UC3M at PoliticEs 2023: Applying The Basics

Brandon Solo[1], Martha María Del Toro Carballo[1], Manuel Santiago Fernández Arias[1] and Isabel Segura Bedmar[1]

[1]*Universidad Carlos III de Madrid, Computer Science Department, Av. de la Universidad, 30, 28911, Leganés, Madrid, Spain*

Abstract

In this article we describe our participation in PoliticEs-2023 competition, in which the objective is to classify a tweet based on the political ideology of his/her author. We explore different basic approaches such as Support Vector Machines, Convolutional Neural Network and BERT which resulted in us being 11th out of 12 participants.

Keywords

Author profiling, Transformers, Deep Learning, Political ideology,

1. Introduction

The ability to infer the political ideology of a person with a given text as input is an extremely valuable tool for understanding a person's ideals and values, which when understood properly can shape how certain politicians present themselves when running for a position.

However, this technology also inherently presents issues regarding the privacy of the author of the texts being analyzed. With this task in particular, we are not only asked to infer the political ideology of the author of the tweet but also their gender and profession which, if easily inferred, can also present possible undesirable privacy breaches that allow for targeted spreading of misinformation and fake news [1] [2]. Furthermore, obtaining the extremism of the political ideology through time can allow us to study the polarization of these ideologies. Therefore it is beneficial to our knowledge to develop and understand the technology in order to investigate its possible consequences.

In this article, we describe the approaches used in order to tackle this task for the PoliticEs2023 competition [3]: Support Vector Machines (SVM) [4], Convolutional Neural Network (CNN) [5], transformers models such as [6] and [7]. This work will join others in the 2023 overview [8].

The classical approaches provided acceptable results but with no visible paths forward for improvement, whilst the transformers approach was incomplete with much potential for improvement.

IberLEF 2023, September 2023, Jaén, Spain

100405959@alumnos.uc3m.es (B. Solo); 100486134@alumnos.uc3m.es (M. M. D. T. Carballo); manuefer@inf.uc3m.es (M. S. F. Arias); isegura@inf.uc3m.es (I. S. Bedmar)

0000-0002-7810-2360 (I. S. Bedmar)

2. Task overview

As previously stated, the objective of this task is to predict the political ideology of a user from a given set of tweets, as well as gender and profession. For each user, a set of 80 tweets is provided, as well as, his / her political ideology (binary and multiclass), gender and profession. The possible values of each trait are as follows:

- gender: male, female.
- profession: political, journalist, celebrity.
- pib: left, right.
- pim: left, moderate left, right, moderate right.

The dataset provided is an extension of the PoliCorpus 2020 dataset [9] and the corpus used for the PoliticES 2022 shared task [10], with the tweets being authored by Spanish politicians, journalists and celebrities whose political affiliations are publicly known.

2.1. Dataset

The evaluation dataset provided contains a total of 2250 clusters of 80 tweets each, amounting to a total of 180,000 tweets. Each cluster is represented by an id (for example, 0008c4fab9e97623a60380ee9c88cb20). Then, all tweets belonging to this cluster are associated with this id. Moreover, each cluster is annotated with the following features: gender, profession, ideology-binary, and ideology-multiclass.

These groupings were made intentionally for three main reasons. The first reason is so that the original authors of these tweets be anonymized for privacy, the second reason is to eliminate any possibility of bias towards any specific author, and the final reason is to not be constrained to having to infer the gender, profession and political bias of a tweet author from a single tweet that is as rich in political bias information as *"Buenos días"* ("Good morning" in Spanish).

Around 66% of users are males. Regarding the professions, 61.6% of clusters are journalists, 33.4% are politicians, and 5% are celebrities. More than 55% of users have a left ideology. Figure 4 shows the distribution of the political ideologies: left, moderate left, moderate right, and right. Users with a left ideology are more frequent than users with a right ideology. Moreover, moderate ideologies are more common than the other ideologies.

Another interesting aspect that could be extracted from this data is to observe whether or not the political ideology could be associated with the gender and/or the profession of the author. Tables 1 and 2 show the relation between gender and ideology, and profession and ideology, respectively. We can see that women are twice as likely to have a "left" ideology than men. Moreover, celebrities are almost four times as likely to have a "left" ideology than the other two professions. This information could provide useful warnings about potential biases that this dataset may press on our techniques when applied to tweets about female celebrities.

This dataset was manually stratified-split into training, validation and test sets for our approaches by proportionally extracting tweets from each cluster. We decided on using 70% for training, 15% for validation and 15% for testing to compensate on the underfitting we were expecting. In other words, the distributions of classes to that of the original dataset that was provided were maintained.

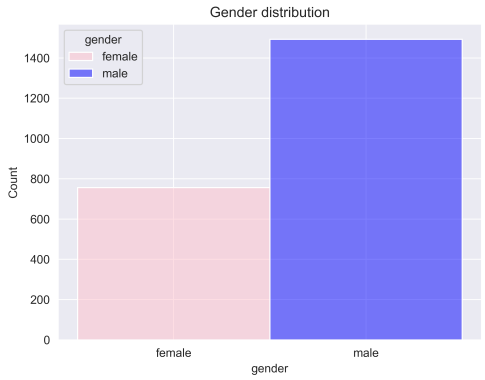

Figure 1: Gender distribution

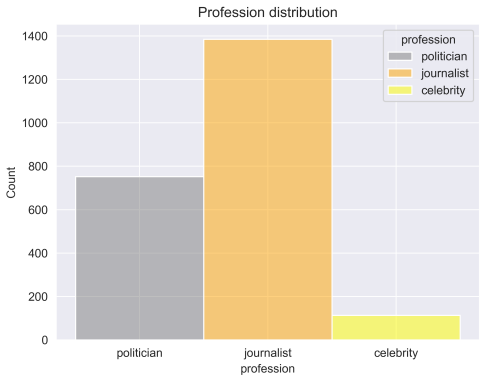

Figure 2: Profession distribution

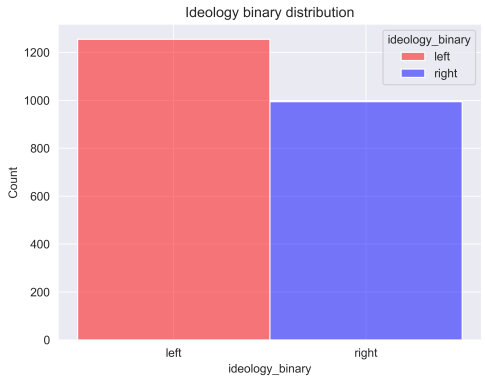

Figure 3: Ideology Binary distribution

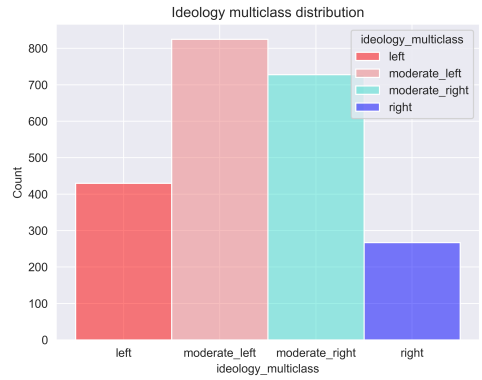

Figure 4: Ideology Multiclass distribution

Table 1
Distribution of gender by ideology

	left	moderate_left	moderate_right	right
female	0.076	0.143	0.092	0.026
male	0.116	0.226	0.232	0.092

