GIL_UNAM_Iztacala at Touché: Benchmarking Classical Models for Multilingual Political Stance and Power Classification*

Notebook for the GIL_UNAM_Iztacala Lab at CLEF 2025

Luis A. H. Miranda[2], Jesús Vázquez-Osorio[3], Adrián Juárez-Pérez[1], Gerardo Sierra[1] and Gemma Bel-Enguix[1]

[1]*Grupo de Ingeniería Lingüística - UNAM, Instituto de Ingeniería, Circuito Escolar -, 04510 Mexico City, Mexico*

[2]*Universidad Nacional Autónoma de México, Facultad de Estudios Superiores Iztacala, Avenida de los Barrios 1, 54090 Estado de México, Mexico*

[3]*Posgrado en Ciencia e Ingeniería de la Computación - UNAM,Circuito Escolar, 04510 Mexico City, Mexico*

Abstract

In this article, we present a methodology developed to address the challenges of the Touché shared task on *Ideology and Power Identification in Parliamentary Debates*, which consists of three sub-tasks: determining the ideological orientation of a speaker's party, identifying whether the party is in government or opposition, and classifying the party's stance on the populist–pluralist spectrum. To tackle these tasks, we implemented a comprehensive pipeline to train, evaluate, and compare several classical machine learning models, including Bernoulli Naive Bayes, Logistic Regression, Support Vector Machines, and Random Forest. Our results show strong and consistent performance in Sub-tasks 1 and 2 across multiple languages, with macro F1-scores indicating reliable generalization. However, Sub-task 3 presented greater challenges, with lower and more variable performance, suggesting the increased complexity involved in modeling populism in multilingual parliamentary discourse.

Keywords

Political discourse, Machine learning, Multilingual text classification, Political NLP

1. Introduction

Parliamentary debates are a rich source of political expression, offering insight into the ideological leanings and governing positions of elected representatives. As both parliaments and citizens engage in discussions on critical issues, language becomes a key medium through which political positions are conveyed, often reflecting the broader semantic and cultural context of a region or country. In this context, the analysis of multilingual parliamentarian speeches represents significant challenges due to structural and semantic differences between cultural edges. The way in which power, disagreement, or support for the government is expressed varies greatly between political cultures. Even the same linguistic pattern can have different meanings depending on the country or historical context. Understanding these nuances is essential for building robust models that generalize across languages and political systems. To address these challenges, we adopt a hybrid evaluation framework that explores combinations of classical text representations (TF-IDF and count-based) with multiple machine learning models. This approach allows us to systematically assess which configurations generalize best in multilingual political discourse.

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain

*Corresponding author.

†These authors contributed equally.

luisheml16@comunidad.unam.mx (L. A. H. Miranda); jesusvo5599@comunidad.unam.mx (J. Vázquez-Osorio); danyjuarez99@ciencias.unam.mx (A. Juárez-Pérez); GSierraM@iingen.unam.mx (G. Sierra); gbele@iingen.unam.mx (G. Bel-Enguix)

2. Background

Identifying populism is a tiny line, and it is also related to strong ideologies such as socialism (left wing) and nationalism (right wing). This may be crucial to understanding the difference between some types of populism (Trump vs. Evo Morales). Both adversaries are classified as populist, but its main reference field is completely opposite. Meanwhile, Trump is associated with a strong right ideology, Morales is related with the left wing [**?**]. As any president, any citizen can be represented as a part of an ideology that completely defines his behaviors [**?**]. This distinction matters because each ideology frames the world differently, shaping who is seen as the problem and who deserves protection.

Kitchener et. al. [**?**] says that political or ideological behaviors are reflected in a physical interaction with the society or even on the Internet. Every interaction on the internet left a digital footprint that can be traced, and Taulli [1] identified these massive amounts of data as Big Data. Also, for some companies or political parties, it is important to understand how people interact and say on social media.

Recent years have witnessed the development of several models addressing the problem of political ideology identification. Notably, Iyyer et al. [2] applied a Recursive Neural Network (RNN) to this task using a sentence-based approach. Their work utilized the Ideological Book Corpus (IBC), a dataset composed of annotated U.S. Congressional debates, labeled for ideological bias (Republican or Democrat) at both the sentence and phrase levels. Their model outperformed traditional methods, such as bag-of-words classifiers, particularly when applied to sentence-level annotated data, highlighting the advantages of leveraging syntactic structures in ideological classification.

More recently, Andruszak [3] contributed to the 2023 Power Identification shared task by implementing four distinct approaches. The first two relied on Large Language Models (LLMs), specifically fine-tuning BERT and LLaMA 3, as well as employing prompt engineering techniques with LLaMA 3. The third approach utilized a statistical method based on a Z-score summation, which combined the mean and standard deviation of token-level representations within a given text. The fourth involved training a Support Vector Machine (SVM) classifier. While none of the methods significantly outperformed the others, the results suggested that performance could be enhanced through the integration of rule-based components tailored to each parliamentary context within a structured pipeline.

