Birkbeck, University of London

Department of Computer Science and Information Systems

MSc Data Science

Project Report

Natural Language Processing for Political Science: exploring the ability of supervised machine learning models to classify policy sentences

by Guillermo Fremd Kanovich – 13178024

Supervisor: Mark Levene

September 2021

This report is substantially the result of my own work, expressed in my own words, except where explicitly indicated in the text. I give my permission for it to be submitted to the JISC Plagiarism Detection Service. The report may be freely copied and distributed provided the source is explicitly acknowledged.

Contents

ABSTRACT 4
INTRODUCTION 5
MOTIVATION AND RELATED WORK 7
Related Work 9
DATA OVERVIEW 12
METHODOLOGY AND MODELS 17
Naive Bayes with TF-IDF vectors 17
XGBoost with GloVe embeddings 20
BERT 23
MODELS PERFORMANCE 26
Naive Bayes 26
XGBoost 28
BERT 29
Performance in the UK Speeches from the Throne Dataset 31
CONCLUSION AND FUTURE WORK 34
REFERENCES 36
APPENDICES 38
Appendix 1. Most common terms per topic 38
Appendix 2. Distribution of the Train subset 40
Appendix 3. Confusion matrices 41
Appendix 4. Some examples of misclassification 44

ABSTRACT

Speeches and policy documents are an important resource for political researchers, who often use them to analyse trends across time and to make comparisons between countries or political parties. In this regard, the classification of political texts with one consistent coding scheme allows researchers to utilise data classified by other colleagues for their own work. With that aim, the Comparative Agendas Project assembles a collection of thousands of policy-related documents manually labelled by political scientists worldwide, using a consistent coding scheme to label text units according to 22 underlying political themes - such as "Macroeconomics", "International Affairs", "Law and Crime" or "No Policy Content" –. However, the manual classification of text is a slow and costly process, and it represents a significant burden.

Aimed at exploring ways to automate this process, this report presents and discusses the training, fine-tuning and testing of three Natural Language Processing models for the classification of policy-related sentences, according to the Comparative Agendas Project's coding scheme. The three models were trained, fine-tuned and tested using almost 40,000 sentences from the party platforms published by the United States' Democratic and Republican parties between 1948 and 2020, which were randomly split into Train, Validation and Test subsets.

The best results were obtained using a fine-tuned version of the Transformers-based model BERT, which achieved an overall accuracy of 79% in the Test subset, with a macro average f1-score of 0.78. In turn, using TF-IDF vectors, a Multinomial Naive Bayes model achieved an overall accuracy of 70% in the Test subset, with a macro average f1-score of 0.68. The third model, which comprises an XGBoost algorithm using pre-trained 300-dimensional GloVe embeddings, achieved an overall accuracy of 68% in the Test subset, with a macro average f1-score of 0.67.

INTRODUCTION

The ultimate objective of this project was to explore and test some avenues in which Machine Learning and Natural Language Processing could provide value to the analysis of politics in general and policy speech in particular. Specifically, during this project, I trained and fine-tuned a number of supervised Machine Learning models and assessed their ability to correctly identify the major topic behind a large corpus of policy-related sentences.

To put a number of examples, the models I trained were aimed at identifying Education as the major policy topic behind the sentence *"Democrats believe we must have the best-educated population and workforce in the world",* or International Affairs for the sentence *"Democrats believe that diplomacy should be our tool of first resort."* Naturally, the topics of some sentences are not always so clear cut, and in some cases, it may prove challenging even for a human to decide what category to assign. Consider, for example, the following sentence from the Republican Party 2012 platform: *"We insist that there should be no regulation of political speech on the Internet"*; preliminary, it is not evident whether to classify this within the Technology or the Civil Rights categories.

The results achieved during this project show that the combination of Natural Language Processing techniques with Machine Learning algorithms does have the ability to analyse policy-related texts and identify the topic of its content. What is more, our results suggest that, with some degree of fine-tuning, models trained with a corpus of sentences from one context can do a decent job at identifying the policy topic of sentences from a completely different context.

For this project, I used two datasets comprising the classification of almost 40,000 sentences from the United States Democratic and Republican parties platforms, dated from between 1948 and 2020. Both datasets were merged and randomly split into train, validation and test subsets, and three different models were trained, fine-tuned and tested with the same data.

Given that some of the categories – such as Culture or Immigration – had a significantly lower number of sentences than other more prominent topics - such as International Affairs or Macroeconomics –, I supplemented those topics presenting less than 1000 examples in the training subset with examples I took from a third dataset, comprising sentences delivered by US presidents between 1946 and 2020 during the State of the Union speeches.

As expected, given the state-of-the-art results it has obtained in a large number of NLP tasks, the best results were obtained when using a fine-tuned version of Google's pre-trained language model BERT,

which achieved an accuracy of 79% in the test subset. The other two models attempted, Multinomial Naive Bayes and XGBoost, returned 70% and 68% accuracy, respectively.

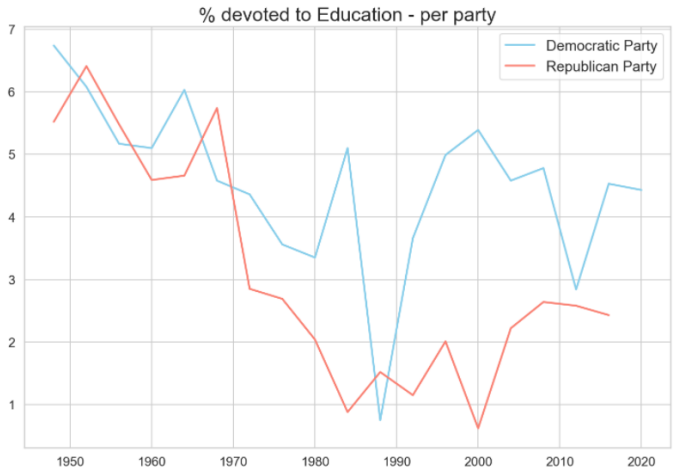
In addition, I also tested the ability of my BERT model to correctly classify sentences from a completely different context: the annual State Opening of Parliament in the UK, also known as the Speeches from the Throne. After fine-tuning it, the same model – trained initially with the data from the US party platforms - achieved an accuracy of 70% in the UK's Speeches from the Throne data.

Following this introduction, this report is organised into five sections. The first chapter, Motivation and Related Work, presents the rationale and relevance of this project and an overview of the most salient work related to my project. Then, the Data Overview presents details about the specific datasets I used for this project. Subsequently, in the Methodology and Models chapter, I present the three models I used– Naive Bayes, XGBoost and BERT –, including the specific details of how these were trained and fine-tuned during this project. Then, in the Models Performance chapter, I present and analyse the results obtained with each model, and finally, the Conclusion and Future Work chapter discusses the overall results of this project, some limitations faced during its development, and some ideas for future research.

MOTIVATION AND RELATED WORK

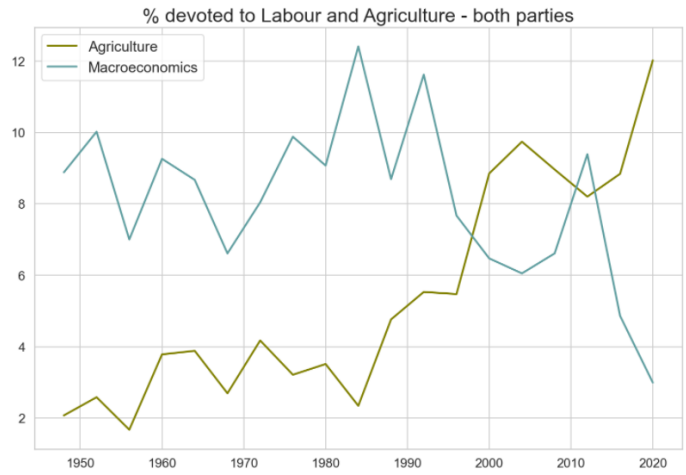
Policy agendas, understood as the range of salient issues that the government and other key public actors concentrate on at any one point in time [1], have been a prominent field of study within comparative politics since the 1960s [2]. To name a few, political scientists have focused on policy agendas to research issues such as variations in presidential priorities [3], the influences of media on the decision-making process of policymakers [4] [5] or parties rhetoric in social media platforms [6].

Consider, for illustration purposes, the following line chart I created with data from the US Political Parties Platforms. It shows the evolution in the proportion of sentences devoted to Education in Democratic and Republican platforms between 1948 and 2020. The chart reveals a drastic drop in the attention put on Education matters by both parties between the late '40s and the late '80s. Also, it shows a radical increase in the attention put to Education by the Democratic party between the late '80s and the year 2000.[1]



Similarly, the line chart below, which I also created with data from the US Political Parties Platforms, shows a consistent increase in the interest put in Agriculture matters between 1948 and 2020 (from 2% to 12%, a six-fold increase), and a significant reduction in the proportion of sentences classified as "Macroeconomics", from 9% to 3%, during the same period.

[1] The dataset comprising the labelled sentences for the Democratic Party Platforms are available only until 2016.



While giving context and analysing the trends shown in the charts above is, obviously, not within the scope of this project, they do show the kind of analysis that is facilitated by having labelled policy data such as speeches, political platforms, and others. What is more, the potential that labelled data offers to political scientists and other researchers, increases exponentially if a consistent labelling system is used in a vast number of investigations, allowing the data compiled by them to be readily comparable.

In this regard, the Comparative Agendas Project (CAP) is probably one of the most comprehensive and robust resources available to social scientists interested in public agendas. The Comparative Agendas Project is a vast library of datasets produced by political scientists worldwide, all of whom have labelled documents using a single and consistent coding scheme, according to the topics they address. These datasets include the labelling of thousands of political speeches, party platforms, congress bills, and more, originated in several countries and comprising several decades.

The Comparative Agendas Project coding scheme, the one used by all the datasets available in the Comparative Agendas Project website, comprises 22 Major Topics (such as "Macroeconomics", "Education", “Defense”, or “No Policy Content”) and 200 Sub Topics (such as "Monetary Policy", "Youth Employment" or "Elementary & Secondary Education"). While some datasets include just one Major Topic and one Sub Topic per document, other datasets include the coding of several sections per document, such as the labelling of each paragraph or each sentence.

Given that the Comparative Agendas Project datasets comprise data from multiple countries and for many decades, it enables both intra-country research – such as the analysis of changes in policy agendas across time, or comparisons between the policy agendas advanced by different political parties,– as well as the comparison between policy agendas in different jurisdictions.

Datasets available on the Comparative Agendas Project website include, among many others, the following:

- the US Congressional Bills dataset, which contains the labelling of 400,000 bills introduced in the US Congress between 1947 and 2016, according to the Major Topic and Sub Topic they address;

- the Media El Pais dataset, which includes the labelling of 56,540 news stories published in the front page of Spanish newspaper El País between 1996 and 2011, according to the Major Topic and Sub Topic they address;

- the Brazil Presidential Investiture Speeches dataset, which comprises the labelling of 517 sentences pronounced in their investiture speeches by newly elected presidents of Brazil between 2003 and 2011, according to the Major Topic and Sub Topic they address;

- the Italy Primary Laws dataset, which includes the labelling of all primary laws adopted by the Italian parliament between October 1983 and February 2013, according to the Major Topic and Sub Topic they address.

However, all these datasets have been manually labelled by human coders, which involves a lengthy and costly process, not only due to the effort needed to code the actual documents but also because of the time needed to train the coders themselves. This makes the labelling of new corpuses a costly endeavour, which represents a real bottleneck for researchers.

Automating the coding of documents with a high level of accuracy would open a wide array of opportunities for researchers, dramatically reducing the times and costs that manual labelling involves. Accordingly, the motivation behind this project was to explore ways to advance the automation of this process, with the ultimate aim to facilitate the labelling of policy-related materials by political scientists and analysts.

Related Work

In 2012 Collingwood and Wilkerson [7] trained and tested four supervised learning methods to classify all bills included in the US Congressional Bills dataset mentioned above, comprising the labelling of 400,000 bills introduced in the US Congress, according to the Comparative Agendas Project's Major Topics. They used the following four methods: SVM, Maximum Entropy, Naive Bayes and LingPipe (N-gram character language model). The two algorithms that presented the best performance, SVM and Maximum Entropy, obtained an average accuracy of 79%, and Naive Bayes achieved a similar accuracy of 75%

It is relevant to note that, while Collingwood and Wilkerson worked with the same coding scheme I used for this project, the nature of the units of text labelled by them was significantly different from those I worked with. While the Congressional Bills dataset comprises the labelling of entire congressional bills (i.e., just one label per bill, each of which comprises hundreds or thousands of words), I worked on the classification of sentences from US party platforms which comprise, in average, just 21.3 words. Details about the dataset I have used are included in the Data Overview chapter below.

What is more, congressional bills comprise whole, independent documents whose general understanding is not dependent on the content of other documents. In turn, the classification of short sentences that are part of a larger document -party platforms, in this case- presents additional challenges, given that the meaning of a sentence is often dependent, or at least related, to the content of sentences that preceded it. As a result, this project's nature is significantly different from the one developed by Collingwood and Wilkerson.

In 2016, Zirn et al. [8] attempted to develop an automatic solution for classifying political party platforms according to the seven Domains (major policy areas) of the Manifesto Project's coding scheme. While similar in nature to the Comparative Agendas Project, the Manifesto Project also comprises an ever-growing collection of documents manually labelled by political scientists worldwide, but it is restricted to political parties' manifestos and programs. It does not include other materials such as speeches, bills and media articles. Similar to the Comparative Agendas Project's, the Manifestos Project codification schema comprises two types of codifications for political discourse: Domains (seven categories) and subdomains (56 categories).