Table 2
Distribution of profession by ideology

	left	moderate_left	moderate_right	right
celebrity	0.002	0.036	0.008	0.004
journalist	0.104	0.223	0.223	0.066
politician	0.085	0.107	0.093	0.049

3. Related work

According to the previous year's overview paper [10], most works used transformers as their best approach. In fact, the winning submission was LosCalis' work [11] which used an architecture based on pre-trained Spanish BERT [12] and RoBERTa [13] models and were trained with an extended dataset that they collected, obtaining a macro-F1 of 90.28%. The submission in second place by NLP-CIMAT [14] also used a transformer-based approach using a BETO model trained in the political language domain that was used to classify at the tweet level. These classifications are then aggregated by author to determine the majority vote (which classification label appeared the most frequently) to determine the label of the author, with a macro-F1 of 89.09%. However, the submission in third place by Alejandro Mosquera [15] merely used L2-regularized logistic regression with a macro-F1 of 88.9%, suggesting that classical machine learning algorithms can provide similar results to those obtained with transformers. These works inspired us to test both classical machine-learning approaches such as CNNs or SVMs as well as a transformer-based approach with pre-trained BERT models (all of which were obtained from HuggingFace [16]) and the use of majority voting of the tweets for label classification.

4. Systems

In this section, we detail the three main approaches taken to solve this task.

4.1. SVM

The Support Vector Machine (SVM) [17] is a successfully used supervised machine learning algorithm for text classification [18]. SVM is based on the concept of finding an optimal hyperplane that separates data points of different classes in a higher-dimensional space. SVM present themselves as a viable option worth considering when it comes to the extraction of political ideology from a corpus of texts. In this study, we tackle this classification task from both binary and multiclass perspectives. The following steps outline the approach taken to achieve this objective.

For binary classification, we have selected only two fields, namely "tweet" and "ideology_binary", from the dataset and proceeded to separate them into training, validation, and evaluation sets. Before performing the classification task, the texts need to be represented as real-number vectors. To prepare the text for classification purposes, various processing tasks are employed, which involve utilizing the NLTK (Natural Language Toolkit)[19] library in Python. These tasks include converting all texts to lowercase, tokenizing, removing stopwords, applying stemming, filtering out words with fewer than three characters, and excluding words with digits or special symbols.

Text processing continues with BoW [20] and TF-IDF [21] techniques. The BoW model is trained using CountVectorizer [22], learning the vocabulary and assigning word indices. The training set is transformed accordingly. The same process is applied to the test set. TF-IDF is then trained using TfidfTransformer [23], assigning word weights based on frequency and inverse document frequency. Both sets are transformed. A pipeline implementing SVM is built,

searching for optimal parameters with cross-validation. Steps include BoW, TF-IDF, and SVC [24] for classification.

Parameters for the SVM algorithm are defined, specifying different values for the kernel type (linear and rbf) and the regularization parameter C (0.1, 1). The GridSearchCV class is then utilized to conduct an exhaustive search for the best parameters. The accuracy scoring metric is employed, and a 3-fold cross-validation (cv=3) is set.

Once the exhaustive search is completed, the best parameter set is determined: {svm_C: 1, svm_kernel: rbf}, with an accuracy of 65.7%. Predictions are subsequently made using the trained and fitted model on the test dataset, returning the predicted labels for each instance. These labels represent the classes or categories assigned to the test data based on the model's classification.

For the multiclass classification task, we used the fields "tweet" and "ideology_multiclass". For the trained model, the best parameter set is determined as {svm_C: 1, svm_kernel: rbf}, with an accuracy of 48.4%.

4.2. CNN with WordEmbedding

As an initial step for the proposed analysis, a convolutional neural network (CNN) was employed. CNN has been succesfully applied to many text classifications tasks [5]. In our specific case, we aimed to enhance the CNN's performance by incorporating word embeddings [25].

The sequential model architecture was utilized, initialized with the "word2vec-google-news-300" word embedding, as depicted in the schematic representation. Initially, we experimented with the "glove-wiki-gigaword-50" word embedding model, trained on an extensive corpus consisting of Wikipedia articles and news texts. However, optimal results were not achieved due to the limited embedding size (50) and the semantic gap between the training corpus and the competition data domain. Consequently, we opted for the "word2vec-google-news-300" word embedding model, which formed the foundation of our final CNN-based model. This particular word-embedding was trained on an extensive corpus of Google news, enabling it to capture semantic and syntactic information of words in the context of news and related texts. Although our case study focused on the domain of tweets, the utilization of news data proved beneficial in terms of political orientation classification. The chosen word embedding boasted a considerably larger feature vector size, comprising approximately 300 tokens. It should be noted that both word embedings were pre-trained with English text. At the beginning of the task solution, we attempted to use a pre-trained word embedding model trained on the 'Spanish CoNLL17 corpus.' However, the preliminary tests yielded unsatisfactory results. Therefore, due to computational limitations, it was exclude from the study.

Three one-dimensional filter layers have been added, consisting of 128, 64 and 32 filters, respectively. All of these layers use a fixed kernel size of 4. These layer are followed by corresponding pooling and flattening layers. The incorporation of multiple one-dimensional filtering layers proved to be highly effective in processing and extracting features from textual data. To facilitate the transition from convolutional layers to dense layers within the neural network, we employed MaxPooling and Flatten layers for distinct purposes. MaxPooling was employed to reduce the dimensionality of the features extracted by the convolutional layers while retaining the most salient features. On the other hand, the Flatten layer served the purpose

of transforming the multidimensional data into a one-dimensional vector. These layers played a pivotal role in connecting the convolutional layers with the subsequent dense layers responsible for classification.

To complete the model, two dense layers were included, utilizing the 'relu' activation function for the first dense layer and the 'sigmoidal' activation function for the final layer, given that our system was designed and evaluated solely for binary classification.

For multiclassification, the softmax activation function was used in combination with the sparse_categorical_crossentropy function in order to minimise the discrepancy between the predicted probability distributions and the actual labels during the training process. This allowed the model to learn to assign the correct probabilities to each class based on the training data.

During the training process, the network was trained for 10 epochs on each problem. However, an early stopping function was implemented to halt the training if no improvement in error was observed over three consecutive epochs. In this particular scenario, the training was terminated after four epochs.

4.3. Transformers

The choice of testing transformers for this task was almost trivial given their stellar track record in text classification problems. We evaluated several transformers such as BERT (the multilingual model *bert-base-multilingual-uncased* [6]) , XLM-RoBERTa [26] and a its version trained on tweets (*cardiffnlp/twitter-xlm-roberta-base*) [7]. We also desired to include models with a larger acceptance of tokens (*xlm-roberta-large* [27] and *facebook/xlm-roberta-xxl* [28]) but our hardware limitations did not allow for that.