This interest in political ideology is not merely academic; ideological orientation influences voting behavior, social media engagement, and policy support. As political expression increasingly takes place in digital environments, understanding how ideological cues manifest in text becomes essential for both political science and computational modeling.

3. System Overview

3.1. Data Overview

This work was carried out as part of the Touché Lab at CLEF 2025 specifically for the task *Ideology and Power Identification in Parliamentary Debates 2025* [4]. This task consists of three subtasks: 1) identify the ideology of the speaker's party, 2) identify whether the speaker's party is currently governing or in opposition, and 3) identify the position of the speaker's party in populist - pluralist scale.

The data set consists of 29 languages that represent an European language. This multilingual data set is provided by Fröbe et. al. [5]. This set consists of:

- **id:** A unique identifier for each individual speech instance.
- **speaker:** A identifier for a unique person. There may be multiple speeches from the same speaker.
- **sex:** The biological sex of the speaker. It can be classified as Female, Male or Unknown sex.
- **text:** The original speech text in the speaker's native European language.
- **text_en:** The English translation of the political speech.
- **orientation:** A binary label indicating the speaker's ideological stance (**0**: left and **1**: right).
- **power:** A binary label reflecting the speaker's political role (**0**: opposition, **1**: coalition, or the governing party).

- **populism:** An ordinal variable capturing the degree of populism in the speaker's party, on a four-point scale (**1**: Strongly Pluralist, **2**: Moderately Pluralist **3**: Moderately Populist, **4**: Strongly Populist).

To ensure a reliable modeling process, it is necessary to explore the class distribution across all the sub-tasks, as class imbalance can significantly affect the performance and generalizability of classification models. Figure 1 shows the class distribution for Sub-task 1 (Ideological Classification). The bars represent the percentage of each class within a given language. Class 0 denotes left-wing ideology (gold), while Class 1 denotes right-wing ideology (blue). Visual inspection reveals a pronounced imbalance, with right-wing parties being more frequently represented than left-wing ones. A chi-square goodness-of-fit test confirms that this imbalance is statistically significant, $\chi^2(1) = 5344.14$, $p < .001$. This suggests that the training data may introduce bias into models trained for this sub-task, particularly favoring the dominant class.

