# ECE1786 Creative Applications of Natural Language Processing

## PolitiTrend - Predicting Political Leanings from Text

## Final Report

Feifan Li, Ge Jin

word count: 1975, penalty: 0%

(Last page is the permissions)

## 1. Introduction

In society, through social media, people can express and convey to each other their views on certain hot issues, with text being the most widely used. Our project aims to use natural language processing techniques to predict the political stance or leaning of a text message. The group was inspired by websites that determine political leanings by answering questions and wanted to do this through NLP.

Our motivation is the following:

- For personal considerations, it is convenient for people to understand their own or others' political positions through their texts
- For politicians and public figures, this tool may help them to quickly understand the public's political attitudes on certain issues, and thus help them to better adjust their strategies.

We split the political positions into two different directions based on the example(Figure1):

1. Y-axis: Regulationism and liberalism in the economic sphere
2. X-axis: Progressivism and conservatism in the political and cultural sphere.

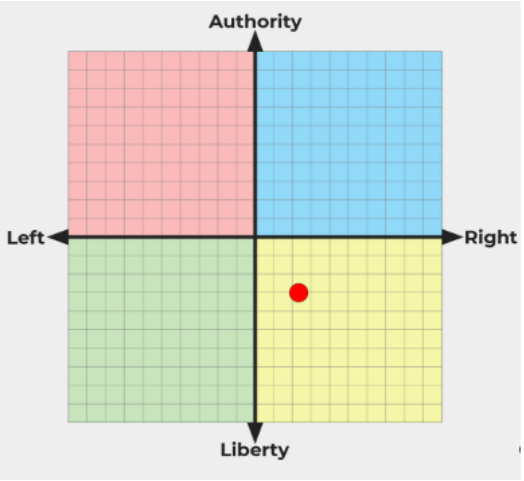

Figure 1: an example of an online ideology test

This means that we need to train two different models for two spheres, each one model will be used to classify a corresponding sphere and get one of the two doctrines in this sphere.

## 2. Background & Related Work

Since this is a project on classification, we drew on research from other classification projects. We found a paper summarizing the use of large language models regarding sentiment classification, whose main use is to help us with data processing. This will be reflected in the data processing section.

Nandwani et al.[1] summarized the previous works on sentiment classification. For example, the paper suggests that unnecessary words such as papers and prepositions that do not contribute to sentiment recognition and sentiment analysis must be removed. For example, stop words like "is", "at", "an", and "the" are irrelevant to sentiment analysis. This paper is not about political leaning prediction though, But it contains several scenarios that we can learn from for the classification work we will be doing.

## 3. Data Processing

The group got data from this website: https://manifesto-project.wzb.eu/. This website contains the election manifestos of political parties from 69 countries, and the sentences in the documents are labeled with specific numbers to indicate their political leanings. The group categorized these labels into four categories. Because there are so many labels, only a few examples are shown below:

- **Regulationism**: Nationalization, Controlled Economy….
- **Liberalism**: Free Market Economy, Decentralisation,...
- **Progressivism**: Multiculturalism,Internationalism,,...
- **Conservatism**: Traditional Morality, nationalism,...

Regulationism and liberalism are used in the economic sphere, and progressivism and conservatism are used in the political and cultural sphere. For each category, we try to make sure that each of the labels in it has an average percentage of the category. But there are still some labels that make up a large percentage of the category (e.g. nationalism makes up almost 30% of conservatism). This is related to the current general political environment.

For each category, 1000 sentences were used for training and validation and 200 sentences for testing. We organized the sentences in these two spheres into two datasets and we divided the datasets into training and validation sets in the ratio of 8:2. Each dataset was used in the same training process for classification. In total, the group collected 12 CSV files from political parties in countries including the US, Germany, Canada, etc.. Our data came from a wide range of political parties, from extreme right-wing parties such as the French National Front to extreme left-wing parties such as the Communist Party.For foreign language data, we translated it into English using Google Translate in Python.
Based on the suggestion in background research, we performed the following three types of data processing:

1. Lemmatisation
2. Removal of Stopwords (e.g., "the," "is," "and")
3. Removal of punctuation

For example:

| | |
|---|---|
| **We believe in at last guaranteeing equal pay for women.** | We believe last guaranteeing equal pay for woman |