However, after some manual experimentation, we decided to only use the *bert-base-multilingual-uncased* for all classification tasks due to it providing the most information through its tokenizer by requiring the *input_ids*.

We classify each tweet, and then, we classify a given user by selecting the most frequent label assigned to his / her set of tweets.

This is how NLP-CIMAT [14] made their approach, which is clearly an imperfect solution since some sets of tweets could be tainted by unimportant tweets that larger models would have ignored, but the system worked nonetheless. Future work would focus on weighing the importance of each tweet and its classification.

Still, further progress could clearly be made, and the simplest change that made a difference was a preprocessing of the texts themselves. Given that the nature of the texts were tweets, many contained emojis. For a human, the meaning behind these emojis would be much easier to grasp, but for a transformer architecture, their appearance amounted to a brand new word being used. However, the nature of emojis is to be complementary to the text, with the latter being the core source of information that the author desires to express. Therefore, removing the emojis was the only logical solution, and this final preprocessing step made the biggest difference in our experimentation.

Nevertheless, merely using the default configuration and settings provided by HuggingFace's Trainer would not suffice for a variety of reasons. One of them was the floating point precision used for calculations. Given that we were training this model for classification on limited hardware, we did not require the precision that regression would benefit from. Therefore,

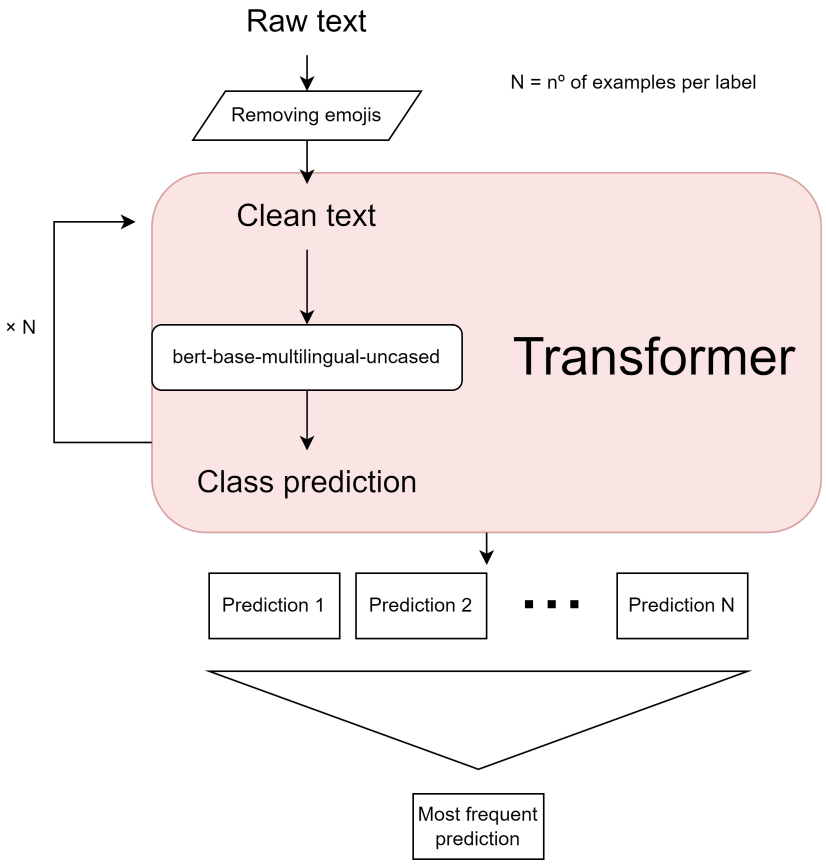

Figure 5: Transformer solution architecture

instead of using FP32 (floating point with 32 bits) by default, which slowed down training and evaluation as well as occupied too much memory, FP16 (floating point with 16 bits) was used instead.

During our experiments, we see that the model provided very inconsistent scores, varying quite drastically throughout different training sessions. We came to the conclusion that the learning rate might be the cause [29], and so the default value of 5×10^{-5} was reduced to 5×10^{-5}, which proved to be the right choice since the scores were more consistent.

5. Results

5.1. SVM results

The attained accuracy reached a value of 67%. The enclosed confusion matrix Table (6) illustrates the comprehensive outcomes obtained through this approach. It is crucial to emphasize that these outcomes are specific to the training set and do not represent the final input data for the competition.

Table 3
SVM results of (binary) political ideology on our test dataset.

	Precision	Recall	F1-score
left	0.67	0.82	0.73
right	0.67	0.47	0.55
macro avg	0.67	0.65	0.64

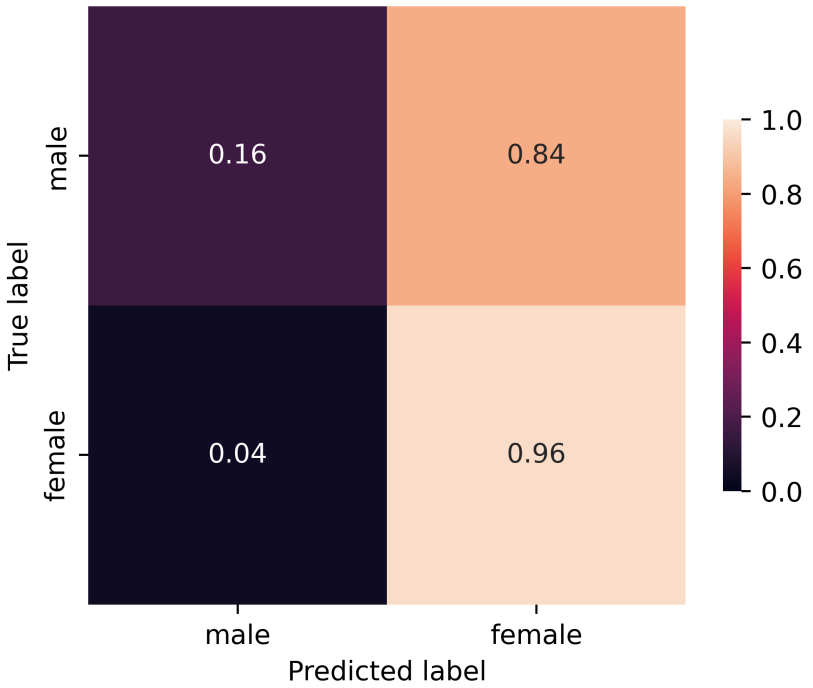

Figure 6: SVM Confusion Matrix of binary classification of political ideology (document level)

Table 4
SVM results of the multiclassification of political ideology.

	Precision	Recall	F1-score
left	0.55	0.15	0.24
moderate_left	0.48	0.77	0.59
moderate_right	0.49	0.49	0.49
right	0.69	0.07	0.13
macro avg	0.56	0.37	0.36

The attained accuracy reached a value of 49%. Once again, a confusion matrix Table (7) is generated, providing insight into and evaluation of the classification model's performance based on the actual and predicted labels.