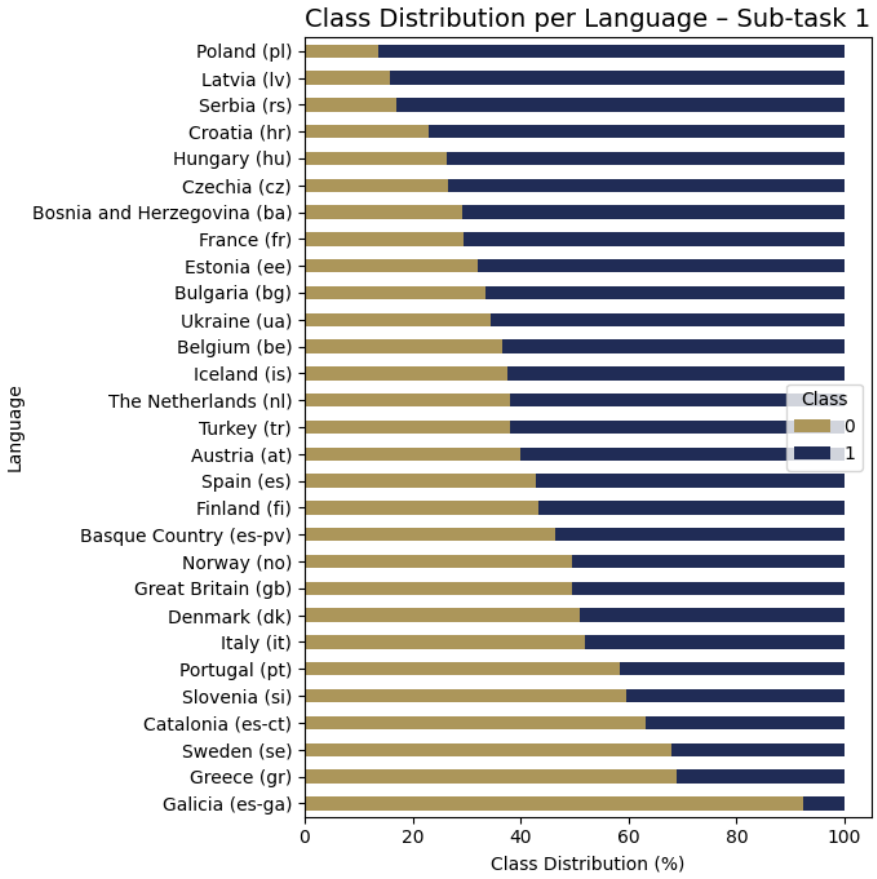

Figure 1: Class Distribution by Language — Sub-Task 1

Figure 2 illustrates the distribution of power roles across languages (sub-task 2 - Governing vs Opposition). Fewer languages are represented in Sub-task 2, likely due to missing information on whether parties were in government or opposition. The golden bars represent parties in opposition, while blue bars denote coalition or governing parties. For instance, Serbia and Croatia exhibit a strong dominance of coalition-class samples, whereas Spain and the Basque Country show the opposite pattern. A chi-square goodness-of-fit test indicates that this imbalance is statistically significant, $\chi^2(1) = 558.45$, $p < .001$. This result indicates that the training data may introduce bias into models developed for this sub-task, potentially favoring the majority class.

Figure 3 displays the distribution for party positions along the populist-pluralist spectrum across languages (Sub-task 3 - Populism). The bars represent four classes: class 0 (Strongly Pluralist), class 1 (Moderately Pluralist), class 2 (Moderately Populist), and class 3 (Strongly Populist). Most languages

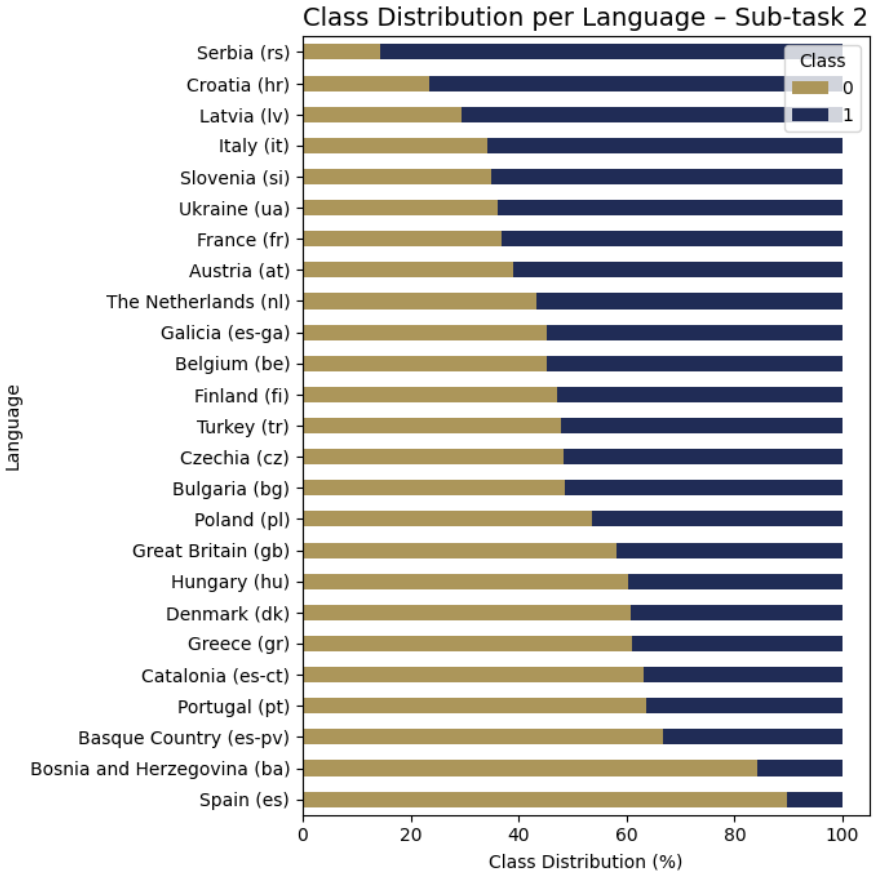

Figure 2: Class Distribution by Language Sub-Task 2

show a great variability and also a notably under representation for some labels. A chi-square goodness-of-fit test confirms that this imbalance is statistically significant, $\chi^2(3) = 11,465.05, p < .001$. This suggests that models trained on this data may be biased toward the more frequent categories, potentially under performing for less represented ideological positions. These insights from the exploratory analysis inform the modeling strategy presented in the next section, with a specific focus on mitigating class imbalance and ensuring fair performance across all categories.