# 4. Illustration

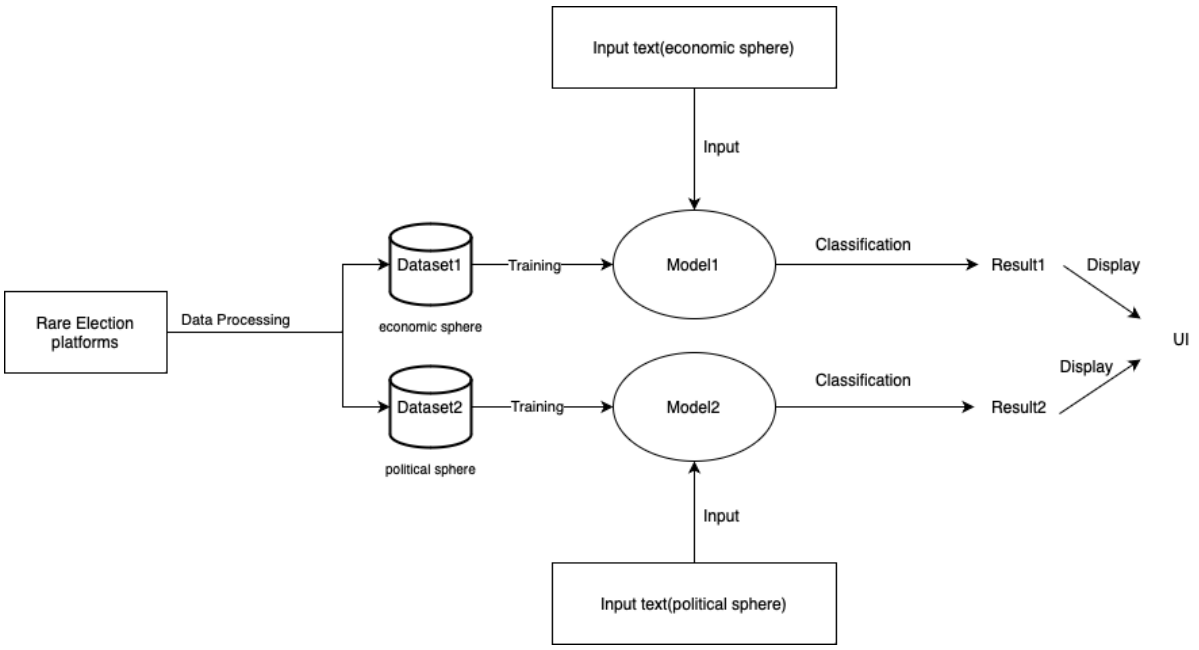


Figure 2: illustrates the entire system. The model1 and model2 are the baseline or fine-tuned transformer models.

# 5. Baseline Model or Comparison Method

Our baseline model, similar in structure to the one used in assignment 2, serves as a reference point for comparing our neural network. It adopts a straightforward CNN design, specifically tailored for feature extraction from political texts. This architecture consists of two convolutional layers, each operating on the word vector group with varying kernel sizes. We intentionally maintain simplicity in this model to grasp the fundamental linguistic indicators of political inclinations. To decode these features and classify political sentiment, we rely on the fully connected layers within the model. During training, we implement early stopping based on validation loss to prevent overfitting.