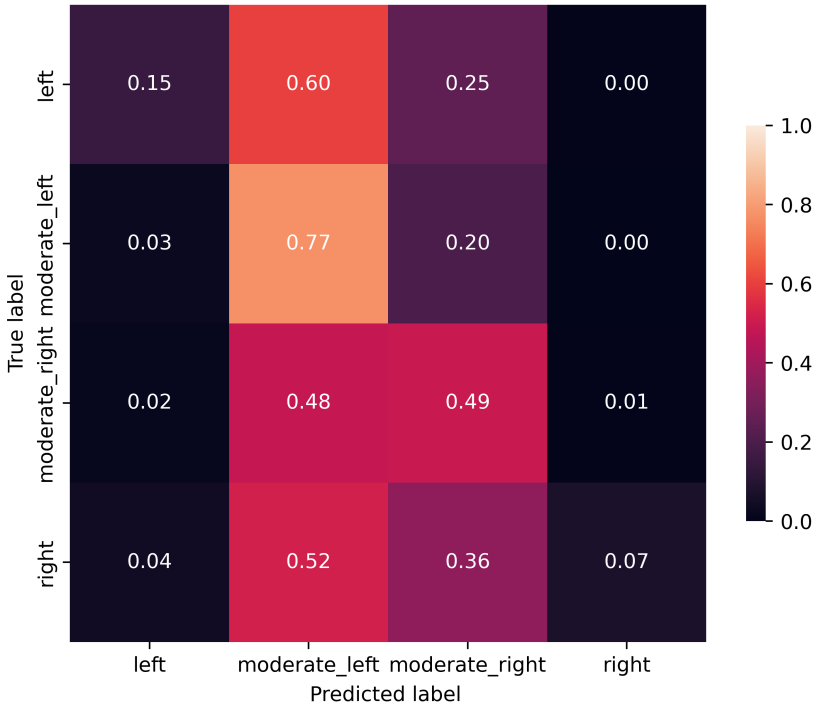

Figure 7: SVM Confusion Matrix of multiclassification of political ideology (document level)

The SVM model can encounter classification failures in situations such as non-linearly separable data, overlapping data, class imbalance, presence of noise in the data, incorrect kernel selection, and high dimensionality. For example, the tweet *Congratulations, @user for your victory in these elections in which Brazil has decided to bet on progress and hope. Let's work together for social justice, equality and against climate change. Your success will be the success of the Brazilian people. Parabéns, Lula!* was classified with the 'left' label, but its true label is right. The classification of a sentence as ideologically left-wing or right-wing can depend on various factors, such as context, tone, and keywords used. In the case of the mentioned sentence, it congratulates someone for their victory in elections and emphasizes a commitment to progress and hope. Additionally, it mentions collaboration for social justice, equality, and against climate change. These topics are often associated with left-wing ideologies that emphasize the importance of equity, justice, and environmental protection. However, it's important to note that ideological interpretation can vary depending on individual perspectives and understanding of the terms used. Some people may interpret this sentence as expressing support for left-wing policies due to the mention of social justice and equity, while others may consider that these concerns can also be shared by certain right-wing ideologies. Another example is the tweet *We are together against male violence. Denying it is a form of exercising it. To take it to the rostrum of the Congress is to cross an intolerable line... We will go forward with conviction and determination..... All my support, @user* ., which was classified as right, but its true label is left. It seems that the

classification of this statement as ideologically right-wing may not align with its content. The sentence expresses a strong stance against male violence and emphasizes the importance of taking action. These sentiments are not inherently exclusive to any particular political ideology and can be supported by both left-wing and right-wing perspectives. It's important to note that political ideologies are multifaceted, and individuals from various ideological backgrounds can share common concerns and goals, such as opposing violence and advocating for social change. The classification of the sentence as right-wing might be subjective and dependent on the specific criteria or analysis framework being used.

SVM was also used for gender classification (see Table 5). The evaluation on our test dataset yielded a macro-F1 of 54& and an accuracy of 70%. We can see that a higher F1 is obtained when identifying male gender. This may be due to the sample of tweets authored by males being twice as large as that of females.

Table 5
SVM results of the binary classification of gender.

	Precision	Recall	F1-score
female	0.67	0.16	0.26
male	0.71	0.96	0.82
macro avg	0.69	0.56	0.54

Table 6
SVM results of the multiclassification of the profession

	Precision	Recall	F1-score
celebrity	0.97	0.05	0.09
journalist	0.74	0.93	0.82
politician	0.78	0.53	0.63
macro avg	0.83	0.50	0.52

Table 6 shows the results of SVM for the classification of professions. The best results are obtained for journalists with an F1 of 82%, followed by politicians with an F1 of 63%. The detection of the profession of celebrities yielded unsatisfactory outcomes due to the scarcity of data available on their tweets.

5.2. CNN results

Table 7 shows the results of CNN for binary classification of ideology. CNN achieved an of 67%.

CNN is prone to failure when any of the following scenarios are present in the problem at hand: ambiguous language, lack of context, and the use of modern terms and references to users or entities. For example, the tweet "They bomb civilians because they lose the war. And they know it." was classified with left, but its true label is right. This tweet certainly looks like a left-wing politically-oriented tweet because of its concern for the civilian population. However, the lack of context results in a false prediction. Another example is the tweet "These are my

Table 7
CNN results for binary classification of political ideology on our test dataset.

	Precision	Recall	F1-score
left	0.71	0.73	0.72
right	0.64	0.61	0.62
macro avg	0.67	0.67	0.67

thoughts today in the @dailyuser.", classified as right, but its true label is left. In the dataset, there are no names of the people referred to in the privacy terms, so the tags are replaced by @user, making it difficult to identify the targeting.

Table 8 shows the confusion matrix for CNN. It is important to emphasize that these results were obtained on our test dataset, and not using the final input data of the competition.

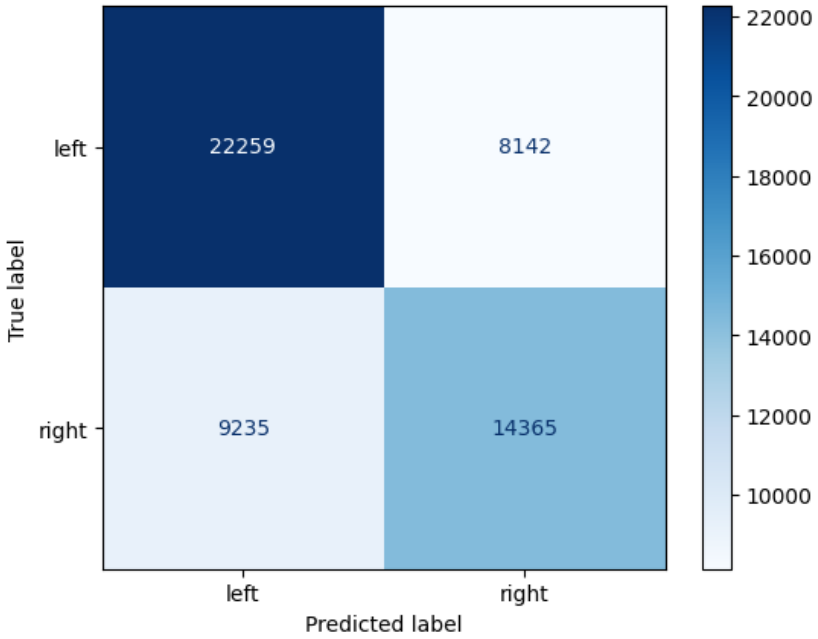

Figure 8: CNN Confusion Matrix of binary classification of political ideology (document level)

In the multi-class classification of political ideology, as expected, poor results were obtained (see Table 8), since multi-class classification is a more complex task than binary classification.