3.2. Proposed Model

For a binary classification like Sub-task 1 (ideology classification) and Sub-task 2 (government vs. opposition classification), a systematic experimentation framework was implemented to compare different machine learning algorithms. Four commonly used machine learning classifiers for experimentation: Bernoulli Naive Bayes, Logistic Regression, Support Vector Machines (SVM), and Random Forest. Each model was tested in combination with two vectorization techniques — `CountVectorizer` and `TfidfVectorizer` — and across two n-grams: unigrams (1,1) and bigrams (1,2). Furthermore, a preprocessing option involving lowercasing and punctuation removal was toggled on and off to evaluate its impact on model performance. In total, the grid of experiments included:

- Classifier (4 options)
- Vectorizer type (2 options)
- Preprocessing (lowercasing and punctuation removal: on/off)
- N-gram (2 options)

To ensure reproductibility and robust comparison, each configuration was trained using `GridSearchCV` for each model. The macro-averaged F1-score was used as the main evaluation performance metric

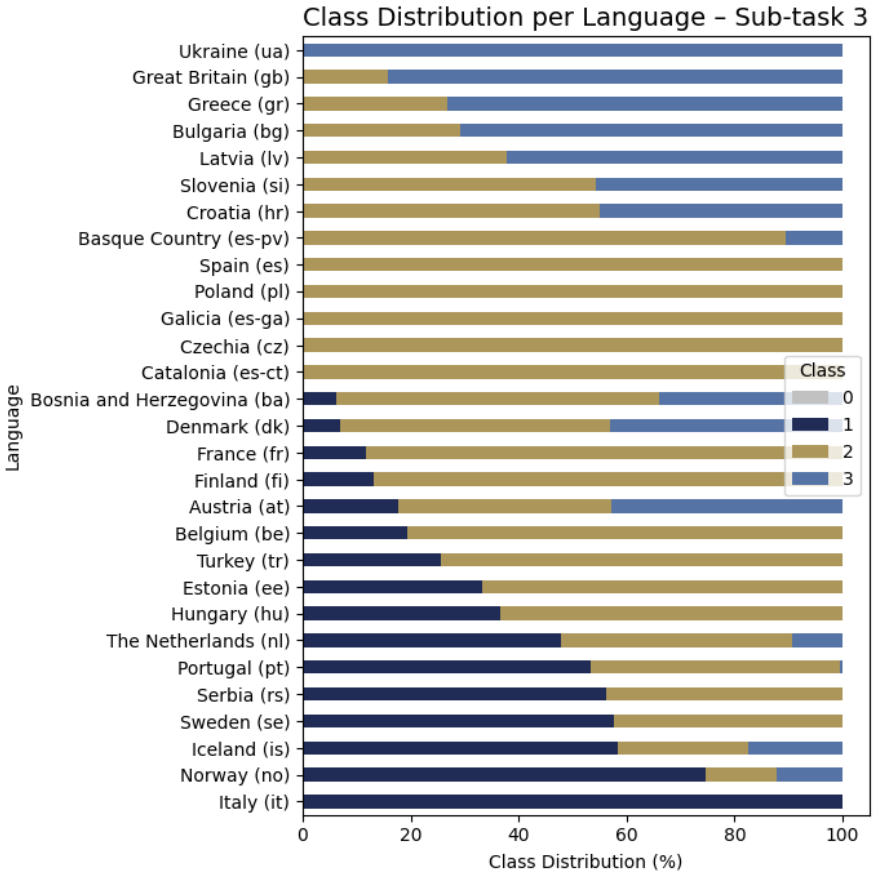

Figure 3: Class Distribution by Language Sub-Task 3

due to the class imbalance and the need to treat all classes equally. The training set was divided using stratified sampling to preserve the distribution of ideological labels. Moreover, for Sub-task 3 (populism classification), the same procedure was extended to a multi-labeled classification, where the labels were treated as ordered categories and the macro F1-score was again selected to reflect the performance across all levels of the populism scale.

3.3. Methodology

This section describes the purpose of each component used in the modeling pipeline.

3.3.1. Vectorization Techniques

- CountVectorizer transforms a corpus of text into a matrix of token counts. This class has a number of parameters that can also assist in text preprocessing tasks, such as stop word removal, word count thresholds (i.e. maximums and minimums), vocab limits, n-gram creation and more.
- TfidfVectorizer gives more weight to words that are more important and distinctive in a document. In addition, it takes away weight from words that do not help distinguish one class from another.

3.3.2. N-gram Range

- It is a sequence of consecutive n-elements arranged in a text. These elements can be traced as individual words or unigrams (1,1) or a pair of words, most often called bigrams (1,2). It helps de model identify a landscape of ideological expressions by the way the text is stored.

3.3.3. Text Preprocessing

- This process includes lowercase and the removal of special characters and numerical digits. This guarantees removing unnecessary noise elements so that the algorithms focus only on relevant language patterns.