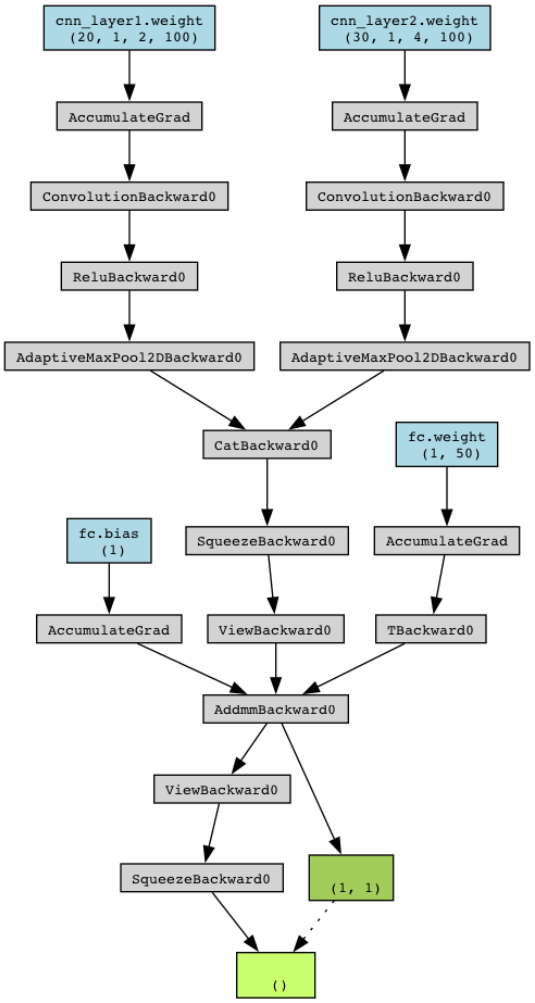


Figure 3: Baseline model architecture

After constructing the model, we initiated a systematic parameter exploration. This involved a thorough investigation of 30 distinct combinations of these parameters, designed to provide a comprehensive insight into their interactions. For each of these combinations, the model underwent a training process, and we evaluated its performance based on metrics such as training, validation, test accuracy, and loss. After a series of careful selections and comparisons, the best parameters have ultimately been documented in the figure. These parameters exhibit exceptional training accuracy and relatively high validation and testing accuracy.

## 6. Architecture and Software

Our "TFBertForSequenceClassification" model, a refined version from Hugging Face, boasts a sophisticated architecture with 110 million parameters, specifically geared toward sequence classification tasks. This model, evolving from the esteemed BERT framework, is uniquely structured to excel in tasks like sentiment analysis and political orientation identification. Unlike traditional BERT models that primarily focus on outputting either class labels or specific spans from the input, our model is enhanced with an additional classification layer. This layer is adept at handling complex classification challenges, making our model a more versatile tool for nuanced linguistic analysis.

The core training strategy involved processing masked statements, leveraging BERT's self-attention mechanisms crucial for extracting complex linguistic features, especially in political discourse. We extensively experimented with epoch counts and batch sizes during training to optimize the model's performance. The best configuration for high-quality headline generation used a batch size of 2, 3 epochs, and a learning rate of 5 x 10^-5. Fine-tuning occurred on a 16GB RAM GPU for efficient training.

Similarly, we fine-tuned a GPT-2 model with the softmax function for sentence classification, aligning methodologies across both models for language processing consistency.

Each model operates as a binary classifier within our UI's four quadrants: regulationism, liberalism, progressivism, and conservatism. The first model assesses political text for progressivism and conservatism, while the second analyzes economic text for regulationism and liberalism. Their outputs help plot positions and distributions in the UI.

## 7. Quantitative Results

We used the test dataset mentioned in the data processing section and obtained the following data. We trained six binary classifiers using CNN, BERT, and GPT-2 models. Each achieved nearly 100% training accuracy. While the CNN model showed promise, all models exhibited potential overfitting issues. To address this, we introduced early stopping in later training stages.

| Model Name | Test Accuracy |
|---|---|
| CNN(Politi) | 0.712 |
| CNN(Econ) | 0.703 |
| BERT(Politi) | 0.732 |
| BERT(Econ) | 0.778 |
| GPT-2(Politi) | 0.761 |

| | |
|---|---|
| GPT-2(Econ) | 0.763 |

Table 1: Test accuracy of individual models

The true measure of a model's effectiveness is its performance on unseen data. Analyzing the testing results revealed that both BERT and GPT-2 performed well, achieving 76% and 75% accuracy, respectively. The CNN model, while slightly behind, reached a respectable 71% accuracy. Notably, all three models excelled in classifying economic (ECON) topics, outperforming political topics by approximately 1-2%. This insight is valuable for guiding the models' future applications, especially in contexts where precise economic topic classification is crucial.