We also provide results for the gender classification on our test dataset (see Table 9). We can see that the model achieves better results for male than for female. This is closely linked to the fact that the dataset contains more instances of male users than of female users.

We also applied CNN to detect the professions of the users (see Table 10). The accuracy obtained was 73% and a macro F1 of 58%. However, it should be noted that given the lack of data on tweets written by celebrities, the detection of this profession has very poor results. As is

Table 8
CNN results of the multiclassification of political ideology on our test dataset

	Precision	Recall	F1-score
left	0.40	0.30	0.34
moderate_left	0.51	0.62	0.56
moderate_right	0.50	0.54	0.52
right	0.43	0.23	0.31
macro avg	0.46	0.42	0.43

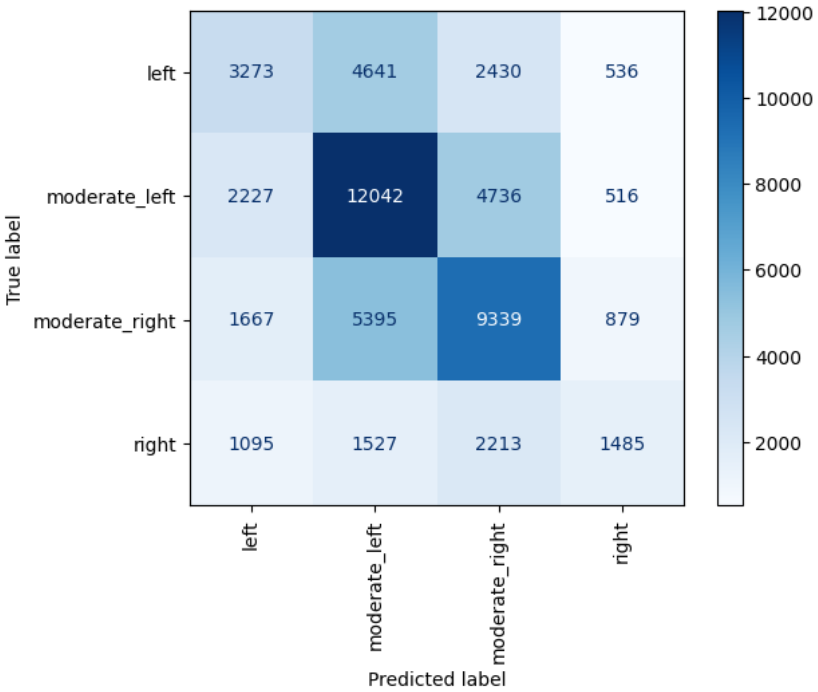

Figure 9: CNN Confusion Matrix of the multiclassification of political ideology (document level)

Table 9
CNN results for the classification of gender.

	Precision	Recall	F1-score
female	0.52	0.44	0.48
male	0.73	0.79	0.76
macro avg	0.62	0.61	0.62

well known in multiple classification scenarios, accuracy is not the best measure of evaluation, so the macro average will be used as an indicator.

Table 10
CNN results of the multiclassification of the profession

	Precision	Recall	F1-score
celebrity	0.29	0.20	0.24
journalist	0.79	0.81	0.80
politician	0.66	0.67	0.67
macro avg	0.58	0.56	0.57

5.3. Transformers results

The evaluation of the transformers produced very similar results scores. This is likely due to the insufficient amount of training dataset, since only one epoch was possible per model.

Table 11
Results for political ideology on our test dataset

	Precision	Recall	F1-score
left	0.74	0.95	0.83
right	0.87	0.50	0.64
macro avg	0.81	0.73	0.74

For the classification of political ideology, both classes ("left" and "right") were relatively balanced in representation in the dataset (see Figure 3), and in fact it was the task in which we performed the best. However, the multi-class version ("left","moderate left", "moderate right", "right") was our worse performing task, even with attempts to undersample and avoid class imbalances. Despite this, we suspect the poor performance is due to the fact that a "left" tweet, as classified by the model for the binary task, could either be a "moderate left" or a "left" tweet in the multi-class task, and incorrectly choosing either one of these options is counted as equally wrong as classifying the tweet as a "moderate right" or "right" tweet. A suggestion for future competitions would be to define the task as a regression one instead of multi-class so that penalties for mislabelling like this are not so unnecessarily severe.

An important detail to note is that the nature of tweets as texts are characterized by their brevity and the public context in which they appear, often relying on the reader's knowledge of the current state of the world for the intended message to be conveyed correctly. This presents an interesting challenge for any of the techniques that we decide to apply since we lack that context. Also, these tweets often come with images and videos attached, which is the context that the reader of the tweet may not be required to know of previously but still shapes the meaning of the tweet nonetheless. This is yet more context that is being missed. After missing all of this context, we welcomed another peculiarity of tweets, and that is the use of emojis. These small images in texts condense a lot of information that is supplementary to the tweet and sometimes are even used to completely change the meaning. However, after various attempts at training the transformers with and without the emojis, their absence resulted in consistently better performance (this surprisingly included the *cardiffnlp/twitter-xlm-roberta-base* model

[7], which we assumed would tolerate emojis). We suspect that the introduction of emojis as brand new tokens to be taken into consideration was throwing off the Large Language Models' attention from the core text, especially since not all tweets had emojis, and much less did they have the same emojis being used.

Table 12
Transformers results for the binary classification of political ideology on our test dataset

	Precision	Recall	F1-score
left	0.80	0.87	0.83
right	0.81	0.73	0.77
macro avg	0.81	0.80	0.80

Table 13
Transformers results for the multiclassification of political ideology on our test dataset

	Precision	Recall	F1-score
left	0.92	0.33	0.49
moderate_left	0.58	0.92	0.71
moderate_right	0.72	0.66	0.69
right	0.91	0.40	0.56
macro avg	0.78	0.58	0.61

Table 14
Transformers results for gender classification on our test dataset

	Precision	Recall	F1-score
male	0.91	0.82	0.87
female	0.71	0.85	0.77
macro avg	0.81	0.84	0.82

Table 15
Transformers results for the classification of professions on the test dataset

	Precision	Recall	F1-score
celebrity	1.00	0.04	0.08
journalist	0.83	0.99	0.91
politician	0.97	0.76	0.85
macro avg	0.93	0.60	0.61

As the tables show, the best and worst macro-F1 scores are obtained in the political ideology tasks with the multiclass version being the worst. Unsurprisingly, this lacking performance is mostly hindered by the F1-score obtained for classifying the more extreme classes, which also

happen to be the classes with the fewest examples to train on. The same situation is occurring in the task of classifying profession, as there are barely any celebrity tweets. The confusion matrices (which can be found in the appendix) simply confirm this as the more frequently appearing classes are guessed most often. All of these signs clearly point to the obvious fact that the model is underfitting.

