Project Proposal

1.1 Team Information

Name	Ningyuan Zhang	Jacky Sa	Sharanya Balaji
NetID	nz13* Captain	mjsa2	sbalaji3

1.2 Topic Introduction

Our topic is Fake news detection. The goal is to develop an RNN model and various model improvements such as GRU, Transformers, LSTM to predict the fakeness of a news article.

Importance of the topic:
With the availability of news exploding across social media, it is very important to filter out fake news to prevent their damage on society. We believe that deep learning techniques can leverage the sequence of words to model the likelihood of a news being fake. This relates to the topic of text analysis and classification as taught in the class.

Approaches:
The planned approach is to preprocess the text, use general word embeddings, and start with a general RNN model architecture. We will then add complexity and improve the model to address any memory issues, such as using LSTM, GRU and transformers.

After training, evaluating and refining the model, we may develop a simple app to let users enter news articles and our app will detect its fakeness.

Tools/Systems/Datasets:
Python/Pytorch/BERT/Transformers
Fake News labeled dataset from kaggle:
https://www.kaggle.com/clmentbisaillon/fake-and-real-news-dataset

Expected Outcome:
Well-trained RNN model to detect fakeness of any given news and an app to leverage the model

Model Evaluation:
Precision, Recall, AUC Score, Confusion Matrix

Programming language:
Python

1.3 Workload Justification

We have 3 people => 3x20 = 60 hours.

Main task:

1. Exploratory data analysis
   To have an overview of how the dataset looks, including n-gram analysis for text attributes, distribution analysis for targets(balanced or not). **5 Hours**

2. Feature Engineering:
   In this case most of the work would be text preprocessing and text embedding. We will explore several general pre-trained word embedding models and compare the performance. **10 Hours**

3. Model Architecture:
   We will develop a simple RNN model and gradually increase the model complexity(LSTM, GRU, BERT, Transformers). **25 Hours**

4. Modeling Training and Tuning Parameters: **15 Hours**

5. Model Evaluation:
   Choose different metrics to compare the performance of different architectures on this dataset and finally choose the best one. **10 hours**

6. Extension to Real Applications:(if have time)
   Build a simple app to help detect the fakeness of a user inputed news, alongside other potential UI features. **10 Hours**

~75 hours total