Zirn et al. used the Republican and Democrat party platforms from 2004, 2008, and 2012 elections to train, validate and test their model, which combined three classifiers: one predicting the topic for the sentence being classified, one predicting the topic of the previous sentence, and another to detect topic-shifts between the two sentences. Their solution, which is an ensemble model, comprises the following elements:

1- For predicting the topic of the sentence being classified and the prior one (the "local classifier"), they trained a linear Support Vector Machine (SVM), using lexical and numerical features of the sentence, such as the term-vector and GloVe embeddings;

2- To determine whether two adjacent sentences belong to the same topic or not (i.e., to predict if a change in the topic took place between the two sentences), they used a Radial Basis Function (RBF) SVM. The SVM is trained with the following variables: the length of

each adjacent sentence, the semantic similarity between them (calculated as the cosine similarity between a GloVe embeddings of both sentences), and the overlap of words between the two sentences.

Finally, the predictions of the local classifier for both sentences are combined with the topic-shift prediction, training a probabilistic graph model, Markov Logic Network. They obtained a macro precision of 77.5% and a weighted precision of 79.3%.

In 2018 Bilbao-Jayo and Almeida built a classifier using Convolutional Neural Networks (CNNs) with Word2Vec embeddings, which they used to classify sentences from political platforms written in Spanish, Finnish, Danish, English, German, French and Italian according to the Manifestos Project codification scheme. The language-specific classifiers were trained and then tested, using sentences extracted from annotated parties' election manifestos. Bilbao-Jayo and Almeida attempted different CNNs algorithms, and the best results were obtained when adding the following features to their models:

1-the political party to which the sentence belonged, and

2-the previous sentence, as additional features in the algorithm.

In this regard, the best results were obtained when classifying documents in the Spanish language, which returned an accuracy of 72.4%.

More recently, in 2020, Chatsiou [10] trained several classifiers using the Manifestos Project's data and coding scheme, which she then used to classify sentences from a corpus of COVID-19 press briefings according to the seven major policy areas of the Manifestos Project's coding scheme. Interestingly, Chatsiou trains the classifiers exclusively with data from political manifestos and uses them to classify documents of a significantly different nature -the COVID-19 press briefings – without any additional training or fine-tuning. Chatsiou trained four different Convolutional Neural Networks (CNNs), using four different embeddings: Word2Vec, GloVe, ELMO and BERT. The best results were obtained using BERT, which returned an f1-score of 0.75 in a test subset of the political manifestos data, and 0.65 for the domain classification of the COVID-19 press briefings.

It appears relevant to note that the authors mentioned above worked with a classification scheme that comprises only seven policy topics and is, therefore, less detailed than the Comparative Agendas Project's scheme I used, which comprises 22 topics. Nonetheless, the performance of my BERT model is above that of Bilbao-Jayo, Almeid and Chatsiou's work and virtually equal to the performance of Zirn et al.'s best model.

DATA OVERVIEW

For this project, I trained, tested and compared the ability of three different supervised algorithms to label documents according to the Comparative Agendas Project's coding scheme. For this, I used data from the Democratic Party Platform dataset and the Republican Party Platform dataset, compiled by University of Notre Dame's Political Science professor Christina Wolbrecht.[2]

In total, these datasets include the labelling of 37,411 quasi-statement from all Political Platforms published by both parties every four years - before presidential elections - between 1948 and 2016. While the dataset includes the 2020 Democratic platform, the 2020 Republican one has not yet been included.

The Comparative Agendas Project defines a quasi-statement as the text between periods, semi-colons, and other punctuation. For example, the following sentence is divided into four quasi-statements and, thus, they are each labelled independently and included in the datasets in separate rows:

"To do this, we need to educate our people and train our workforce [quasi-statement 1]; *attract and retain talented people from all over the world* [quasi-statement 2]; *attract and retain talented people from all over the world* [quasi-statement 3]*; and invest in research and development, innovation hubs, as well as in getting ideas to market"* [quasi-statement 4].

For simplicity and ease of comprehension, and since most of the quasi-statements included in the datasets are complete sentences, I have used the terms sentences and quasi-statements interchangeably across this report.

The 37,411 quasi-statements included in the datasets are labelled twice: according to their Major Topic and Sub Topic. For this project, I only focused on the Major Topics.

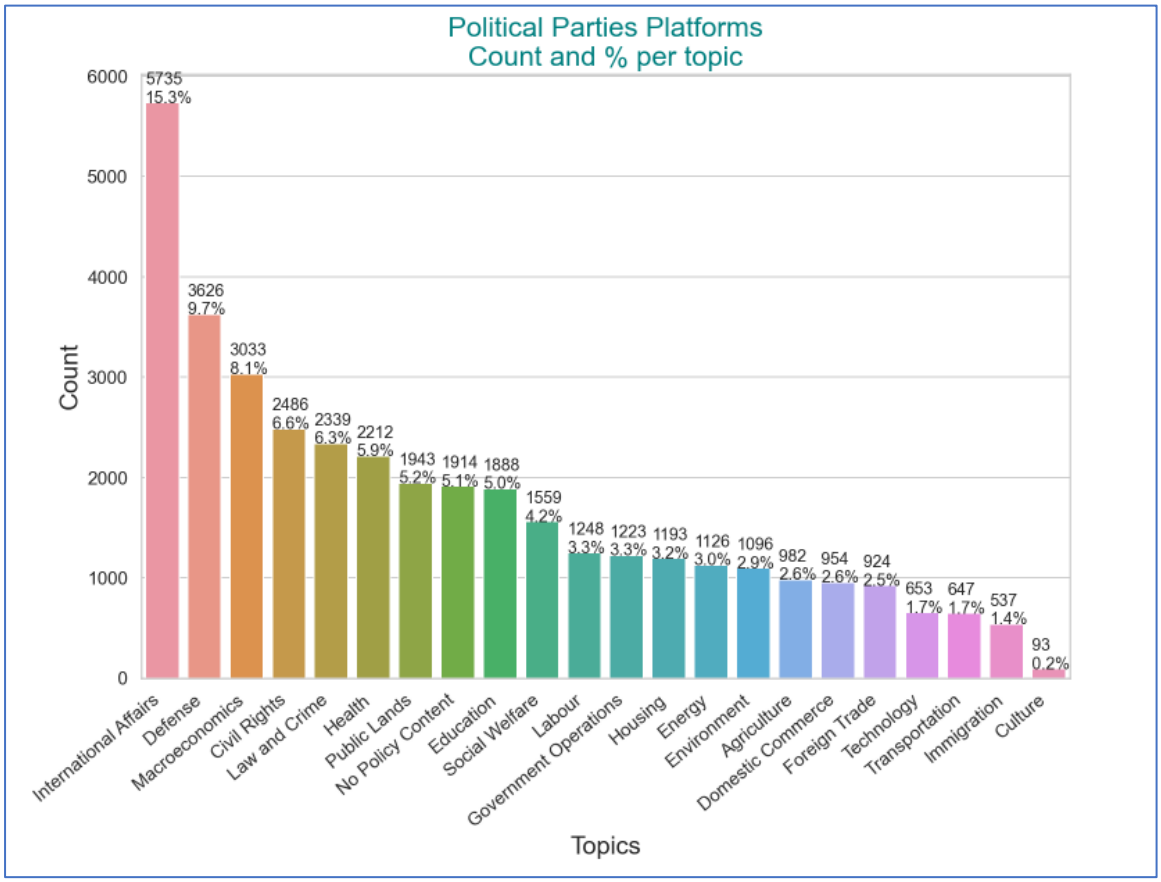
For illustration purposes, I have included below a number of quasi-statements that made part of the latest Democratic Party Platform, published before the November 2020 Presidential elections, along with the respective Major topic.

[2] Christina Wolbrecht. American Political Party Platforms: 1948-2007. University of Notre Dame, 2016

Sentence	CAP Major Topic
To ensure that federal funds are invested as effectively and efficiently as possible, the federal government should be using the best available evidence when making budget and spending decisions.	Macroeconomics
We believe Education is a critical public good, and will increase investments to guarantee all students can access high-quality public schools, no matter where they live, so students of color are well prepared to thrive in college and careers.	Education
The truth is that our immigration system was broken long before President Trump came into office, and his departure alone won't fix it.	Immigration
Democrats will lead with diplomacy as our tool of first resort and mobilise our allies and partners to meet the tests none of us can meet on our own.	International Affairs
Turning the page on two decades of large-scale military deployments and open-ended wars in the Middle East does not mean the United States will abandon a region where we and our partners still have enduring interests.	Defense
We will recommit the United States to the principles of an open internet, including net neutrality, and vigorously oppose efforts to digitally silo off countries and populations from the rest of the world.	Technology
Democrats will work to secure a better future for younger generations	No Policy Content
Democrats believe in bringing the American people together, not stoking division and distrust.	No Policy Content

A review of the terms most frequently used in each category - filtering non-informative stop-words -, returns unsurprising results. For example, terms such as "Taxes", "Economy", "Spending" and "Jobs" are some of the ten most frequently used words in the category Macroeconomics, and "Military", "Weapons", "War" and "Security" are within the ten more used words in Defense. I have included a complete list of the ten words most used per topic in Appendix 1 for reference.

As shown in the bar chart below, the distribution of frequencies between these topics is not balanced, and some are significantly more prominent than others. As shown in the chart, the three most frequent topics are International Affairs (covering 15,3% of the total), Defense (9.7%) and Macroeconomics (8.1%). In turn, less than 2% of the sentences are classified as Technology (1.7%); Transportation (1.7%); or Immigration (1.4%), and less than 1% belong to the topic Culture (0.2%).

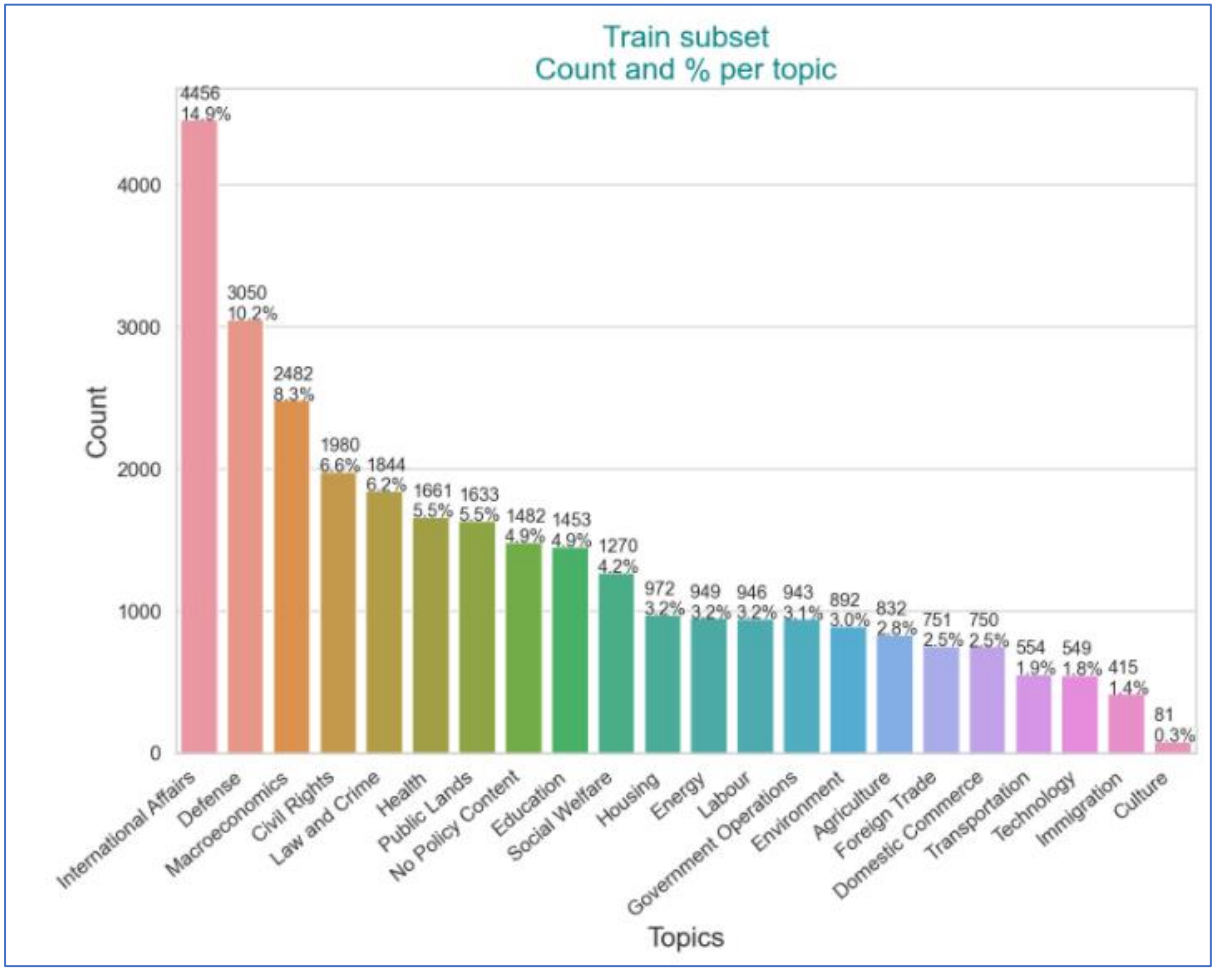


Together, the Republican and Democratic datasets include 37 party platforms, each of which was published before an election, every four years. As mentioned above, the 2020 Republican platform has not been yet labelled and included in the dataset. As a result, merging the two datasets results in an uneven number of platforms, with 18 belonging to the Republican Party and 19 to the Democratic. These platforms were randomly split into three groups:

- a Train subset, comprising six platforms and 29945 sentences;

- a Validation subset, with four platforms and 3713 sentences; and
- a Test subset, with four platforms and 3753 sentences.