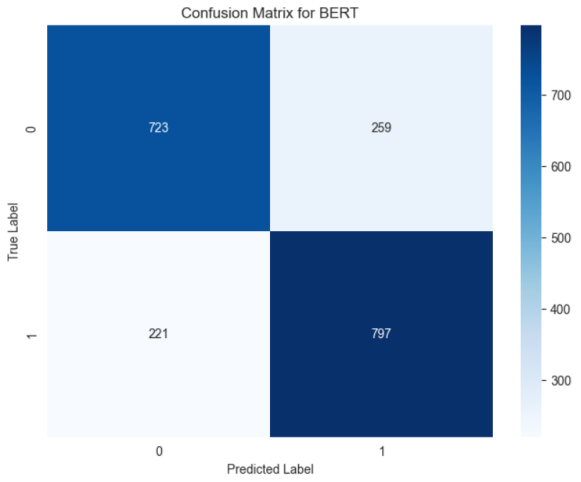


Figure 4: Confusion matrix for BERT

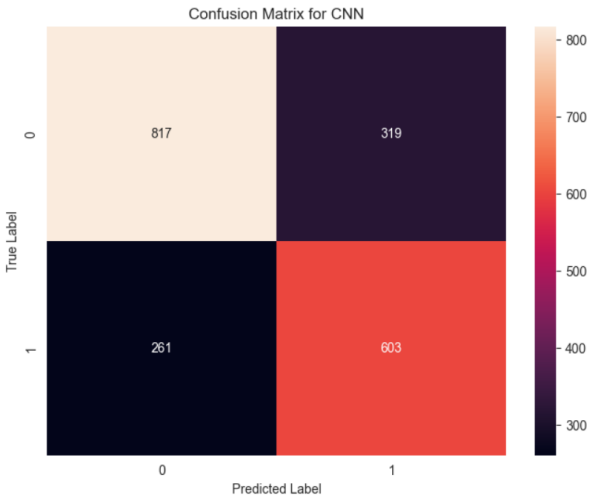


Figure 5: Confusion Matrix for CNN

By analyzing the confusion matrices of our models, we gained a deeper understanding of their performance. The BERT model notably excelled in Precision, Recall, and F1-score. These metrics are essential as they offer a comprehensive view of the model's accuracy and its effectiveness in identifying all pertinent cases.