3.3.4. Classification Algorithms

- **Bernoulli Naive Bayes**is a predictive model classification based on the Bayes theorem. The presence of one variable is not related to the probability of another variable occurring. But as we gain more data, we can associate the presence of that variable (independently of the others) with the classification characteristic.
- **Logistic Regression** is used to predict the probability of an event occurring. It involves a binary classification, with the output being a likelihood between 0 and 1.
- **Support Vector Machines (SVM)** aims to find the optimal line that separates two classes in a hyperplane with maximum margin.
- **Random Forest** is a predictive model arranged by binary rules (Yes/No). It is robust to noise and nonlinear patterns, but may overfit if not properly tuned.

Each algorithm was evaluated in various configurations to identify the most stable and high-performing approach per language. The decision to train models separately for each language rather than on a combined dataset was informed by initial tests.

3.3.5. Metrics Performance

- Precision measures the proportion of positive predictions that were correct.
- Recall evaluates the model's ability to detect all true positive cases.
- Accuracy reflects the total percentage of correct predictions the model made, both positive and negative.
- Macro F1-score combines precision and recall into a single measure, using their harmonic average, but does so for each class individually and then averages the results without weighting by class size. This helps mitigate class imbalance by giving equal importance to all classes regardless of their frequency.

3.3.6. Implementation Network and Tools

All experiments were conducted using Python and the *scikit-learn* library, which supported model selection, text vectorization, classifier implementation (e.g., MultinomialNB, LogisticRegression, RandomForest, SVC), and performance evaluation through metrics such as accuracy, precision, and F1-score. The use of *Pipeline* ensured reproducibility and modular design across experiments.

4. Results

The performance of the metrics for Sub-task 1 (Ideology Classification) is depicted in Figure 4. Panel A shows a violin element above the baseline (0.5) and a median value of 0.76. A descriptive analysis of the Macro F1-scores was conducted to understand the overall performance of the data distribution. The variance score was 0.0079, the standard deviation was 0.089. A Shapiro-Wilk normality test did not reflected a normal distribution for Macro F1-scores across languages ($p < 0.05$). Based on this, a bootstrap-based one-sample t-test (10,000 iterations) was conducted to assess whether the average F1-score exceeds the chance performance (0.5). The result was highly significant, $t(24) = 88.70$, $p < .001$, indicating that the model performs substantially better than random guessing in the Ideology Classification task.

The second panel displays all the evaluations metrics per language. The interquartile range spans from 0.7 (Q1) and 0.83 (Q3), indicating a strong general performance. Languages such as Catalonia (es-ct), Galicia (es-ga), and the Basque Country (es-pv) consistently surpassed the third quartile in all metrics. In contrast, languages like Bosnia and Herzegovina, Belgium, and Croatia consistently scored below the first quartile across metrics, which may point to language-specific challenges in the classification process or imbalanced data representation.

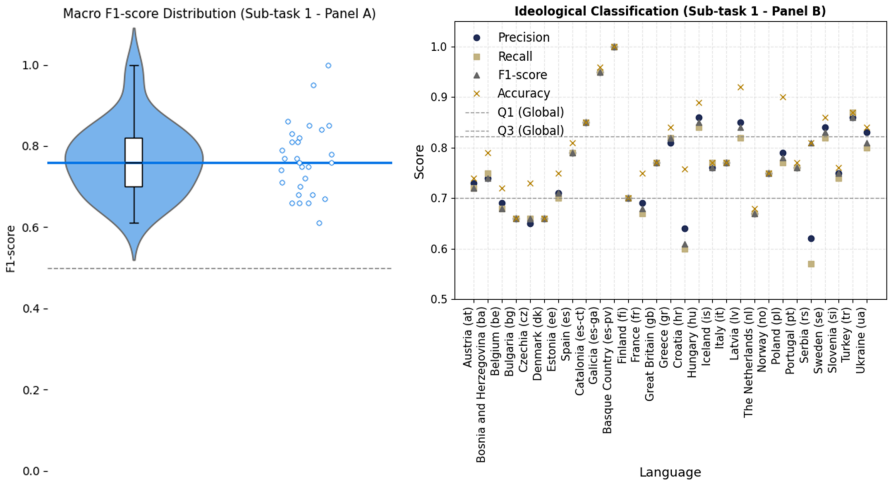

Figure 4: Ideological Classification (Sub-Task 1)

Figure 5 shows performance metrics for Sub-task 2 (Goverment vs Opposition) broken down by language. Panel A represents a raincloud plot where the Macro F1-score is represented for all languages. It is plausible to observe that the typical value of Macro F1-score is 0.76 (median value), and a great part of the language are above this value. The variance scores are 0.007 and the standard deviation is .08, which represents a slight variation in the data. A normal test was implemented, and results dit not find a normal distribution across data ($p < 0.05$). A bootstrap-based one-sample t-test was performed to assess whether the average Macro F1-score for Sub-task 1 exceeded a baseline of 0.5. The results were statistically significant, $t(24) = 84.50, p < .001$, indicating that the model performed significantly better than random guessing in the Goverment vs Opposition task.