Naturally, given that the data was randomly split, the resulting train subset had a similar distribution to that of the complete data, with some topics having a significantly lower number of examples than others, as seen in the chart below:



In order to mitigate potential problems derived from the unbalance, I complemented the Training subset with data from the US State of the Union (SOTU) dataset, which also includes sentences classified according to the coding scheme of the Comparative Agendas Project. The SOTU dataset includes the labelling of 23,044 sentences from all presidential State of the Union speeches delivered between 1946 and 2020. To compensate for those topics presenting a reduced number of examples and provide some level of balance to the training sample, I added randomly selected sentences from the SOTU dataset, with the aim to bring each topic to a total of 1000 examples. However, given that some of these topics presented a reduced number of examples in the SOTU dataset, some of them

did not make it to 1000 sentences. A bar chart showing the distribution of the Train subset after including sentences from the SOTU dataset has been included in Appendix 2.

While some attempts were made to further balance the training data by eliminating a percentage of the sentences belonging to the most frequent topics, this did not improve the accuracy of my models, and therefore datapoints were finally not removed.

After training, fine-tuning and testing three models in the test subset of the US Party Platforms, I also tested the ability of my BERT Model to identify the major topic of policy sentences from a totally independent context: the UK Speeches From The Throne dataset, which includes the coding of 105 speeches delivered by the King or Queen of the United Kingdom between 1911 and 2012. Like the Party Platforms and the SOTU datasets, the UK Speeches From The Throne dataset was coded at the 'quasi-sentence' level with the Comparative Agendas Project's coding scheme.

METHODOLOGY AND MODELS

In order to automate the labelling of sentences according to the Comparative Agendas Project coding scheme, three models were developed during this project:

1) Naive Bayes, with TF-IDF vectorisation;
2) XGBoost, with a pre-trained set of 300-dimensional GloVe embeddings;
3) A fine-tuned BERT model, using Hugging Face's Transformers library for Pytorch.

During the experimentation phase of this project, all three models were trained, first, using only the sentence being classified. Subsequently, I also added the previous sentence to the model. For the three of them, better results were obtained when adding the previous sentence, and, as such, the models described below include the previous sentence.

Naive Bayes with TF-IDF vectors

Multinomial Naive Bayes is a probabilistic learning method that works by estimating the probability of a document being in every possible class in order to find the best class for that document [11]. It makes use of the Bayes' Rule for classification, which estimates the probability of event H, conditioned on observing evidence E:

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)}$$

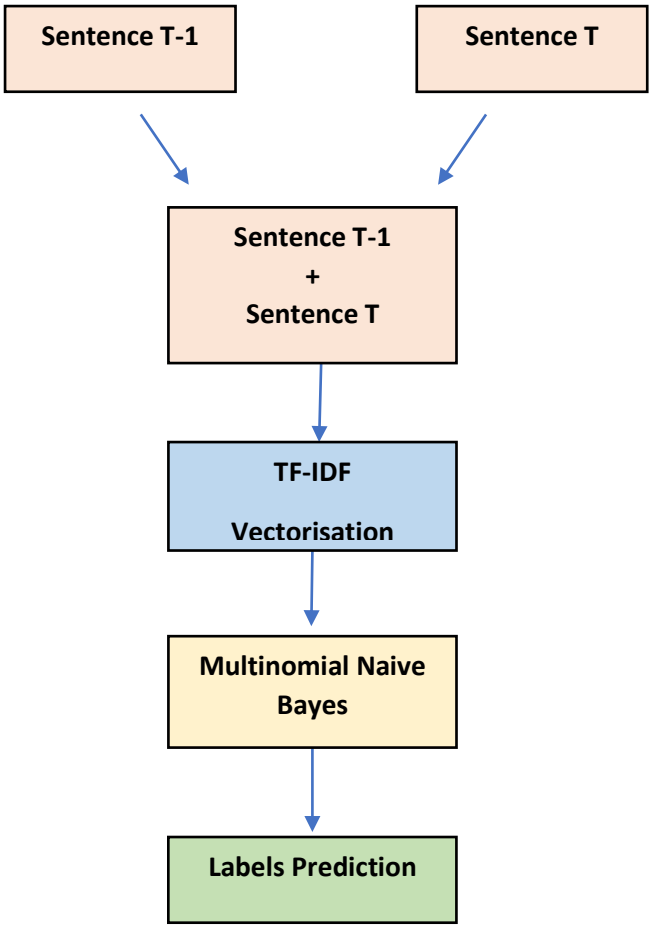
Unlike other Bayesian network models for classification, which also include feature-feature dependencies to calculate probabilities, Naive Bayes classifiers simply assume that all features (in this case, words or N-grams) are independent from each other. Therefore the probability is calculated only taking into consideration the dependency between the words and the classes.

One other important and potentially problematic independence assumption made by this model is the positional independence assumption, which means that the algorithm does not consider the order of the words. This is known as a bag-of-words model. In natural language, of course, stating "*The American people will fight back: we will never surrender*" is very different from saying "*The American people will surrender: we will never fight back*", but bag-of-words models do not perceive this difference.

Following the Bayes' Rule for classification, the probability that a sentence belongs to a particular topic is a function of the probability of every word in the sentence to belong to that class, which is calculated according to the words seen in the training data. Thus, any term in the test subset that was not seen in a given category in the training subset leads to estimating that the probability of such a document belonging to such a class is zero, even if all other words in the document relate very strongly to that class. In order to mitigate this problem and to avoid multiplying by zero, the Laplace smoothing is incorporated. This is simply adding one (or another value, known as "alpha" parameter in Naive Bayes) to the count of each term in each class when calculating probabilities,

I decided to use this algorithm due to its simplicity and parsimony: unlike other more advanced and complex models, Naive Bayes algorithms make their predictions in a very transparent, explainable way. In addition, and also as a result of its simplicity, Naive Bayes is, compared to other algorithms, relatively easy and fast to implement.

Model details



After filtering stop-words, removing symbols, lower-casing and stemming the words using a Porter stemmer, sentences were converted into numerical vectors using the Term Frequency — Inverse Document Frequency (TF-IDF) technique. TF-IDF's main objective is to assign high weights for rare or unique terms (which are expected to be more informative) and low weights for more frequent terms. TF-IDF computes the importance of a term in a document, scaled by the inverse of its frequency in the collection. In the TF-IDF formula below, "tf" means the frequency of the term in the document, "N" is the total number of documents in the collection, and "df" is the number of documents in which the term is present:

$$w_{t,d} = (1 + \log \mathrm{tf}_{t,d}) \cdot \log \frac{N}{\mathrm{df}_t}$$

Given that both TF-IDF and Naive Bayes have several tuneable hyperparameters, I ran a grid search using the training data to test the performance of almost 200 combinations of parameters in the Validations subset, including:

- The size of the n_grams used during the TF-IDF vectorisation;
- The maximum number of n_grams to consider during the vectorisation (i.e., the maximum length of the resulting vectors);
- The threshold to ignore n-grams that are present in the corpus above a specific frequency during the vectorisation, as they are considered to provide limited information gain;
- The value ("alpha") to add during the Laplace smoothing for the Naive Bayes explained before, and
- Whether to consider the probability of each class in the training corpus (i.e., the prior probability distribution) when calculating the probabilities of each sentence.

The best combination of parameters identified was:

- n_grams of size 1 and 2, (i.e., unigrams and bigrams);
- a maximum of 8,000 n_grams;
- including n_grams present in 10% or less of the sentences;
- Alpha value 1 for the Laplace smoothing;
- not to consider the prior probability distribution when calculating the class in the train set.

Considering the simplicity of the Naive Bayes algorithm, as well as its ease and speed of training, this model was included as the baseline model, and I expected that the XGBoost and BERT models, which are more complex and require significantly more training time, would both return a better

performance. However, as detailed in the Models Performance section of this report, I achieved better performance with the Naive Bayes than with the XGBoost model.

XGBoost with GloVe embeddings

XGBoost, which stands for Extreme Gradient Boosting, is a tree-based ensemble method. In order to describe this algorithm, three key concepts should be introduced first, as they constitute its pillars: Ensemble methods, Boosting and Gradient Descent.

First of all, Ensembles refers to the aggregation of several predictors, be it by using majority-vote in the case of classification problems (i.e., predicting the class that got the highest number of votes across the different ensembled methods) or predicting the average of the predictions made by the different ensembled models in the case of regression problems. For example, the Random Forests method is an ensemble of Decision Trees trained with randomly selected sub-samples of the dataset. Ensemble models are usually able to reduce the variance of single estimators.

Boosting is a special kind of Ensemble method. Instead of averaging or voting among several independent predictors, booting methods are built sequentially, each trying to improve the output of its predecessor. However, unlike other boosting models (such as AdaBoost), Gradient Boosting does not work by tweaking the models' weights at every iteration but works by fitting the new predictor to the residual errors made by the previous predictor [12].

This is achieved by using the Gradient Descent method – the third key concept to describe Extreme Gradient Boosting –, which can be defined as an iterative optimisation algorithm that minimises a cost function (e.g., the mean squared error or the log loss) by tweaking parameters iteratively, using steps proportional to the negative gradient of such function [13] [14].

A key difference, then, between Random Forests and Gradient Boosting is how trees are built. While in Random Forests each tree is built independently, and their predictions are combined at the end of the process, in Gradient Boosting, trees are created in sequence, making use of the results of the previous one.

Once these concepts are introduced, XGBoost can be defined as an optimised implementation of Gradient Boosting, which supports classification problems and regression and ranking tasks.

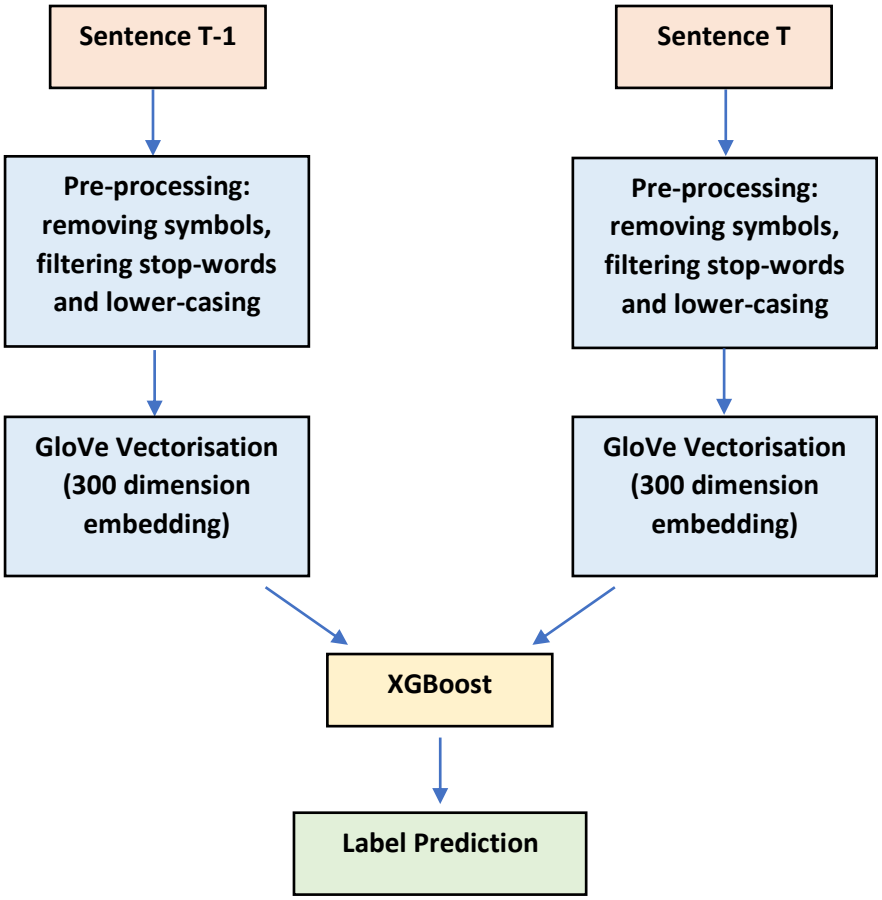
XGBoost has achieved excellent results in several NLP classification tasks (see, for example, [15]), being more efficient than other Gradient Boosting algorithms. In addition, it works very well in combination with pre-trained word embeddings such as GloVe [16], which I used in my model.

GloVe – which stands for Global Vectors- is an unsupervised learning algorithm to create vector representations for words from a given corpus [17]. Published by Stanford University researchers in 2014, the GloVe algorithm obtains word vectors by computing the co-occurrence probabilities between words in the corpus, according to a defined context window (i.e., the maximum distance between two words for them to be considered co-occurring). Subsequently, the word-to-word co-occurrence probability matrix's dimensionality is reduced using a matrix factorisation technique, similar to Latent Semantic Analysis. When released, the GloVe model returned state-of-the-art performance in several word similarity and word analogies tasks.