# 8. Qualitative Results

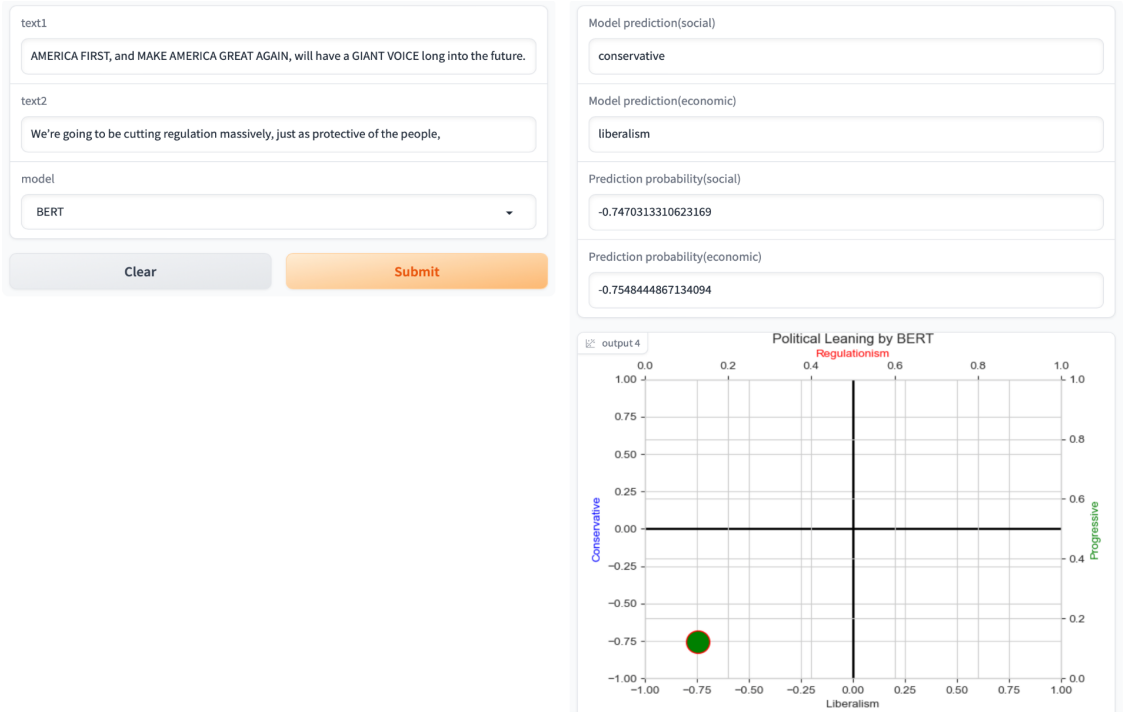


Figure 6: Demonstration of  User Interface result using sentences from Twitter

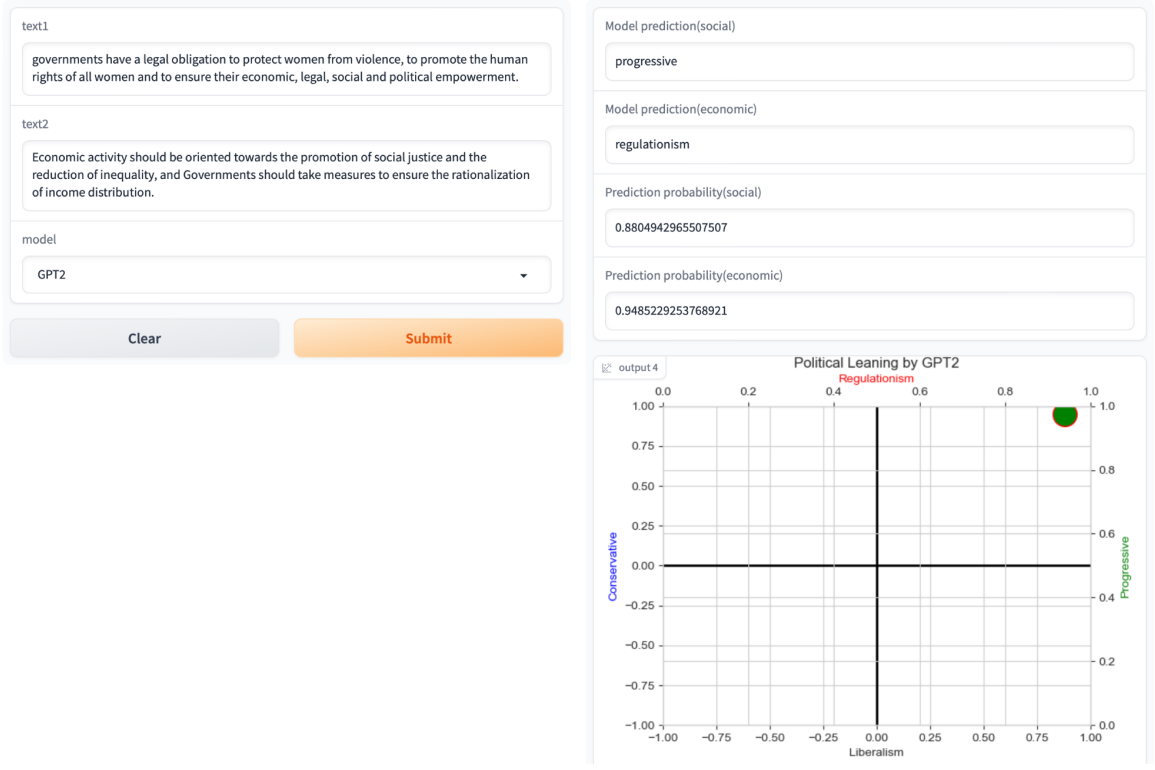


Figure 7: Demonstration of  User Interface result using sentences from the EU parliament

The two figures demonstrate our UI presentation interface. The first input, "text1," pertains to political discourse, while "text2" relates to economic statements. Users can select models from A drop-down menu called “model”. In the first test, using the Bert model, "Make America Great Again" (political) and "Reduce regulations" (economic) yielded conservative

and liberal results, respectively. The second test, utilizing GPT2, analyzed statements on women's rights protection and regulational economy, resulting in progressivism and regulationism. These outcomes align with common perceptions, suggesting accurate predictions.

However, we identified three issues:
1. **Misclassification**: Results occasionally contradict common sense, indicating a need for model training refinement. For instance, statements about increasing welfare were classified as conservative.
2. **Misuse of Softmax Probability Outputs**: The model's category probability shouldn't equate to political leaning degrees.This is because this probability is only a mathematical probabilistic inference and cannot be used to describe qualitative political descriptions. For example, the progressivist tendency regarding women's rights protections is higher than 90%. We do not think this score is justified because the protection of women's rights is somehow a consensus.
3. **Neutral Statement Handling**: The model struggles with neutral statements like "everyone needs to eat," The position of the sentence in the coordinate system is not at the origin.