For panel B, most languages are found to perform highly. The interquartile range has values between 0.72 and 0.83. Languages like Catalonia, Galicia, Basque Country, Greece, Hungary, and Turkey have all their metrics consistently above the upper quartile, suggesting robust and balanced performance in precision, recall, accuracy, and F1-score. In contrast, there are important variations between languages; Belgium and Croatia exhibit consistently lower performance, with all metrics falling below the first quartile.

The Sub-task 3 (Populism Scale Classification, Figure 3) involved a multi-class classification problem, it aimed to classify according to the degree of populism exhibited by the speaker's party. Panel A exhibits the performance of the results for the Macro F1-score in all languages. The violin plot shows a concentration of values in the higher performance range; however, it also displays a non-negligible spread below the baseline (0.5), indicating that for some cases, the model's predictions may approximate chance level. Panel A exhibits the performance of the Macro F1-scores across languages, with a median value of 0.66. Despite the concentration of values in the upper range, the distribution shows a notable spread, as reflected in the variance (0.0220) and standard deviation (0.148). A Shapiro-Wilk normality test indicated no significant deviation from normality ($W = 0.9600$, $p = 0.3285$), supporting the validity of subsequent parametric analysis. A bootstrapping t-test was calculated to identify if the average Macro F1-score of the model is significantly higher. The results were significant, $t(24) = 38.25$, $p<.001$, confirming that the model significantly outperforms random guessing.

Panel B shows the second panel where languages are represented by their evaluation metrics. Out-

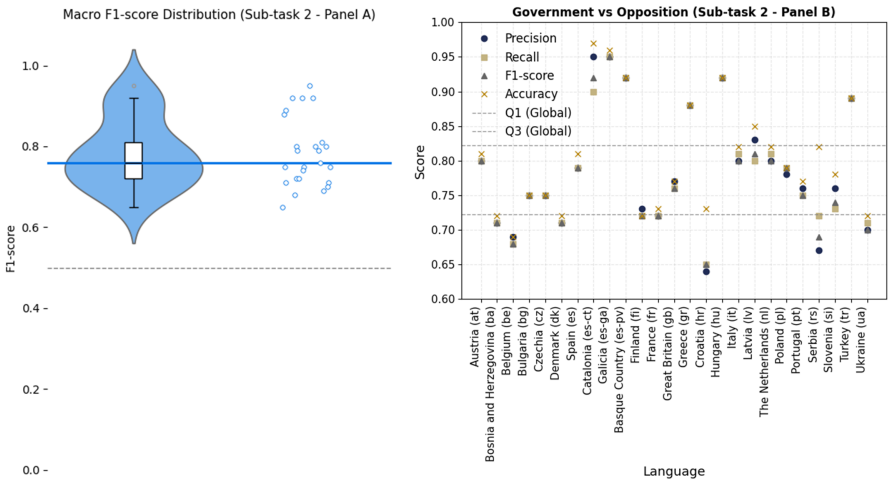

Figure 5: Government vs Opposition (Sub-task 2)

comes exhibit a greater variability among languages than the previous sub-tasks. Some languages drop below first quartile in all their metrics such as: Estonia, Finland, France, Great Britain, The Netherlands, and Norway, indicating that this task is more challenging for the model. Despite this, there are languages that perform outstandingly, with values greater than the third quartile, such as: Catalonia, Galicia, the Basque Country and Greece, even with metrics close to or equal to 1.0.

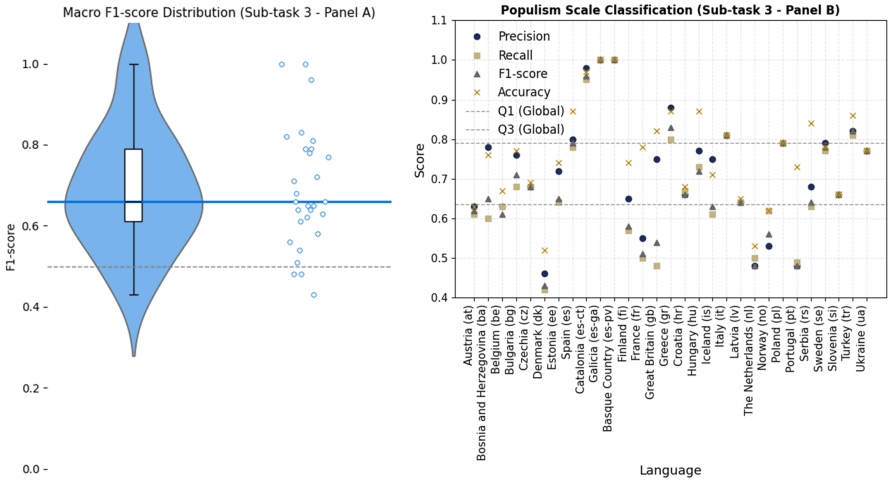

Figure 6: Populism Scale Classification (Sub-task 3)