Despite the success it has demonstrated for several NLP tasks, using GloVe embeddings to vectorise sentences has two significant weaknesses that may impact its performance. Given that GloVe models include vectors for words, how GloVe vectors are computed for sentences is by calculating the average for the vectors of every word that is present in the sentence. Consequently, a very relevant word in the sentence may lose steam by being averaged with the vectors of all other words present in the sentence. This is especially relevant in the case of long sentences. In addition, GloVe vectorisation does not take into account the order of the words, and, as a result, sentences that have opposite meanings may be vectorised in the exact same way.

For this project, I did not create vector representations with my corpus but used an English model available in the NLP Spacy library, named “en_core_web_md”, which comprises 300 dimensions and 20.000 unique vectors.

Model details



As mentioned before, sentences were transformed into vectors using SpaCy's pre-trained GloVe model. Then, the 300 dimensions of the sentence being classified and the previous one were concatenated, and the 600 were used in the XGBoost classifier.

Using cross-validation grid search on the Train subset, the combination of 5 parameters were attempted, resulting in a total of 162 candidate estimators. XGBoost's parameters tuned during this process included:

- The number of trees to be created;
- The maximum depth of each tree;
- The number of features (from the total 600) to be used in each tree; and
- Two regularisation parameters: Gamma, which sets the minimum reduction in the loss required to split a node, and min_child_weight, which determines the minimum number of instances needed to be left in each node after a split for it to take place.

Despite the significant number of attempted combinations, the best results were obtained with parameters remarkably close to XGBoost's default parameters. In fact, the grid search best results were obtained when the regularisation parameters Gamma and min_child_weight were set in 0 and 1, respectively, the default values, which imply no regularisation.

BERT

Bidirectional Encoder Representations from Transformers, or simply BERT, is a pre-trained language representation model released by Google in 2018 [18]. Unlike previous state-of-the-art pre-trained language models, BERT embeddings are not trained by sequentially feeding the algorithm with left-to-right or right-to-left words. In turn, BERT relies upon Transformers, a bidirectional network architecture published by Google researchers in 2017. Unlike previous sequence-to-sequence models, the Transformers architecture does not incorporate recurrent or convolutional networks and is entirely based on attention mechanisms [19].

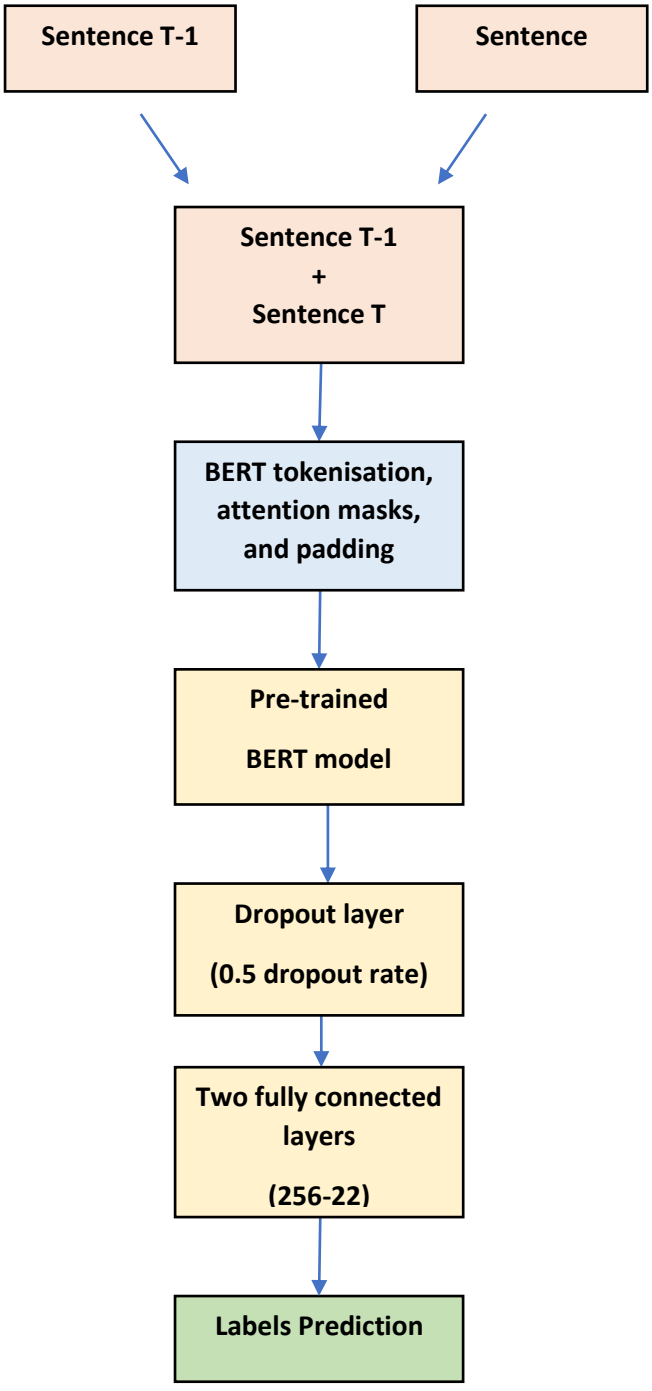
BERT's encoding uses a "masked language model" which randomly masks tokens from the input and attempts to predict the token that has been masked based only on its context. In addition, BERT is also trained through binary classification tasks, consisting of the following: given a pair of sentences, the model is trained to predict whether one is followed by the other. BERT was trained with two corpuses: the Toronto Book Corpus dataset, which comprises more than 11,000 books, and English Wikipedia.

Two different versions of BERT have been released: a larger one, comprising 24 encoder layers and 340 million parameters; and a base one with 12 encoder layers and 110 million parameters.

BERT embeddings can be used to solve a significant number of NLP tasks by simply adding one output layer (or more) and fine-tuning it. Subsequently, for a classification task, we can predict the class with the highest value in the output, or we can obtain the probabilities of each class by simply adding a SoftMax function.

When released in 2018, BERT showed state-of-the-art results on eleven natural language processing tasks, and it still ranks as one of the best models for many problems, such as question answering, part-of-speech tagging, information retrieval and sentence classification [20]. Given the excellent results, BERT has become one of the "de facto" go-to ways to establish benchmarks in NLP [21]. Considering the state-of-the-art results it has obtained in several NLP tasks, it is not surprising that this model returned the best results from the three I advanced during this project.

Models details



For this project, I used a base (12 layers) pre-trained BERT model available through Hugging Face's Transformers library in its uncased version, i.e., not distinguishing capital from lower-case letters. Using Pytorch, I created a unified model composed of the base BERT mentioned above model, plus:

- a dropout layer (dropout rate=0.5), to provide regularisation and prevent overfitting on the training data;

- a fully connected layer with 256 neurons and ReLU activation function; and
- a fully connected layer of 22 neurons, leading to the output.

The model was trained end-to-end, including fine-tuning the BERT layers, during ten epochs. I used PyTorch's AdamW optimiser, a batch size of 32 and a learning rate of 3e-5, which is in line with the recommendations provided by the authors of the BERT original paper for fine-tuning [18].

The BERT model was trained using both the sentence being classified and the prior one, which were concatenated before feeding them into the model. In addition, BERT requires a series of pre-processing steps for the sentences, which include:

- each word (or subworld, in some cases) is converted into its correspondent BERT id;
- a special token "[CLS]" is added in front of every sentence. This is required in BERT for classification tasks. Also, the token "[SEP]" is added at the end of each sentence;
- Given that BERT requires sentences of consistent lengths, pad tokens ("0") are added at the end of each sentence shorter than a determined maximum length (I selected 150 tokens, in this case) in order to unify the length of sentences; and
- Following the padding, an attention mask is created for each sentence, differentiating real tokens from the pad tokens added in the previous step.

After training and testing the performance of this model in the US Political Parties Platforms data, I used the same model (without any additional fine-tuning) to classify sentences from the UK Speeches from the Throne dataset. As expected, the performance of this model when used to predict data from a different context decreased.

Subsequently, I used a small percentage of the UK Speeches from the Throne dataset to fine-tune the BERT model I had previously fine-tuned for the US Political Parties Platforms data. While the model was fine-tuned using only 15% of the UK data, the improvement in its performance when then tested in the UK Test subset was notable. More details about the performance of the three models are included in the following section of this report.

MODELS PERFORMANCE

As expected, the fine-tuned BERT model was the one that presented the best results, with an overall accuracy of 79% in the test set of the US Political Party Platforms data. In this regard, the performance achieved by this model appears to be equivalent to the one developed by Zirn et al. in 2016 and better than the performance achieved by Chatsiou in 2020.

As detailed in the Related Work section of this report, both of the projects mentioned above used a coding scheme that classifies policy sentences into seven categories. In turn, the Comparatives Agenda Project's coding scheme that I used includes 22 topics. As a result, considering the use of a different coding scheme, it is not possible to drive a direct and definitive comparison between my results and theirs.

While we present detailed classification reports for each of the three models below, as well as complete confusion matrices in Appendix 1, the results obtained in the US Political Party Platforms Test subset are summarised in the following table:

PERFORMANCE COMPARISON - POLITICAL PARTY PLATFORMS DATA							
		PRECISION		RECALL		F1	
	OVERALL ACCURACY	macro average	weighted average	macro average	weighted average	macro average	weighted average
Naive Bayes	70%	70%	70%	68%	70%	0.68	0.70
XGBoost	68%	69%	69%	66%	68%	0.67	0.68
BERT	79%	76%	80%	81%	79%	0.78	0.79

After providing further details about the performance of these models in the US Political Party Platforms data, I also present details about the performance of the BERT model when used to classify data from the UK Speeches from the Throne dataset, both before and after fine-tuning it.

Naive Bayes

The classification report below shows that this model returned an overall accuracy of 70%, with a macro average f1-score of 0.68.

Classification Report - Political Parties Platforms

	precision	recall	f1-score	support
No Policy Content	0.66	0.65	0.66	311
Macroeconomics	0.61	0.76	0.68	264
Civil Rights	0.58	0.66	0.62	152
Health	0.76	0.86	0.81	274
Agriculture	0.67	0.65	0.66	40
Labour	0.69	0.57	0.62	134
Education	0.78	0.88	0.83	267
Environment	0.61	0.69	0.65	85
Energy	0.82	0.81	0.82	58
Immigration	0.64	0.55	0.59	38
Transportation	0.71	0.67	0.69	30
Law and Crime	0.74	0.59	0.66	331
Social Welfare	0.70	0.55	0.62	168
Housing	0.72	0.60	0.65	88
Domestic Commerce	0.70	0.34	0.46	83
Defense	0.62	0.64	0.63	337
Technology	0.60	0.64	0.62	61
Foreign Trade	0.62	0.80	0.70	106
International Affairs	0.78	0.76	0.77	748
Public Lands	0.54	0.67	0.60	102
Government Operations	0.83	0.67	0.74	73
Culture	1.00	1.00	1.00	3
accuracy			0.70	3753
macro avg	0.70	0.68	0.68	3753
weighted avg	0.70	0.70	0.70	3753

From the 22 categories, Domestic Commerce was by far the one presenting the worst f1-score, at just 0.46. As shown in the Confusion Matrix 1 included in Appendix 3, more than 20% of the sentences whose real topic was Domestic Commerce were classified by this model as Macroeconomics, and 13% as Law and Crime. While, clearly, an f-1 score of 0.46 is disappointing, an analysis of those sentences incorrectly classified sheds some light, and suggest that at least some of the errors made by the algorithm could well have been made by human coders, given that the categories are not mutually exclusive, and a large proportion of sentences classified in the dataset as Domestic Commerce may well be considered as belonging, for example, to Macroeconomics.

Consider, for example, the following sentences, whose actual label was Domestic Commerce, but the Naive Bayes model classified it as Macroeconomics:

- *"It has so discouraged private enterprise that the annual increase in the number of businesses has plummeted from the Republican level of 70,000 a year to 47,000 a year."*
- *"Much of the tax relief came from reductions in individual income tax rates"*
- *"CEOs, who pay themselves 100 times what they pay the average worker, shouldn't get big raises unrelated to performance."*

A similar picture is seen when inspecting those Domestic Commerce sentences classified as Law and Crime. In fact, it could be argued that a large number of sentences classified in the original Dataset as Domestic Commerce should have been classified as Law and Crime:

- *"investigators have been given new tools to root out corporate fraud"*
- *"And like other forms of stealing, identity theft leaves the victim poorer and feeling terribly violated."*
- *"We praise President Bush and Republicans in Congress for passing the Fair and Accurate Credit Transactions Act, which established a national system of fraud detection so that identity thieves can be stopped before they run up tens of thousands of dollars in illegal purchases."*
- *"We further praise President Bush and Republicans in Congress for passing the Identity Theft Penalty Enhancement Act, which provides a real deterrent by toughening the prison sentences for those who use identity theft to commit other crimes, including terrorism."*
- *"It reflects our government's resolve to answer serious offenses with serious penalties."*
- *"The U.S. Postal Inspection Service, the FBI, and Secret Service are working with local and state officials to crack down on the criminal networks that are responsible for much of the identity theft that occurs in America."*

Given its simplicity and ease of training, Naive Bayes was included in this project as the baseline model, and I expected both the BERT and the XGBoost models to be more accurate for the classification task. However, despite its simplicity, the Naive Bayes model returned a better performance than the much more complex XGBoost, presented below. Even if we consider their results to be reasonably close, Naive Bayes offers some additional benefits over XGBoost. Besides the radically shorter time needed to train and fine-tune its parameters, Naive Bayes is a white-box algorithm that performs its predictions in an intuitive and easily interpretable way.

