Project:

Text classification competition

Progress:

I was able to achieve an accuracy of 0.77 with my initial implementation which includes pre-trained language model along with pytorch neural network framework for fine-tuning the model to fit to the sarcastic dataset. However I forgot to set the seed values due to which I couldn't reproduce the results although the output predictions were saved. With a specific seed set, I am able to reproduce the output predictions with accuracy ~0.765, almost close to my max.

I have used Google-Colab free version for training the model so it comes with its own limitations. I am kind of struck on achieving maximum accuracy of 0.77 (currently (as on 25-Nov) best on the leaderboard) on the hidden dataset, although the models performance is good enough (~0.86) on the hold-out validation set created from the overall training dataset.

I will try to use few more derived heuristic variables along with pre-trained models to see if that helps in improving the performance a bit more.

Remaining tasks:

- Need to do a thorough documentation of the code
- Create tutorial presentation as part of project deliverables

Challenges/Issues:

- Limited GPU memory is allocated in Google-Colab free version. Layering the pre-trained models with any more slightly complex or deep neural layers is resulting in out of memory issues. Had to settle with shallow layers.
- Struck within a performance range of 0.7 to 0.77 with the approach listed above. Need to come up with a different approach to improve the performance while working with limited memory