However, we suspect that some of the models would be the best contenders in the future with better resources for different reasons:

- *cardiffnlp/twitter-xlm-roberta-base* [7]: despite the results we obtained, we presume that this model could flourish with emoji-rich tweets if given the right resources for training.
- *bert-base-multilingual-uncased* [6]: unlike the RoBERTa based models, this model additionally tokenizes the text with "token type ids", which essentially gives the model more information as to how the text is split up into sentences. A larger version of this model (one that would accept more than 512 tokens) could take all of the tweets associated to a single identifying label at once for classification, instead of classifying each tweet individually and then selecting the most frequent classification (choosing randomly if there is an even split among two or more choices) as the simpler models do now.

One possible improvement would be to use the binary classifier to double-check the multiclass classifier's result. If there were a discrepancy between the two results, that of the binary classifier's choice would be selected, mostly choosing the "moderate <X>" choice since the datasets have more of the moderate tweets than the more extremist ones.

5.4. Submissions

The one and only submission from our team during the competition was using the transformer approach. The results are shown in the following tables.

Table 16
Transformers results for the binary classification of political ideology on the provided test dataset

	Precision	Recall	F1-score
left	0.80	0.87	0.83
right	0.81	0.73	0.77
macro avg	0.81	0.80	0.80

The most surprising result is how the F1-score of the multiclass "right" value obtained in our dataset compares to the corresponding F1-score from the provided test dataset. The difference can be explained by observing the confusion matrix on the provided dataset (see Figure 10), where most of the "right" predictions were mistakenly labelled as "moderate_right" by our transformer.

This implies that our transformer is fairly capable of detecting "right" tweets but it is unable to determine the degree to which how right-leaning those tweets are. This circles back to our suggestion that perhaps re-framing the multiclass ideology task as a regression problem instead of a classification one would be a fairer metric.

Table 17
Transformers results for the multiclassification of political ideology on the provided test dataset

	Precision	Recall	F1-score
left	1.00	0.19	0.32
moderate_left	0.48	0.96	0.64
moderate_right	0.52	0.37	0.43
right	1.00	0.03	0.06
macro avg	0.75	0.39	0.36

Table 18
Transformers results for gender classification on the provided test dataset

	Precision	Recall	F1-score
male	0.87	0.74	0.80
female	0.56	0.74	0.64
macro avg	0.71	0.74	0.72

Table 19
Transformers results for the classification of professions on the provided test dataset

	Precision	Recall	F1-score
journalist	0.72	1.00	0.83
celebrity	1.00	0.04	0.07
politician	1.00	0.64	0.78
macro avg	0.91	0.56	0.56

6. Conclusions and future work

In this work, we attempted to infer the political ideology of Spanish tweet authors with several machine learning approaches such as SVM; CNN and transformers with limited success. The lack of computational resources is one of our main limitations, which combined with the unbalanced classes (see Figures 1 and 2) produced biased results. Our weakest performing category was the multiclassification of the political ideology with a macro-F1 of 36.17% (far from first place at 69.13%), and our strongest was the binary political ideology task with a macro-F1 of 73.37% (also far from first place at 89.67%). It is clear that there is much room for improvement.

In the future, we would regard doing the following:

- Use of more computational resources for faster training (allowing for more than just one epoch of training), and models that allow more input tokens to conglomerate multiple tweets for the transformers.
- Addition of a system to previously go through the tweets and assign a numerical score of how much information a tweet may reveal (such as through TF-IDF, for example), thereby only feeding the transformer a subset of the cluster containing the most promising tweets.
- Use of the model for identifying the political ideology as a binary to guide the multiclass

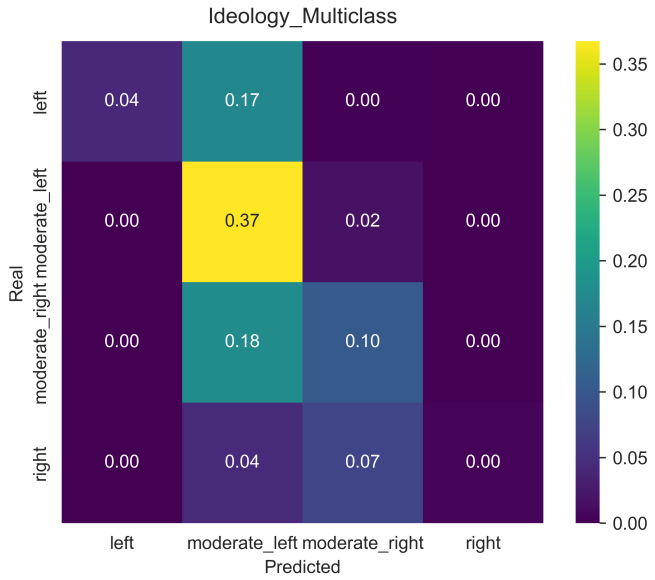

Figure 10: Transformer Confusion Matrix for Ideology (Multiclass) on the provided Dataset (cluster level)

model.

- Extension the dataset to have a more balanced selection of classes for the multiclass political ideology task, since the extremes were lacking in examples.
- Inclusion of a model to separately analyze the emojis and better infer the sentiment of the tweet.

The team has also considered how the multiclass task could be better presented as a regression task to more fairly and accurately represent political inclination in future competitions.

References

[1] M. R. Holman, M. C. Schneider, K. Pondel, Gender targeting in political advertisements, Political Research Quarterly 68 (2015) 816–829. URL: https://doi.org/10.1177/1065912915605182. doi:10.1177/1065912915605182. arXiv:https://doi.org/10.1177/1065912915605182.

[2] R. Shorrocks, Cohort change in political gender gaps in europe and canada: The role of modernization, Politics & Society 46 (2018) 135–175. URL: https://doi.org/10.1177/0032329217751688. doi:10.1177/0032329217751688. arXiv:https://doi.org/10.1177/0032329217751688.

[3] J. A. García-Díaz, S. M. Jiménez-Zafra, M. T. Martín-Valdivia, F. García-Sánchez, L. A. Ureña-López, R. Valencia-García, Overview of PoliticES at IberLEF 2023: Political ideology detection in Spanish texts, Procesamiento del Lenguaje Natural 71 (2023).