XGBoost

While the accuracy level achieved by the XGBoost model was not too far from the one achieved by the Naive Bayes – 68% vis-a-vis 70% –, my expectation was for this to present a far better performance, given the outstanding results it has obtained in other NLP classification tasks [15], and in particular when combined with pre-trained word embeddings such as GloVe [16]. The detailed results are presented in the following table:

Classification Report - Political Parties Platforms

	precision	recall	f1-score	support
No Policy Content	0.63	0.51	0.56	311
Macroeconomics	0.67	0.79	0.72	264
Civil Rights	0.55	0.66	0.60	152
Health	0.84	0.84	0.84	274
Agriculture	0.73	0.75	0.74	40
Labour	0.65	0.55	0.60	134
Education	0.86	0.84	0.85	267
Environment	0.61	0.66	0.63	85
Energy	0.79	0.72	0.76	58
Immigration	0.64	0.61	0.62	38
Transportation	0.54	0.73	0.62	30
Law and Crime	0.72	0.56	0.63	331
Social Welfare	0.63	0.55	0.59	168
Housing	0.76	0.59	0.67	88
Domestic Commerce	0.58	0.37	0.46	83
Defense	0.60	0.67	0.63	337
Technology	0.69	0.69	0.69	61
Foreign Trade	0.68	0.73	0.70	106
International Affairs	0.72	0.77	0.74	748
Public Lands	0.43	0.61	0.50	102
Government Operations	0.77	0.67	0.72	73
Culture	1.00	0.67	0.80	3
accuracy			0.68	3753
macro avg	0.69	0.66	0.67	3753
weighted avg	0.69	0.68	0.68	3753

As mentioned before, a significant amount of time was devoted to finding the best combination of parameters using a grid search cross-validation. Despite this, the regularisation parameters attempted did not improve the performance of this model. In fact, following the grid search, it was concluded that the best combination of parameters was that in which the regularization parameters Gamma and min_child_weight were not used.

Naturally, it also possible that the modest performance of the model is not related to XGBoost itself but with the embeddings used. However, I did attempt to train and fine-tune XGBoost models with other pre-trained word embeddings and TF-IDF, but these did not yield better performances.

Like in the Naive Bayes model, the lowest f1-score was achieved with Domestic Commerce, which returned an f1-score of 0.46. Moreover, the XGBoost model also returned an f1-score of 0.56 for the "No Policy Content" category, which had obtained an f1-score of 0.66 with Naive Bayes. Considering this drop, I attempted to use XGBoost to conduct - as a first step - a hierarchical classification, to first distinguish sentences with policy content from sentences without policy content. However, this did not return better results.

BERT

Finally, and as previously stated, BERT was the model that returned the best performance, achieving an overall accuracy of 79% and a macro average f1-score of 0.78.

Classification Report - Political Parties Platforms

	precision	recall	f1-score	support
No Policy Content	0.80	0.60	0.69	311
Macroeconomics	0.78	0.82	0.80	264
Civil Rights	0.57	0.76	0.65	152
Health	0.87	0.89	0.88	274
Agriculture	0.81	0.85	0.83	40
Labour	0.75	0.72	0.73	134
Education	0.92	0.91	0.92	267
Environment	0.71	0.75	0.73	85
Energy	0.89	0.86	0.88	58
Immigration	0.73	0.87	0.80	38
Transportation	0.62	0.93	0.75	30
Law and Crime	0.76	0.68	0.72	331
Social Welfare	0.80	0.73	0.76	168
Housing	0.86	0.89	0.87	88
Domestic Commerce	0.62	0.54	0.58	83
Defense	0.75	0.81	0.78	337
Technology	0.76	0.82	0.79	61
Foreign Trade	0.77	0.80	0.79	106
International Affairs	0.85	0.83	0.84	748
Public Lands	0.68	0.85	0.76	102
Government Operations	0.88	0.93	0.91	73
Culture	0.60	1.00	0.75	3
accuracy			0.79	3753
macro avg	0.76	0.81	0.78	3753
weighted avg	0.80	0.79	0.79	3753

Like in the previous two models, the category presenting the worst results was Domestic Commerce, which returned an f1-score of 0.58. As discussed above, some of the sentences under this category may have been classified incorrectly, or at least they could reasonably be classified in a different class.

Besides comparing the "True" label with the one predicted, it is also insightful to observe the probabilities returned by the BERT model for some of the sentences it misclassified.

The following sentence, for example, was predicted by the BERT model to belong to International Affairs, with a confidence level of almost 100%:

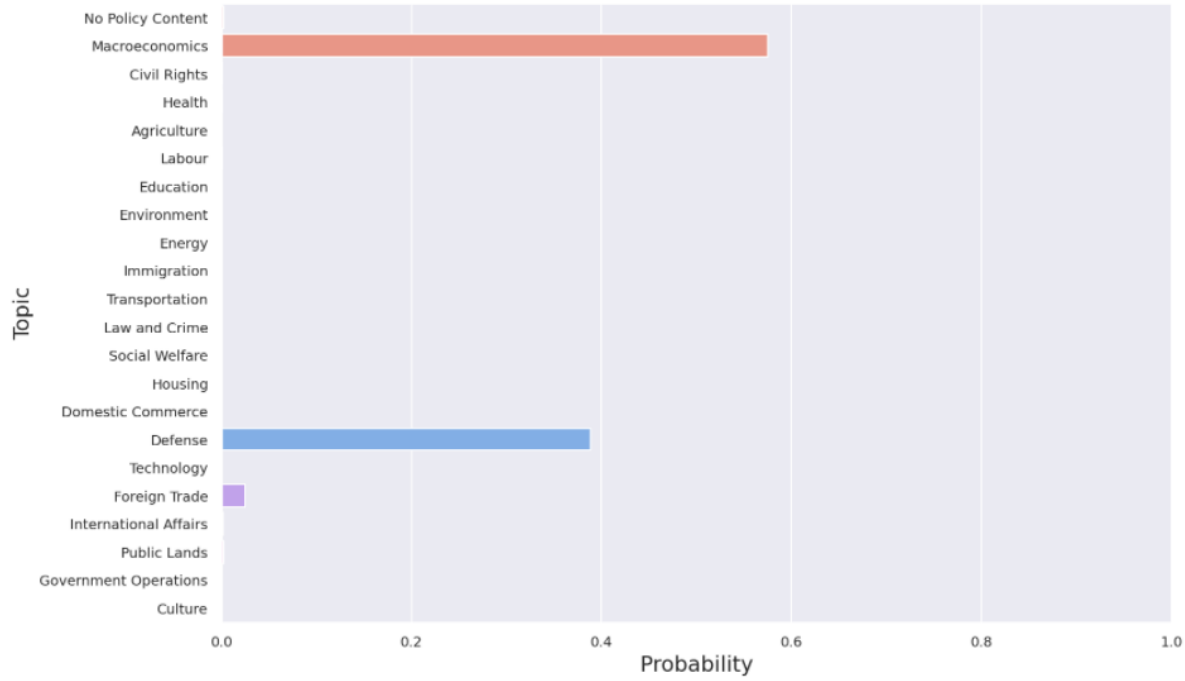
> *"We have full confidence in the plan for Iraqi self-government that is currently being implemented by Iraq's interim government."*

The sentence, however, was in fact labelled as Defense in the dataset. While the broader context in which this sentence was included was referring to Defense matters, there appears to be little in the sentence itself or in the previous sentence (*"It is a proud moral achievement for members of our coalition.")* that would have signalled to the algorithm that it was related to Defense. Considering that 14% of sentences whose valid label was Defense were classified by the BERT model as International Affairs, it is possible that, sometimes, including just one previous sentence may be less than enough for the model to really appreciate finer details about the context.

It is also interesting to note how some sentences are misclassified when vocabulary characteristic from one category is used -sometimes metaphorically - in a sentence of another topic. Consider, for example, the following sentence, which was included in the context of weapons policies:

“We will end the false economies which place price ahead of the performance upon which American lives may depend.”

While the previous sentence *(“We will end "second-best" weapons policies.”)* did include the word “Weapons”, which is directly attached to defence matters, it is notable that the terms "Economies" and "Price" included in the sentence are more usually associated with matters of the economy than of Defense. As a result, the model returned a probability of almost 60% that this sentence was related to Macroeconomics, and almost 40% of it pertaining to Defense.



Other examples of sentences misclassified by the BERT model, and the probabilities returned, are included in Appendix 3 of this report.

Performance in the UK Speeches from the Throne Dataset

The performance of the BERT model was tested with data from the UK Speeches from the Throne dataset, in two ways. First, I used the model to predict the categories of the complete dataset, without conducting any degree of additional fine-tuning with it. As expected, the performance diminished considerably, returning an accuracy of 58%.

Classification Report – Speeches from the Throne

	precision	recall	f1-score	support
No Policy Content	0.66	0.29	0.40	482
Macroeconomics	0.66	0.58	0.62	586
Civil Rights	0.39	0.45	0.41	112
Health	0.71	0.72	0.72	138
Agriculture	0.68	0.68	0.68	198
Labour	0.50	0.46	0.48	217
Education	0.81	0.83	0.82	213
Environment	0.55	0.44	0.49	81
Energy	0.69	0.77	0.73	97
Immigration	0.38	0.88	0.53	24
Transportation	0.70	0.80	0.75	169
Law and Crime	0.77	0.67	0.71	351
Social Welfare	0.52	0.46	0.49	192
Housing	0.63	0.59	0.61	217
Domestic Commerce	0.60	0.52	0.56	160
Defense	0.63	0.64	0.63	584
Technology	0.58	0.53	0.55	34
Foreign Trade	0.37	0.61	0.46	223
International Affairs	0.58	0.81	0.68	1365
Public Lands	0.39	0.57	0.46	501
Government Operations	0.37	0.07	0.12	667
Culture	0.44	0.31	0.36	13
accuracy			0.58	6624
macro avg	0.57	0.58	0.56	6624
weighted avg	0.57	0.58	0.55	6624

Subsequently, I selected 15% of the UK Speeches from the Throne Dataset to fine-tune the model and 5% for validation. After fine-tuning it with ten epochs, I tested the model in the remaining 80% of the sentences. In this way, the performance of the model increased notably, reaching an accuracy of 70%.

Classification Report – Speeches from the Throne

	precision	recall	f1-score	support
No Policy Content	0.96	0.96	0.96	357
Macroeconomics	0.69	0.68	0.68	409
Civil Rights	0.60	0.42	0.50	71
Health	0.73	0.79	0.75	98
Agriculture	0.68	0.70	0.69	158
Labour	0.47	0.52	0.50	145
Education	0.83	0.86	0.85	146
Environment	0.56	0.58	0.57	53
Energy	0.72	0.69	0.70	67
Immigration	0.71	0.71	0.71	14
Transportation	0.64	0.85	0.73	114
Law and Crime	0.84	0.72	0.78	245
Social Welfare	0.56	0.53	0.54	133
Housing	0.67	0.59	0.63	158
Domestic Commerce	0.52	0.54	0.53	106
Defense	0.70	0.74	0.72	429
Technology	0.57	0.77	0.65	22
Foreign Trade	0.50	0.63	0.56	167
International Affairs	0.77	0.76	0.76	1002
Public Lands	0.74	0.68	0.71	356
Government Operations	0.55	0.53	0.54	477
Culture	0.17	0.14	0.15	7
accuracy			0.70	4734
macro avg	0.64	0.65	0.65	4734
weighted avg	0.70	0.70	0.70	4734

Interestingly, after fine-tuning the BERT model for this specific dataset -for which, as mentioned, I used just 15% of the sentences – it returned almost a perfect precision and recall for the identification of sentences classified as “No Policy Content”, with an f1-score of 0.96. In other words, that limited

degree of fine-tuning was enough for the model to learn to distinguish almost entirely those sentences with policy content from those sentences without.

Notably, and although it was fine-tuned with a much larger corpus, the BERT model in the US Parties Platforms returned a precision of 80% and a recall of 60% for the No Policy Content category. This difference suggests that, possibly, the UK Speeches from the Throne dataset is more consistent in relation to what does and what does not constitute policy content. This is not a trivial question in the context of speeches delivered at parliament and in political parties platforms published before elections.

CONCLUSION AND FUTURE WORK

My project aimed to explore and test some avenues in which Machine Learning and Natural Language Processing could be used to automatise the important but costly and time-consuming process of classifying policy-related texts according to the topics they address. In particular, the primary datasets used during this project to train and test my models were the political platforms published between 1948 and 2020 by the US Republican and Democratic parties. Consequently, the main points of reference were the research projects published by other researchers who have worked with these documents, such as Zirn et al., Bilbao-Jayo and Almeida and Chatsiou, discussed in the Related Work section of this report.

The performance achieved with the BERT model appears to be equivalent or superior to the performance achieved by previous attempts to automate the labelling of political platforms. Considering that the most prominent work on the field was developed using a different coding scheme, a future avenue for this project would be to retrain my models according to the Manifestos Project's schema. This would enable a more direct comparison between the results of my work and that of previous researchers.

Beyond the specific results achieved in this project, the work I developed made evident the potency of the pre-trained BERT model and showed its evident superiority over the other models used in this work for tackling complex classifications tasks. Furthermore, the improvements in performance achieved in the classification of the UK Speeches from the Throne data when fine-tuning the BERT model with a small train set also showed the degree of improvements that is possible to achieve in the performance of these models through fine-tuning, even with a small amount of data.

