand and if not, what blockers are you facing?

Yes, I have all the data fully in hand.

1. Have you done a full EDA on all of your data?

Yes, I believe so.

1. Have you begun the modeling process? How accurate are your predictions so far?

Yes, I have begun the modelling.

Baseline accuracy (based on majority class) is 0.28696.

Baseline model:

* Based on nltk POS tokenisation, Extra Trees Classifier trained on CountVectorizer word embeddings
* Has a validation accuracy score of xx.
* Has OVO (one-vs-one weighted) validation AUC score of xx.

Alternative model:

* Based on nltk POS tokenisation, Extra Trees Classifier trained on Word2Vec word embeddings using a Mean Embedding Vectorizer.
* Has a validation accuracy score of 0.66426
* Has OVO (one-vs-one weighted) validation AUC score of 0.919691.

1. What blockers are you facing, including processing power, data acquisition, modeling difficulties, data cleaning, etc.? How can we help you overcome those challenges?

* **Long execution times**.:
  + My models take a long time to fit (or causes kernel crashes) due to the many features (~77,000) – 10+ hour execution times are not uncommon. I’ve had to deliberately simplify or shorten processing by reducing the number of cross-validation folds (use default of 3), and also the number of concurrent jobs (set at n\_jobs=4 instead of n\_jobs=-1), for example.
  + I am experimenting with ELMo contextual word embeddings right now, but face an uphill task as my own laptop (Intel UHD 620 GPU) cannot support keras/tensorflow. As a workaround, I am exploring cloud-based computing via the use of the Google Colab Pro environment. By paying a small subscription fee, I can set a notebook with high memory and GPU/TPU. So far processing has not crashed, but it still takes a very long time to generate ELMo word embeddings.
* **Modelling difficulties**. I was hoping to explore how my model performance could improve using contextual word embeddings such as BERT and BioBERT, as they are state of the art. However I recently found out that they are limited to sentence sizes of 512 words, whereas my training data that has 500k+ words in some sentences. I do not view truncating or splitting my text into 512 word (or fewer) chunks as feasible due to the complexity involved and very high resource requirements (CPU and memory). I have given up on BERT and BioBERT and am currently focusing on alternative contextual word embeddings such as ELMo.
* **Kaggle scoring**. The Kaggle scores (multi-factor loss) for my submissions appear to always be the same – I am not sure why. As such, I cannot rely on the Kaggle scores to measure my performance, but will instead choose between my baseline and alternative model using by comparing AUC and accuracy scores based on my validation dataset.

1. Have you changed topics since your lightning talk? Since you submitted your Problem Statement and EDA? If so, do you have the necessary data in hand (and the requisite EDA completed) to continue moving forward?

No, I’ve not changed topics since my lightning talk, or submission of problem statement and EDA.

1. What is your timeline for the next week and a half? What do you have to get done versus what would you like to get done?

For the next week and a half, I plan to:

* Finish up on my baseline model analyses and visualisations
* Complete my assessment of how ELMo word embeddings can improve the performance of my alternative model

1. What topics do you want to discuss during your 1:1?
2. My dataset for modelling consists of:
   1. Word embeddings/vectors – of which the magnitude has a significance. Examples are CountVectorizer output
   2. One-hot encoded columns

My question is whether we need to apply StandardScaler to this dataset before the start of actual modelling?

1. I originally thought of manually increasing the weights of the word vectors by:
   1. Searching external websites (e.g. ClinVar) for related terms based on the gene and/or variant) into training dataset
   2. Increasing the weights of any such related terms that I manage to find within the given expert text

However, I currently lack the time and know-how of how to do this effectively.

1. The diagram below shows my overall workflow:
   1. Is my scope of work sufficient for a capstone project?
   2. Do you have any suggestions/comments on the workflow?