[4] N. Cristianini, J. Shawe-Taylor, An introduction to support vector machines and other kernel-based learning methods, 2000.
[5] C. Li, G. Zhan, Z. Li, News text classification based on improved Bi-LSTM-CNN, in: 2018 9th International conference on information technology in medicine and education (ITME), IEEE, 2018, pp. 890–893.
[6] bert-base-multilingual-uncased · hugging face, 2023. URL: https://huggingface.co/bert-base-multilingual-uncased.
[7] F. Barbieri, L. E. Anke, J. Camacho-Collados, XLM-T: Multilingual language models in twitter for sentiment analysis and beyond, 2022. 2104.12250.
[8] S. M. Jiménez-Zafra, F. Rangel, M. Montes-y Gómez, Overview of IberLEF 2023: Natural Language Processing Challenges for Spanish and other Iberian Languages, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2023), co-located with the 39th Conference of the Spanish Society for Natural Language Processing (SEPLN 2023), CEUR-WS.org, 2023.
[9] J. A. García-Díaz, R. Colomo-Palacios, R. Valencia-García, Psychographic traits identification based on political ideology: An author analysis study on spanish politicians' tweets posted in 2020, Future Generation Computer Systems 130 (2022) 59–74. URL: https://www.sciencedirect.com/science/article/pii/S0167739X21004921. doi:10.1016/j.future.2021.12.011, iD: 271521.
[10] J. A. García-Díaz, S. M. J. Zafra, M. T. M. Valdivia, F. García-Sánchez, L. A. U. López, R. V. García, Overview of PoliticEs 2022: Spanish author profiling for political ideology; resumen de la tarea PoliticEs 2022: Perfilado del autor español por su ideología política, 2022. URL: http://hdl.handle.net/10045/127445. doi:10.26342/2022-69-23.
[11] S. S. Carrasco, R. Cuervo, Rosillo, LosCalis at PoliticEs 2022: Political author profiling using BETO and MarIA, 2022. URL: https://ceur-ws.org/Vol-3202/politices-paper1.pdf.
[12] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, 2019. 1810.04805.
[13] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, RoBERTa: A robustly optimized BERT pretraining approach, 2019. 1907.11692.
[14] E. Villa-Cueva, I. González-Franco, F. Sanchez-Vega, A. P. López-Monroy, NLP-CIMAT at PoliticEs 2022: PolitiBETO, a Domain-Adapted transformer for multi-class political author profiling, 2022. URL: https://ceur-ws.org/Vol-3202/politices-paper2.pdf.
[15] A. Mosquera, Alejandro mosquera at politices 2022: Towards robust spanish author profiling and lessons learned from adversarial attacks, 2022. URL: https://ceur-ws.org/Vol-3202/politices-paper3.pdf.
[16] Models - hugging face, 2023. URL: https://huggingface.co/models.
[17] W. S. Noble, What is a support vector machine?, Nature biotechnology 24 (2006) 1565–1567.
[18] K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, D. Brown, Text classification algorithms: A survey, Information 10 (2019) 150.
[19] S. Bird, Nltk: the natural language toolkit, in: Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions, 2006, pp. 69–72.
[20] Y. Zhang, R. Jin, Z.-H. Zhou, Understanding bag-of-words model: a statistical framework, International journal of machine learning and cybernetics 1 (2010) 43–52.
[21] P. Bafna, D. Pramod, A. Vaidya, Document clustering: Tf-idf approach, in: 2016 Interna-

tional Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), IEEE, 2016, pp. 61–66.
[22] A. Patel, K. Meehan, Fake news detection on reddit utilising countvectorizer and term frequency-inverse document frequency with logistic regression, multinominalnb and support vector machine, in: 2021 32nd Irish Signals and Systems Conference (ISSC), IEEE, 2021, pp. 1–6.
[23] E. G. Grigoryeva, V. A. Klyachin, Y. V. Pomelnikov, V. V. Popov, Algorithm of key words search based on graph model of linguistic corpus, Vestnik Volgogradskogo Gosudarstvennogo Universiteta. Seriia 2, IAzykoznanie 16 (2017) 58.
[24] K. Lau, Q. Wu, Online training of support vector classifier, Pattern Recognition 36 (2003) 1913–1920.
[25] W. Zhao, L. Zhu, M. Wang, X. Zhang, J. Zhang, Wtl-cnn: A news text classification method of convolutional neural network based on weighted word embedding, Connection Science 34 (2022) 2291–2312.
[26] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, 2020. 1911.02116.
[27] xlm-roberta-large · hugging face, 2023. URL: https://huggingface.co/xlm-roberta-large.
[28] N. Goyal, J. Du, M. Ott, G. Anantharaman, A. Conneau, Larger-Scale transformers for multilingual masked language modeling, 2021. 2105.00572.
[29] D. Rothermel, M. Li, T. Rocktäschel, J. Foerster, Don't sweep your learning rate under the rug: A closer look at cross-modal transfer of pretrained transformers, 2021. arXiv:2107.12460.

A. Online Resources

The repository containing the code used in this work is available via GitHub (https://github.com/MIBbrandon/PLN_PoliticES.git) and the models used for the transformers approach can be found via HuggingFace (https://huggingface.co/models).