# 9. Discussion and Learning

## Training

During the training process, we tried different learning rates and training batches. To prevent overfitting, the number of epochs for each model was limited.
Below are graphs of the training results for CNN, BERT and GPT2 in two different domain datasets, respectively. The value of the horizontal coordinate of each plot is the epoch.
**Progressivism and conservatism**

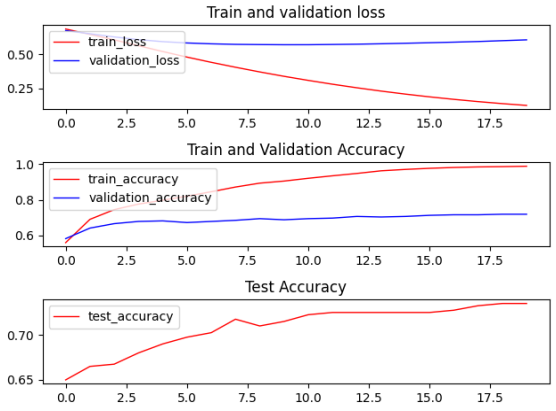


Figure 8: Training/Validation/Test Results of CNN(progressive and Conservative)

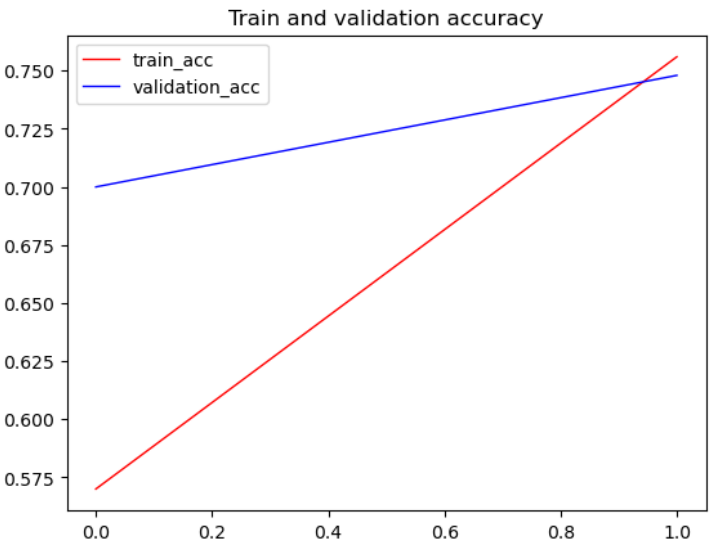


Figure 9: Training/Validation Results of BERT(progressive and Conservative)

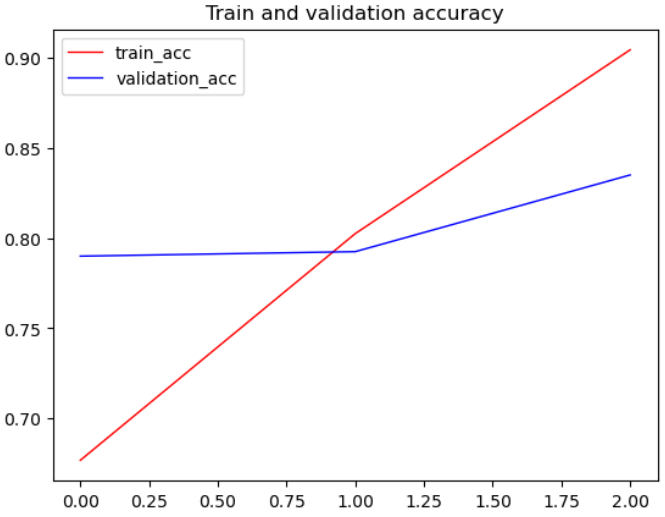


Figure 10: Training/Validation Results of GPT-2(progressive and Conservative)

## Regulationism and liberalism

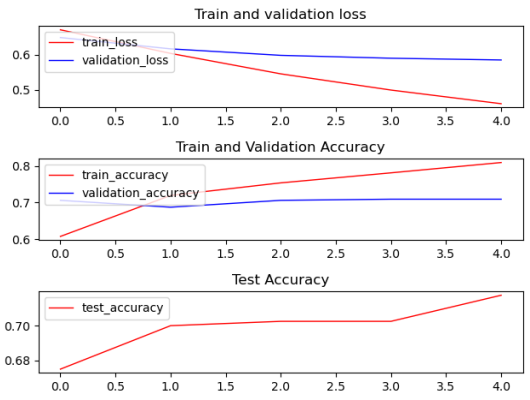


Figure 11: Training/Validation/Test Results of CNN(Reg and Lib)