The results for hyperparameter tuning are shown in Table 1 for sub-task 1. Support Vector Machine (SVM) combined with TF_IDF and either unigrams or bigrams was the most robust hyperparameter tuning across languages. Second, Logist Regression performed best with regional languages (e.g., Catalan, Galician, Slovenian) when using preprocessing and bigrams. For all vectorizers in all sub-taks, min_df=2 and max_features=10,000 were set to control vocabulary size and reduce noise.

The results for hyperparameter tuning for sub-task 2 are shown in Table 2. Support Vector Machine (SVM) combined with TF-IDF and either unigrams or bigrams consistently outperformed other classifiers in 15 of the 25 languages such as Austria, Czechia, Denmark, Spain, France *et cetera.* Second most used algorithm was Logistic Regression for regional languages such as Turkish, Slovenian and Galician.

In sub-task 3 (populism detection), Support Vector Machine (SVM) with TI-IDF was again the most consistent configuration, used in over all of the languages. Logistic Regression remained competitive and Naive Bayes combined with CountVectorizer was effective for some languages. In contrast with

Table 1
Hyperpatameters for Sub-task 1

Languages	Model	Vectorizer	lowercase/punctuation_removal	ngram_range
Austria (at)	svm	tfidf	False	(1, 1)
Bosnia and Herzegovina (ba)	svm	tfidf	False	(1, 2)
Belgium (be)	logistic_regression	tfidf	True	(1, 1)
Bulgaria (bg)	logistic_regression	tfidf	True	(1, 2)
Czechia (cz)	svm	tfidf	False	(1, 1)
Denmark (dk)	logistic_regression	tfidf	True	(1, 2)
Estonia (ee)	svm	tfidf	True	(1, 2)
Spain (es)	svm	tfidf	False	(1, 2)
Catalonia (es-ct)	logistic_regression	tfidf	True	(1, 2)
Galicia (es-ga)	logistic_regression	tfidf	True	(1, 1)
Basque Country (es-pv)	bernoulli_nb	count	True	(1, 1)
Finland (fi)	svm	tfidf	True	(1, 2)
France (fr)	bernoulli_nb	count	True	(1, 1)
Great Britain (gb)	logistic_regression	count	False	(1, 2)
Greece (gr)	logistic_regression	tfidf	True	(1, 1)
Croatia (hr)	logistic_regression	count	True	(1, 1)
Hungary (hu)	svm	tfidf	False	(1, 2)
Iceland (is)	svm	tfidf	True	(1, 2)
Italy (it)	svm	tfidf	True	(1, 1)
Latvia (lv)	svm	tfidf	True	(1, 2)
The Netherlands (nl)	svm	tfidf	True	(1, 1)
Norway (no)	random_forest	count	False	(1, 1)
Poland (pl)	svm	tfidf	True	(1, 2)
Portugal (pt)	svm	tfidf	True	(1, 1)
Serbia (rs)	bernoulli_nb	count	True	(1, 1)
Sweden (se)	logistic_regression	tfidf	True	(1, 1)
Slovenia (si)	logistic_regression	tfidf	True	(1, 2)
Turkey (tr)	svm	tfidf	True	(1, 2)
Ukraine (ua)	logistic_regression	tfidf	False	(1, 2)

other subtasks, preprocessing (lowercasing and punctuation removal) was less frequently applied, suggesting that case sensitivity and punctuation may contain stylistic signals relevant to populist expression. In contrast with other subtasks, preprocessing (lowercasing and punctuation removal) was less frequently applied, suggesting that case sensitivity and punctuation may contain stylistic signals relevant to populist expression.

5. Conclusion

In this work, a pipeline was designed to train four classical machine learning models to identify the best F1-score. This pipeline consisted of preprocessing and systematically vary vectorization methods, n-grams and the machine learning model. The optimal model and the corresponding parameter set were selected for each language on the basis of performance.

The pipeline demonstrated strong results across all three subtasks. For Sub-task 1, overall performance was strong, with most languages achieving F1-scores above the typical benchmark of 0.76. Similarly, in Sub-task 2, results were generally satisfactory; notably, languages such as Catalan, Galician, and Basque consistently exceeded the third quartile across all evaluation metrics. Sub-task 3 revealed that, for certain languages, model predictions were not significantly better than chance.