B. Extra figures

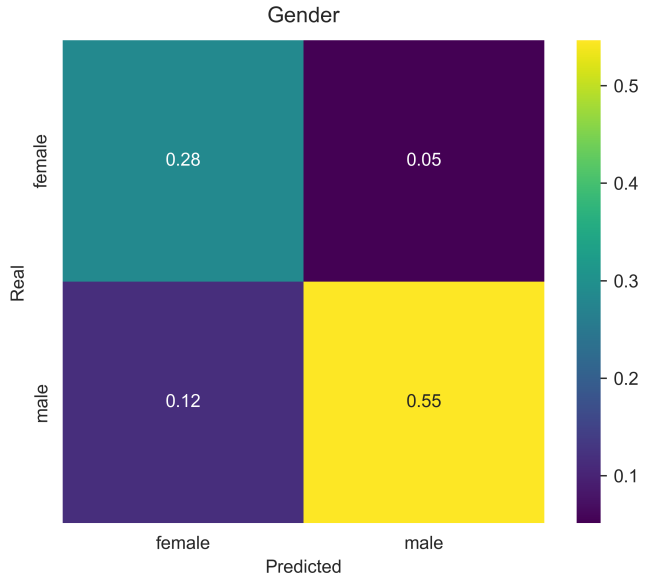

Figure 11: Transformer Confusion Matrix for Gender on Our Dataset

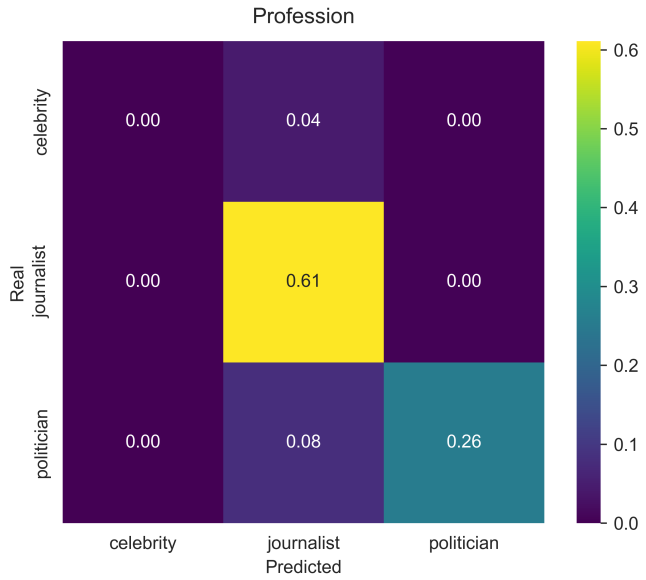

Figure 12: Transformer Confusion Matrix for Profession on Our Dataset

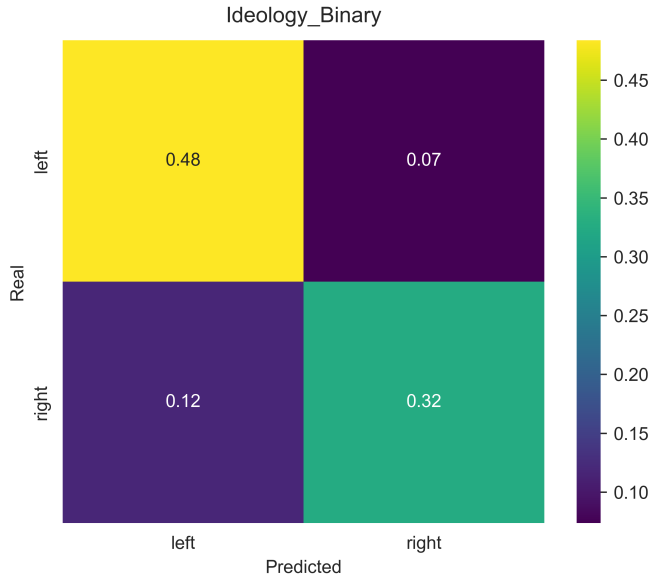

Figure 13: Transformer Confusion Matrix for Ideology (Binary) on Our Dataset

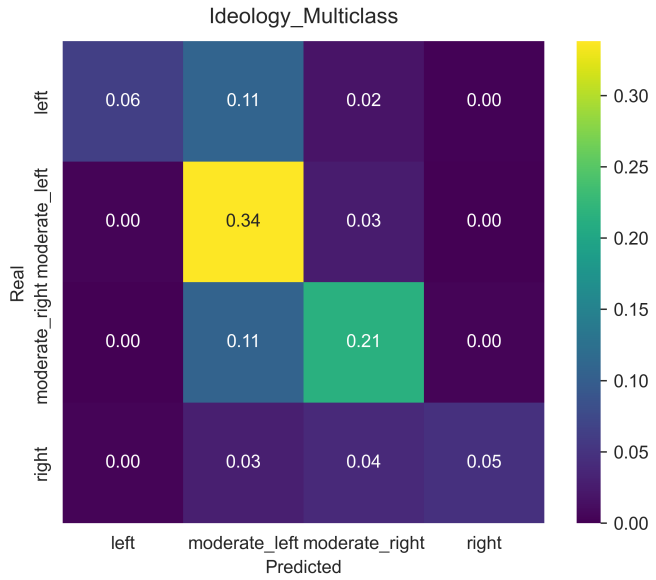

Figure 14: Transformer Confusion Matrix for Ideology (Multiclass) on Our Dataset

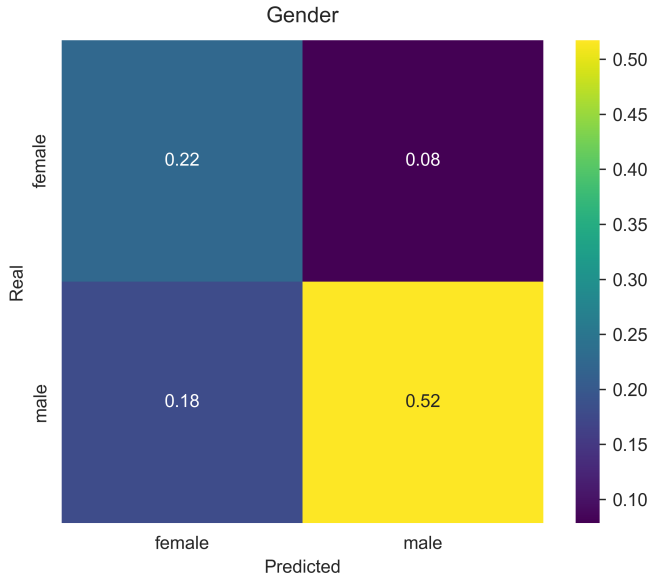

Figure 15: Transformer Confusion Matrix for Gender on the Provided Dataset

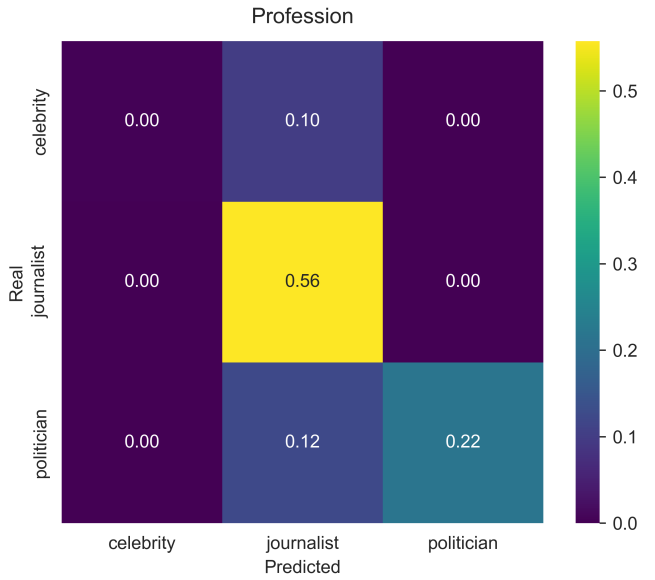

Figure 16: Transformer Confusion Matrix for Profession on the Provided Dataset

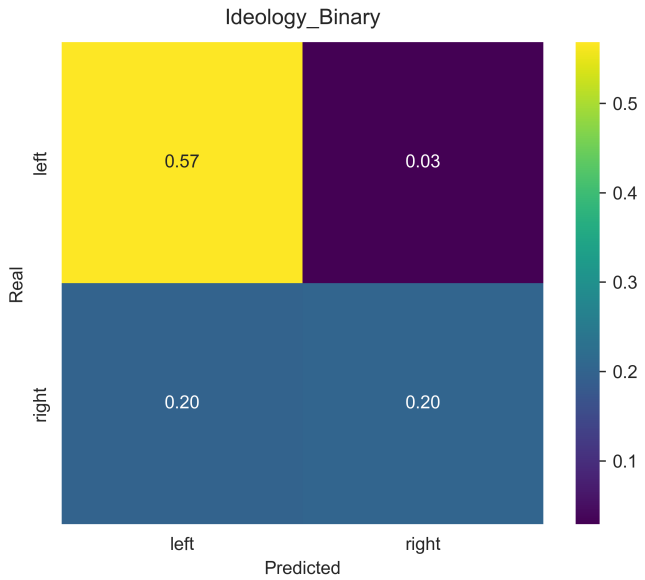

Figure 17: Transformer Confusion Matrix for Ideology (Binary) on the Provided Dataset