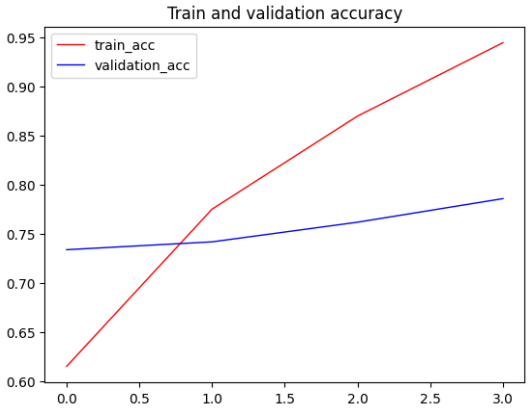


Figure 12: Training/Validation of BERT(Reg and Lib)

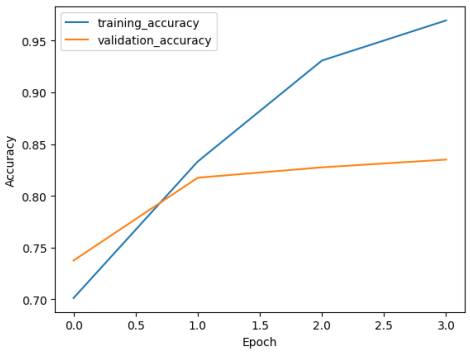


Figure 13: Training/Validation Results of GPT-2(Reg and Lib)

The group controlled the epoch value to avoid overfitting. For example, in Figure 9, our BERT model is trained for only two epochs, so it is a straight line. This is because in the subsequent epochs, the overfitting gets worse and the accuracy of the validation set decreases.

In addition, we also found that in most cases, in the early stages of training, the validation accuracy is higher than the training accuracy, which we believe is due to the large number of samples in the training dataset, resulting in a larger variance in it, which leads to a larger error than the validation set with fewer samples.
What is more, our group found that no matter which model, its testing and validation accuracy can only reach about 70% at most. If the value of the epoch is increased, overfitting is easy to occur. This may require us to increase the size of the dataset and the diversity of the dataset.

## Difficulties in political classification

We encountered three challenges in the project:

1. Political statements pose a unique difficulty in categorization due to their subtle nature, expressed through various forms like insinuation, irony, incitement, and jokes. This necessitates language models with enhanced contextual understanding.
2. Political positions often lack consistency, with both left and right employing similar rhetoric on the same issue. Politicians, driven by pragmatism, make it challenging to infer their true political leanings based solely on their statements.
3. Cultural diversity plays a significant role, as the project focused solely on Western perspectives, neglecting the varied political axes in different countries. For instance, issues like the right to free marriage may differ in political relevance between Western and non-Western societies.

## Suggested Solutions:

1. Diversify the dataset to avoid model homogenization and accommodate various types of political statements.
2. Refine political classifications for a more nuanced understanding, acknowledging subtle differences even among those supporting the same policy.

3. Employ advanced large language models like GPT-4 to gain detailed insights into the political context and speaker nuances, enabling more informed and rational judgments.

## 10. Individual Contribution

Feifan Li:

- Responsible for Collecting and preprocessing the PolitiTrend dataset.
- Responsible for categorizing these labels into four categories
- Responsible for fine-tuning the 'TFBertForSequenceClassification' BERT model
- Responsible for training the 'GPT2' model used for classification
- Responsible for Qualitative Analysis of Model performance
- Responsible for Gradio implementation
- Responsible for the final report

Ge Jin:

- Responsible for categorizing these labels into four categories
- Responsible for constructing the baseline model  and selecting the best parameters
- Responsible for training the 'GPT2' model used for classification
- Responsible for Gradio implementation
- Responsible for Quantitative Analysis of Model performance
- Responsible for the final report

## Reference

[1] Nandwani, P., & Verma, R. (2021b). A review on sentiment analysis and emotion detection from text. *Social Network Analysis and Mining*, *11*(1).

## Permission:

Feifan Li

permission to post video: no

permission to post final report: yes

permission to post source code: yes

Ge Jin

permission to post video: no

permission to post final report: yes

permission to post source code: yes