The work carried out for this project also highlights that the complexity of a model does not necessarily entail that it will achieve a better performance than more simple ones. In that sense, it is notable that the Naive Bayes model, fed with plain TF-IDF vectors, returned better results than XGBoost, an ensemble method that incorporates boosting and gradient descent, and whose training and fine-tuning take considerable more time than Naive Bayes.

By analysing sentences misclassified by my models, this project also highlighted the problematic nature of coding schemes that assign just one label to sentences that could well be classified in more than one category. As discussed before and illustrated with further examples in Appendix 4, it is possible that a sentence about Education, for example, also relates strongly to Technology. Should this be a prevalent phenomenon, it may prevent any system – regardless if it is an automated or a

manual one– to achieve very high levels of accuracy, given that it entails comparing predictions to "True" labels that are subject to some degree of arbitrarity.

REFERENCES

[1] John, P., Bertelli, A., Jennings, W., & Bevan, S. (2013). The Policy Agenda and British Politics. In Policy Agendas in British Politics (pp. 1-22). Palgrave Macmillan, London.

[2]Baumgartner, F. R., Green-Pedersen, C., & Jones, B. D. (2006). Comparative studies of policy agendas. Journal of European Public Policy, 13(7), 959-974.

[3] Brasil, F. G., & Capella, A. C. N. (2019). The Brazilian policy agendas: an analysis of capacity and diversity over priorities in public policies for the period 2003 to 2014. Cadernos de Gestao Publica e Cidadania, 24(78).

[4] Tresch, A., Sciarini, P., & Varone, F. (2013). The relationship between media and political agendas: Variations across decision-making phases. West European Politics, 36(5), 897-918.

[5] Borghetto, E., & Belchior, A. M. (2020). Party manifestos, opposition and media as determinants of the cabinet agenda. Political Studies, 68(1), 37-53.

[6] Russell, A. (2018). US senators on twitter: Asymmetric party rhetoric in 140 characters. American Politics Research, 46(4), 695-723.

[7] Collingwood, L., & Wilkerson, J. (2012). Tradeoffs in accuracy and efficiency in supervised learning methods. Journal of Information Technology & Politics, 9(3), 298-318.

[8] Zirn, C., Glavaš, G., Nanni, F., Eichorts, J., & Stuckenschmidt, H. (2016). Classifying topics and detecting topic shifts in political manifestos (pp. 88-93). University of Zagreb.

[9] Bilbao-Jayo, A., & Almeida, A. (2018). Automatic political discourse analysis with multi-scale convolutional neural networks and contextual data. International Journal of Distributed Sensor Networks, 14(11), 1550147718811827.

[10] Chatsiou, K. (2020). Text Classification of Manifestos and COVID-19 Press Briefings using BERT and Convolutional Neural Networks.

[11] Schütze, H., Manning, C. D., & Raghavan, P. (2008). Introduction to information retrieval (Vol. 39, pp. 234-265). Cambridge: Cambridge University Press.

[12] Hardisty, E., Boyd-Graber, J., & Resnik, P. (2010, October). Modeling perspective using adaptor grammars. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (pp. 284-292).

[13] Géron, A. (2019). Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. O'Reilly Media.

[14] Athanasiou, V., & Maragoudakis, M. (2017). A novel, gradient boosting framework for sentiment analysis in languages where NLP resources are not plentiful: a case study for modern Greek. Algorithms, 10(1), 34.

[15] Talun, A., Drozda, P., Bukowski, L., & Scherer, R. (2020, October). FastText and XGBoost Content-Based Classification for Employment Web Scraping. In International Conference on Artificial Intelligence and Soft Computing (pp. 435-444). Springer, Cham.

[16] Ollagnier, A., & Williams, H. (2019). Classification and event identification using word embedding. neural networks, 6, 7.

[17] Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).

[18] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[19] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008).

[20] Chang, W. C., Yu, H. F., Zhong, K., Yang, Y., & Dhillon, I. S. (2020, August). Taming pretrained transformers for extreme multi-label text classification. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (pp. 3163-3171).

[21] Wu, Z., & Ong, D. C. (2021). On Explaining Your Explanations of BERT: An Empirical Study with Sequence Classification. arXiv preprint arXiv:2101.00196.

APPENDICES

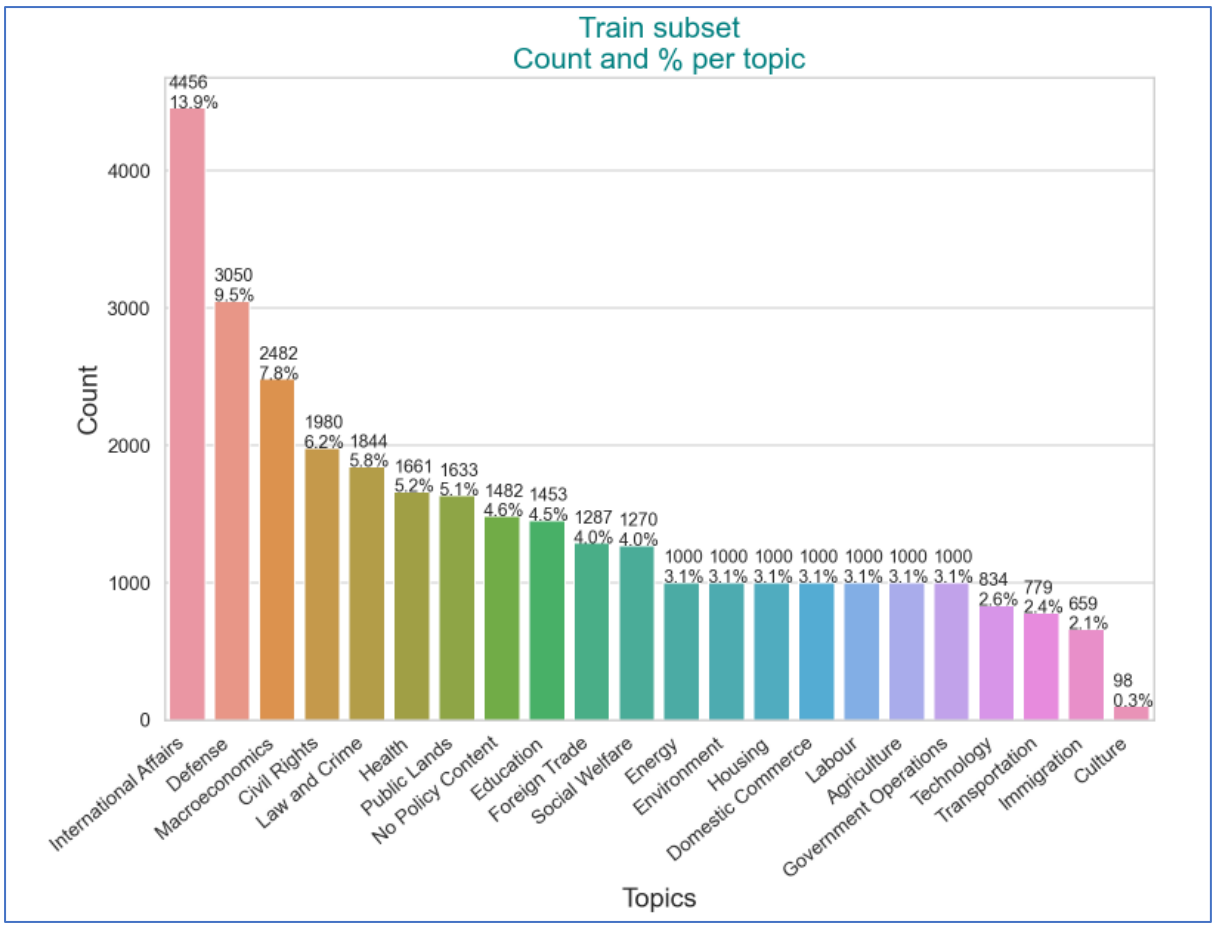
Appendix 1. Most common terms per topic

MAJOR TOPIC	MOST COMMON TERMS
Macroeconomics	['tax', 'economic', 'economy', 'american', 'growth', 'government', 'federal', 'taxes', 'spending', 'jobs']
Civil Rights	['rights', 'women', 'equal', 'discrimination', 'government', 'support', 'right', 'federal', 'americans', 'must']
Health	['health', 'care', 'medical', 'americans', 'insurance', 'research', 'support', 'medicare', 'must', 'access']
Agriculture	['farm', 'farmers', 'agriculture', 'agricultural', 'food', 'american', 'programs', 'family', 'income', 'products']
Labour	['workers', 'labor', 'jobs', 'work', 'right', 'act', 'job', 'training', 'must', 'wage']
Education	['education', 'schools', 'students', 'school', 'children', 'federal', 'support', 'teachers', 'public', 'programs']
Environment	['environmental', 'climate', 'water', 'must', 'environment', 'clean', 'air', 'pollution', 'change', 'new']
Energy	['energy', 'oil', 'power', 'development', 'must', 'gas', 'new', 'coal', 'clean', 'america']
Immigration	['immigration', 'immigrants', 'must', 'country', 'states', 'nation', 'law', 'american', 'united', 'system']
Transportation	['transport', 'system', 'nation', 'highway', 'need', 'infrastructur', 'invest', 'feder', 'improv', 'must']
Law and Crime	['crime', 'law', 'federal', 'support', 'enforcement', 'children', 'must', 'drug', 'justice', 'family']
Social Welfare	['social', 'security', 'welfare', 'work', 'care', 'programs', 'children', 'americans', 'poverty', 'government']
Housing	['housing', 'rural', 'urban', 'cities', 'programs', 'new', 'development', 'federal', 'families', 'areas']
Domestic Commerce	['small', 'business', 'businesses', 'consumer', 'financial', 'government', 'new', 'federal', 'must', 'administration']
Defense	['military', 'nuclear', 'defense', 'must', 'weapons', 'security', 'forces', 'war', 'administration', 'veterans']
Technology	['space', 'technology', 'research', 'science', 'internet', 'new', 'support', 'development', 'national', 'world']

Foreign Trade	['trade', 'american', 'world', 'international', 'economic', 'agreements', 'workers', 'free', 'foreign', 'economy']
International Affairs	['world', 'united', 'states', 'nations', 'peace', 'support', 'people', 'must', 'international', 'economic']
Public Lands	['government', 'federal', 'public', 'people', 'congress', 'must', 'states', 'state', 'programs', 'service']
Government Operations	['federal', 'indian', 'water', 'support', 'american', 'resources', 'states', 'lands', 'government', 'development']
Culture	['arts', 'humanities', 'support', 'national', 'cultural', 'endowment', 'federal', 'programs', 'public', 'nation']
No Policy Content	['people', 'america', 'party', 'american', 'government', 'new', 'nation', 'americans', 'democratic', 'world']

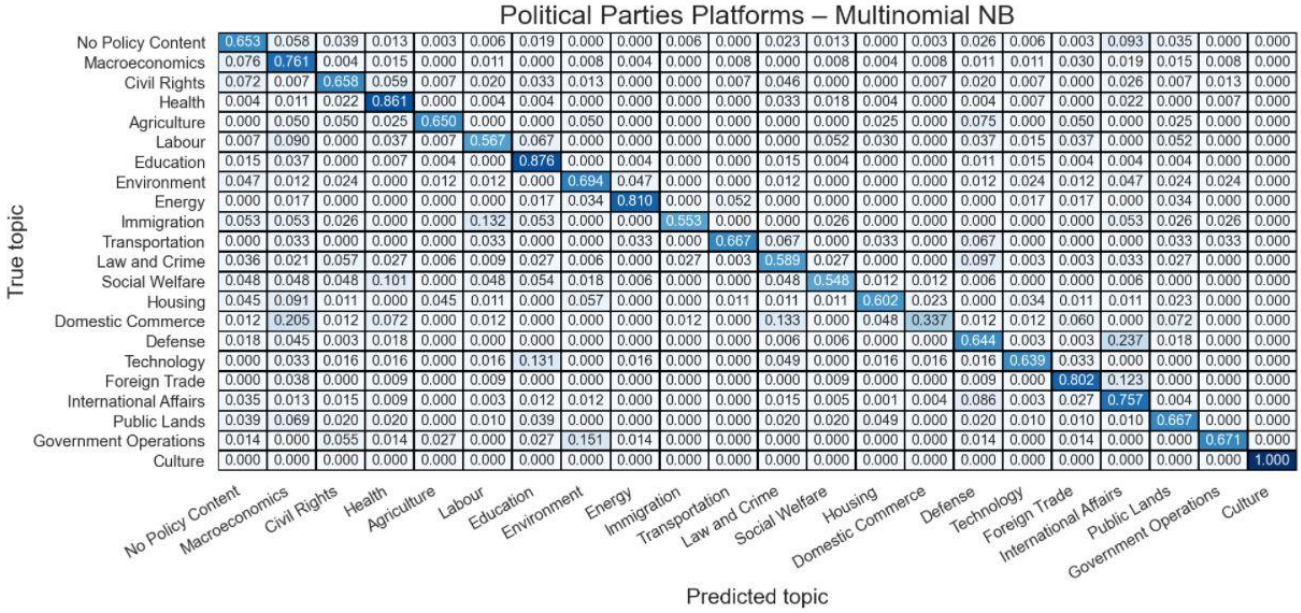
Appendix 2. Distribution of the Train subset

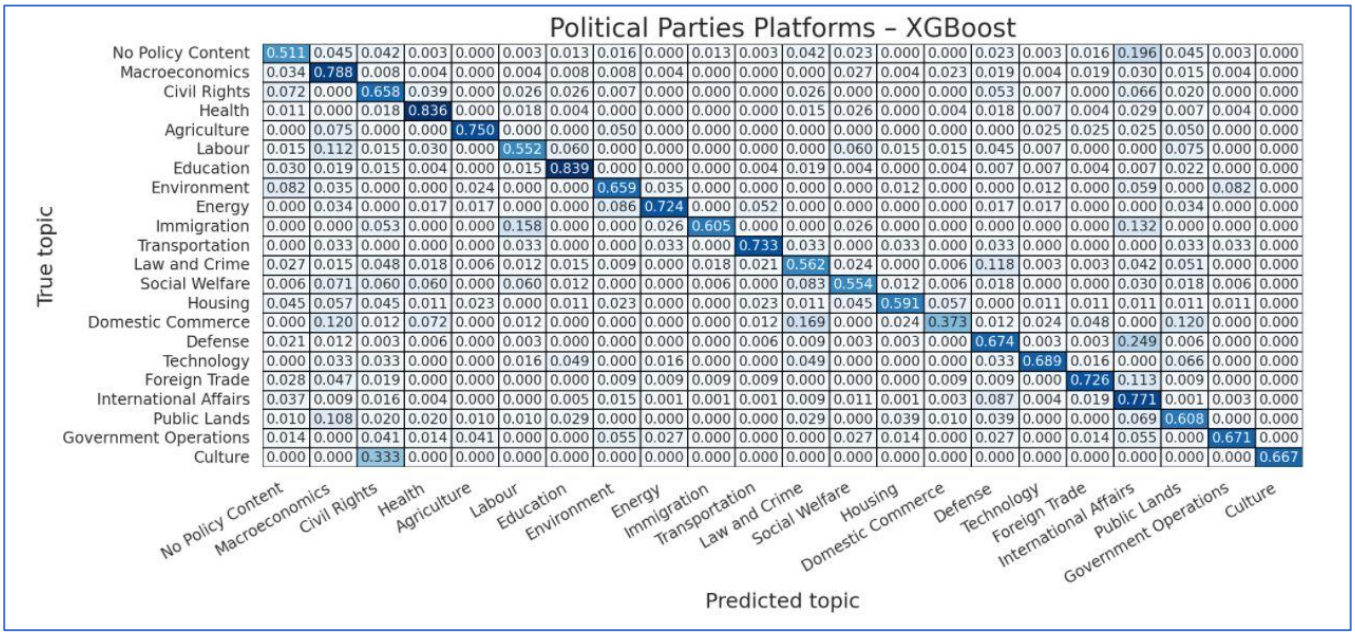
The following chart shows the distribution of the US Parties Platforms Train subset after adding examples from the US State of the Union Speeches dataset for those categories presenting less than 1000 examples in the original Train subset.



Appendix 3. Confusion matrices

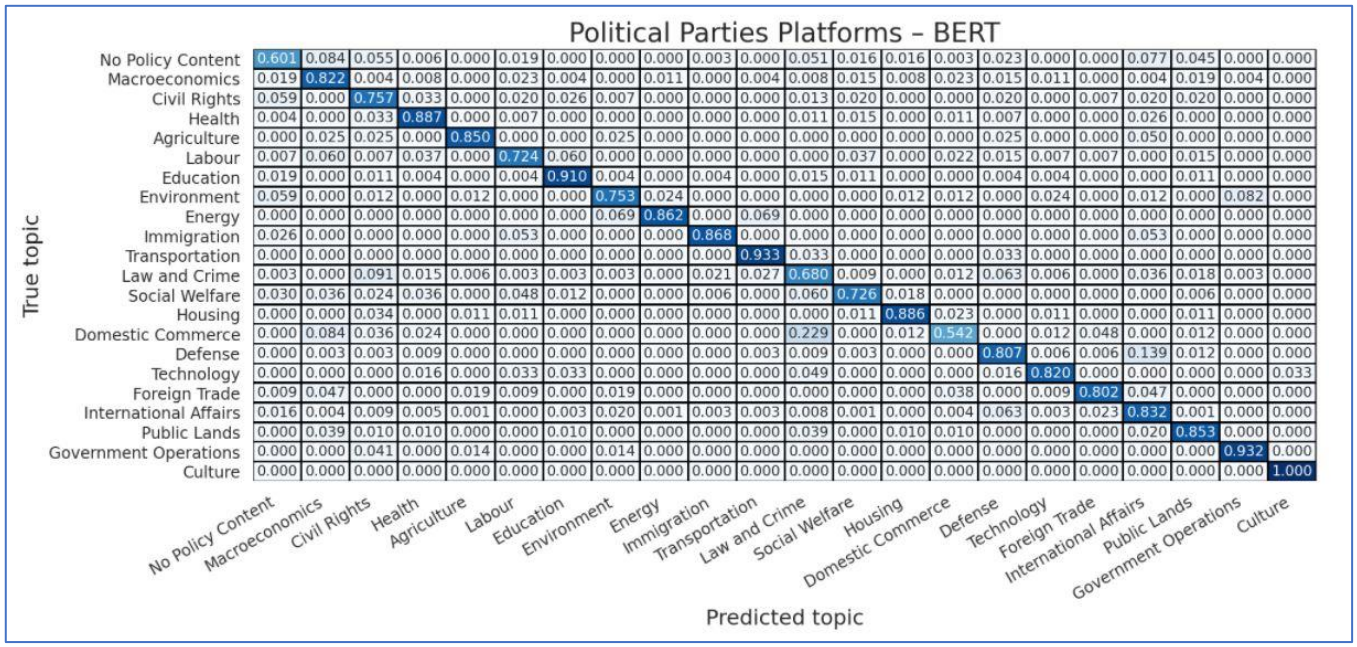
Confusion Matrix 1



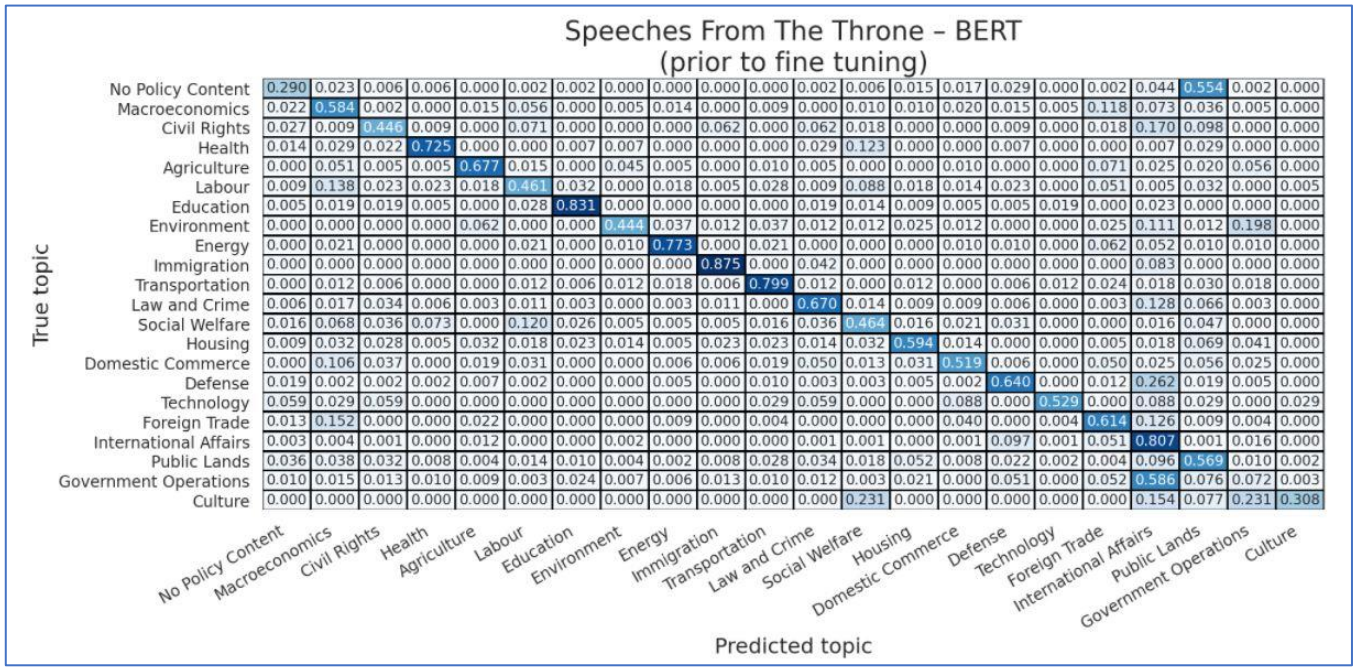


Confusion Matrix 2

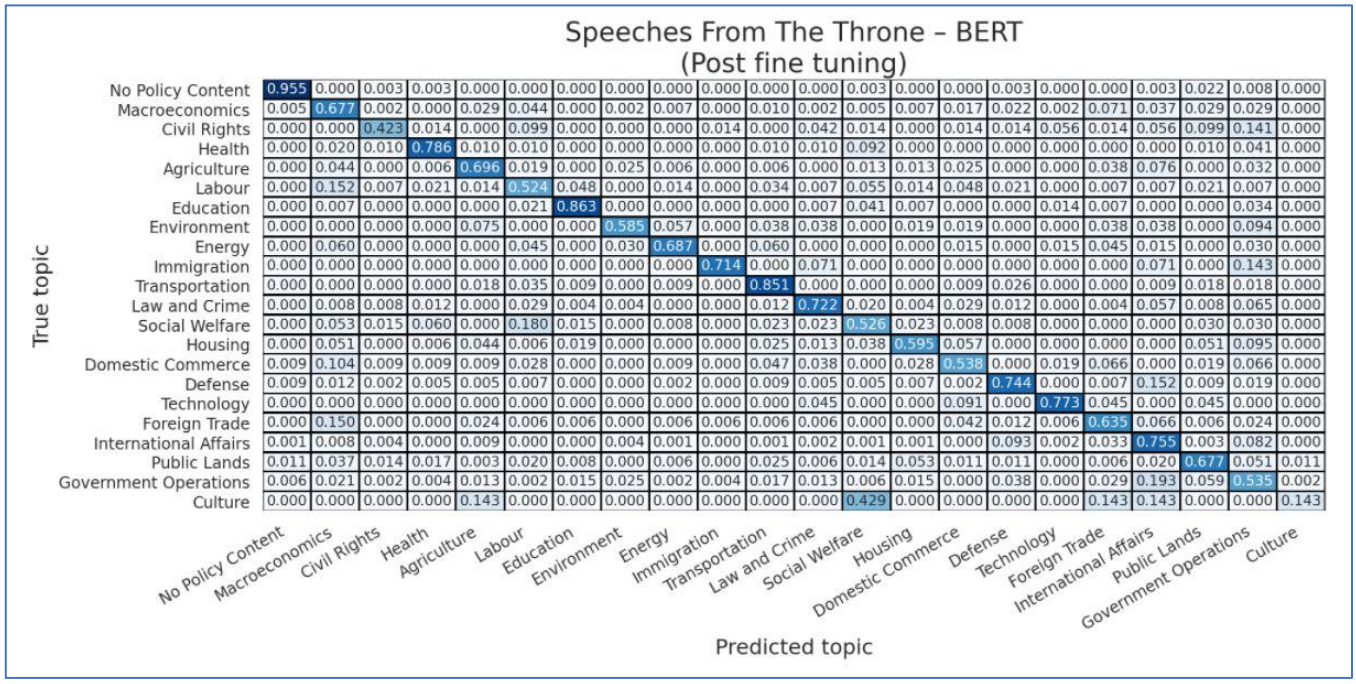
Confusion Matrix 3



Confusion Matrix 4



Confusion Matrix 5



Appendix 4. Some examples of misclassification

The following are examples of sentences that were misclassified by the BERT model, from the US Parties Platforms data.

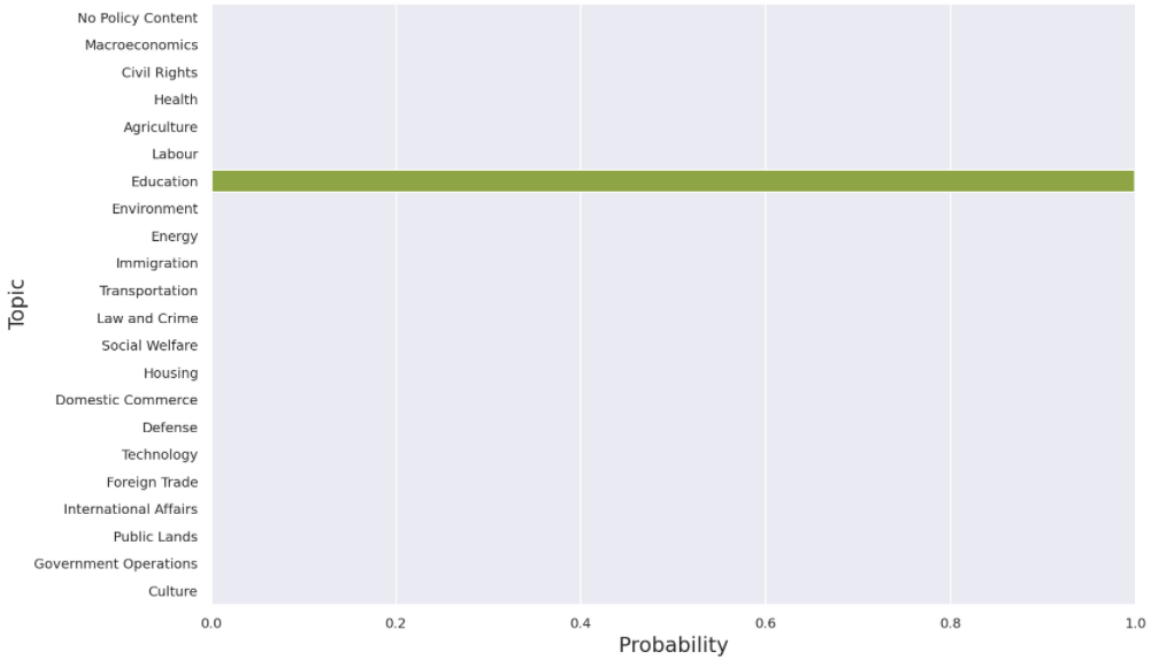
EXAMPLE 1:

Prior sentence: "*America was the pioneer of universal education;"*

Sentence*: "now America must become the pioneer of universal computer literacy."*

True topic: Technology

Prediction: Education



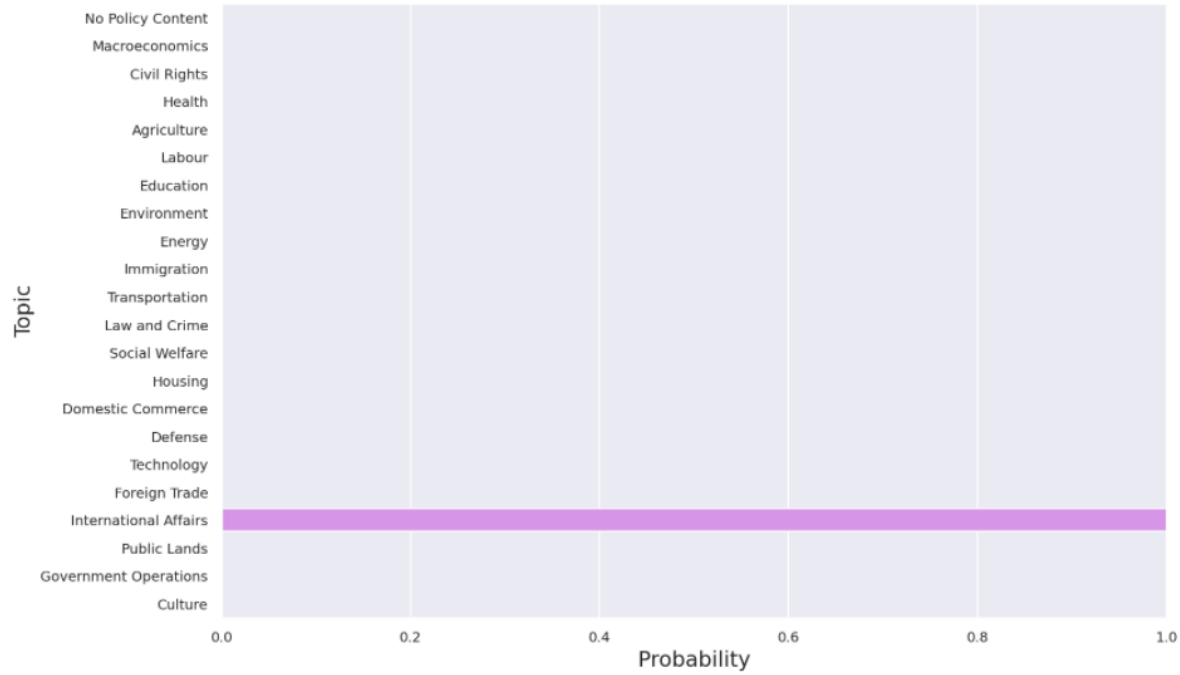
EXAMPLE 2:

Prior sentence: *"It is a proud moral achievement for members of our coalition."*

Sentence*: "We have full confidence in the plan for Iraqi self-government that is currently being implemented by Iraq's interim government."*

True topic: Defense

Prediction: International Affairs



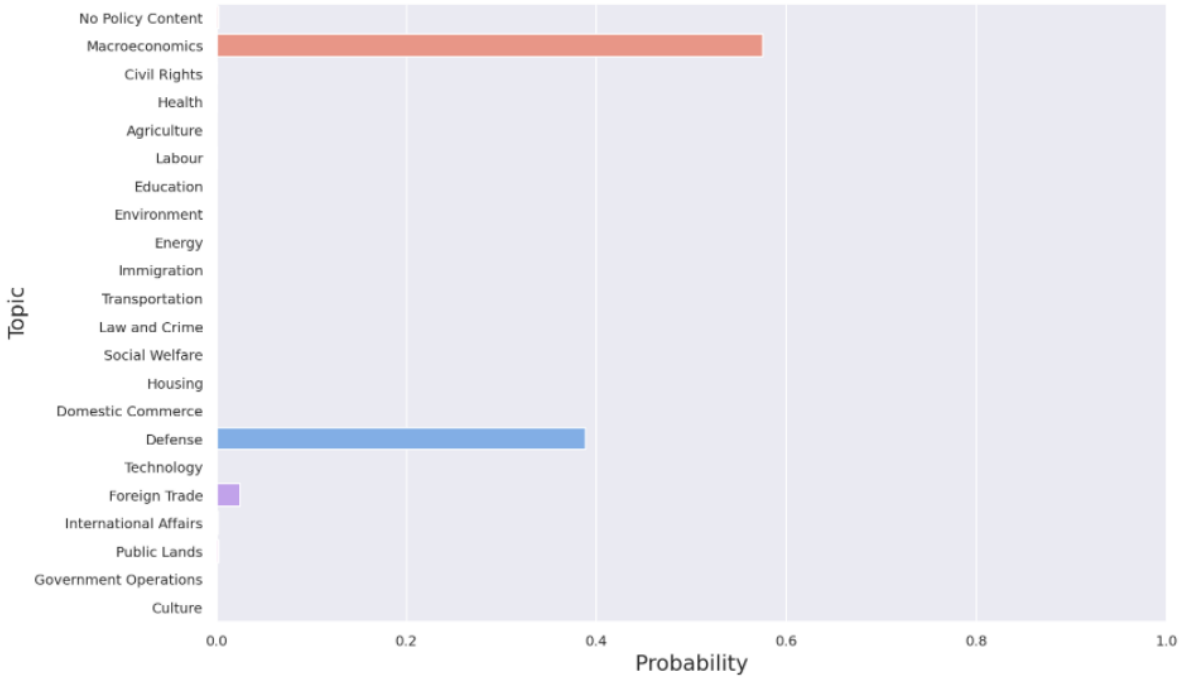
EXAMPLE 3:

Prior sentence: *"We will end "second-best" weapons policies."*

Sentence: *"We will end the false economies which place price ahead of the performance upon which American lives may depend"*

True topic: Defense

Prediction: Macroeconomics



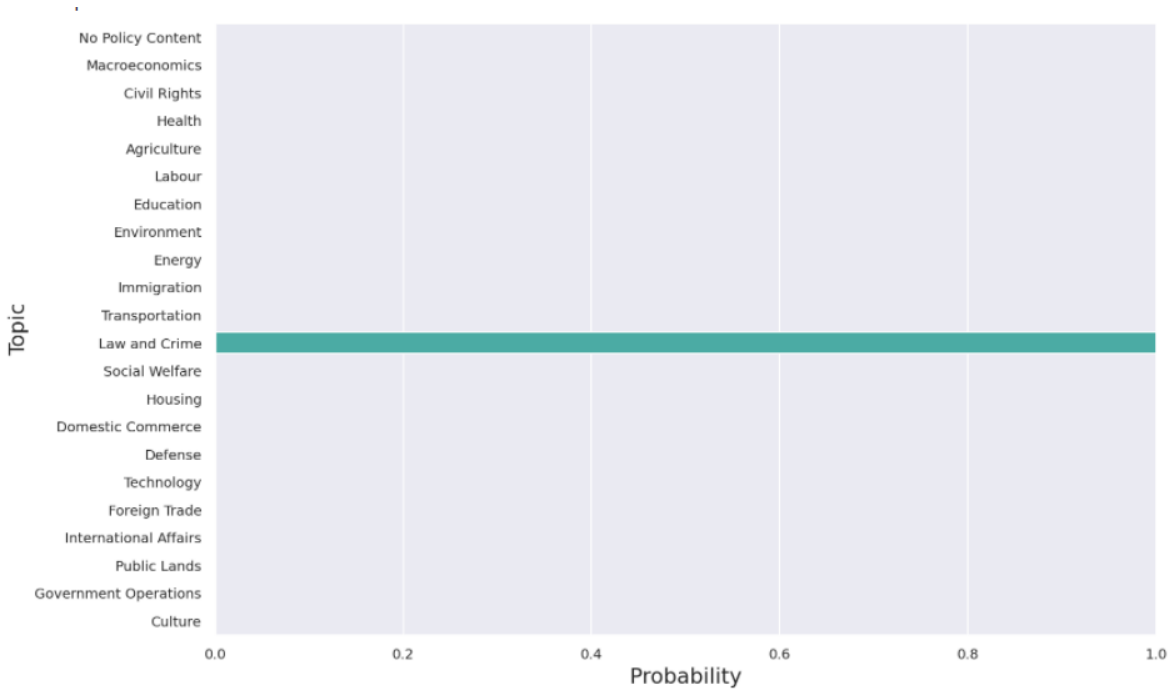
EXAMPLE 4:

Prior sentence: *"Thanks to this law, victims can make one phone call to report the crime to alert all three major credit rating agencies and to protect their credit ratings."*

Sentence: *"We further praise President Bush and Republicans in Congress for passing the Identity Theft Penalty Enhancement Act, which provides a real deterrent by toughening the prison sentences for those who use identity theft to commit other crimes, including terrorism."*

True topic: Domestic Commerce

Prediction: Law and Crime



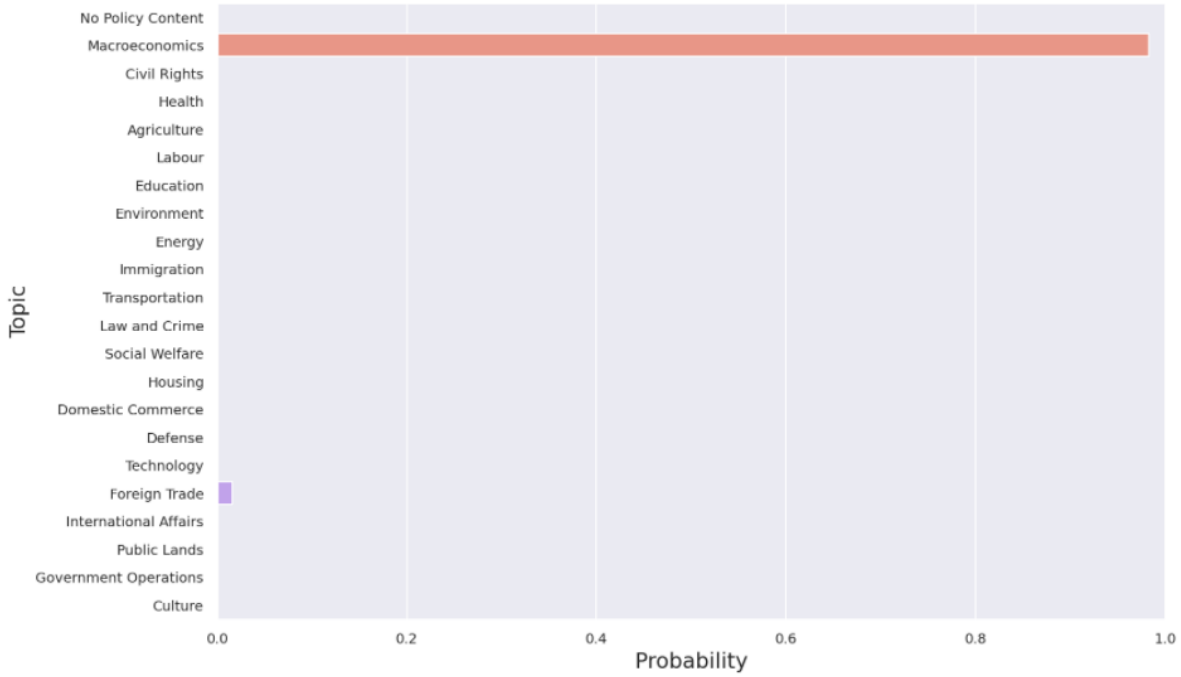
EXAMPLE 5:

Prior sentence: *"The strength of nations, once defined in military terms, now is measured also by the skills of their workers, the imagination of their managers and the power of their technologies.*

Sentence: *"Either we develop and pursue a national plan for restoring our economy through a partnership of government, labor and business, or we slip behind the nations that are competing with us and growing."*

True topic: Foreign Trade

Prediction: Macroeconomics


No Policy Content
Macroeconomics
Civil Rights
Health
Agriculture
Labour
Education
Environment
Energy
Immigration
Transportation
Law and Crime
Social Welfare
Housing
Domestic Commerce
Defense
Technology
Foreign Trade
International Affairs
Public Lands
Government Operations
Culture
Topic
0.0
0.2
0.4
0.6
0.8
1.0
Probability