Statistical validation using bootstrap-based t-tests confirmed that the observed performance in all three sub-tasks was significantly above chance ($p < .001$). These findings suggest that even classical models, when systematically optimized, can deliver robust results in multilingual political classification

Table 2
Hyperpatameters for Sub-task 2

Languages	Model	Vectorizer	lowercase/punctuation_removal	ngram_range
Austria (at)	svm	tfidf	True	(1, 2)
Bosnia and Herzegovina (ba)	logistic_regression	count	False	(1, 1)
Belgium (be)	logistic_regression	tfidf	False	(1, 1)
Bulgaria (bg)	logistic_regression	tfidf	True	(1, 2)
Czechia (cz)	svm	tfidf	True	(1, 2)
Denmark (dk)	svm	tfidf	True	(1, 2)
Spain (es)	svm	tfidf	False	(1, 2)
Catalonia (es-ct)	svm	tfidf	False	(1, 2)
Galicia (es-ga)	logistic_regression	tfidf	True	(1, 1)
Basque Country (es-pv)	svm	tfidf	False	(1, 1)
Finland (fi)	logistic_regression	tfidf	True	(1, 2)
France (fr)	svm	tfidf	False	(1, 1)
Great Britain (gb)	svm	tfidf	False	(1, 2)
Greece (gr)	logistic_regression	tfidf	True	(1, 2)
Croatia (hr)	svm	tfidf	False	(1, 1)
Hungary (hu)	svm	tfidf	True	(1, 1)
Italy (it)	svm	tfidf	False	(1, 1)
Latvia (lv)	logistic_regression	tfidf	True	(1, 2)
The Netherlands (nl)	svm	tfidf	False	(1, 1)
Poland (pl)	svm	tfidf	False	(1, 2)
Portugal (pt)	logistic_regression	count	False	(1, 1)
Serbia (rs)	svm	tfidf	True	(1, 2)
Slovenia (si)	logistic_regression	tfidf	True	(1, 2)
Turkey (tr)	logistic_regression	tfidf	True	(1, 2)
Ukraine (ua)	svm	tfidf	True	(1, 1)

tasks.

The best-performing configuration across the majority of languages involved the use of TF-IDF vectorization, bigrams (n-gram range = (1,2)), and Support Vector Machines (SVM) with default hyperparameters. This combination consistently yielded higher macro F1-scores, particularly in Sub-task 1 and Sub-task 2. In contrast, for languages with smaller datasets or higher class imbalance, Logistic Regression with L2 regularization and CountVectorizer showed more stability, suggesting that performance may vary depending on language-specific characteristics such as vocabulary richness and class distribution.

Overall, the results are encouraging. We acknowledge that training classical models is considerably less time-consuming compared to fine-tuning large language models (LLMs), which provides us with the flexibility to conduct deeper analyses in the future for those languages with lower performance. Additionally, we remain open to exploring LLM-based approaches for these cases if needed.

Acknowledgments

This work was partially supported by UNAM PAPIIT project IG400725, and by the Mexican Government through SECIHTI Project FC-2023-G-64. The first author additionally acknowledges support from SECIHTI (CVU: 123456).

Declaration on Generative AI

During the preparation of this work, the author(s) used GPT-4 solely for grammar and spelling checks. After using this tool, the author(s) reviewed and edited the content as needed and take(s) full responsi-

Table 3
Hyperpatameters for Sub-task 3

Languages	Model	Vectorizer	lowercase/punctuation_removal	ngram_range
Austria (at)	logistic_regression	tfidf	False	(1, 2)
Bosnia and Herzegovina (ba)	svm	tfidf	True	(1, 2)
Belgium (be)	multinomial_nb	tfidf	False	(1, 1)
Bulgaria (bg)	logistic_regression	tfidf	False	(1, 1)
Czechia (cz)	logistic_regression	tfidf	True	(1, 2)
Denmark (dk)	svm	tfidf	True	(1, 1)
Estonia (ee)	logistic_regression	tfidf	False	(1, 1)
Spain (es)	svm	tfidf	False	(1, 2)
Catalonia (es-ct)	svm	tfidf	False	(1, 2)
Galicia (es-ga)	multinomial_nb	count	True	(1, 1)
Basque Country (es-pv)	multinomial_nb	count	True	(1, 1)
Finland (fi)	svm	tfidf	False	(1, 1)
France (fr)	svm	tfidf	False	(1, 2)
Great Britain (gb)	logistic_regression	count	True	(1, 1)
Greece (gr)	logistic_regression	tfidf	False	(1, 1)
Croatia (hr)	multinomial_nb	count	True	(1, 2)
Hungary (hu)	svm	tfidf	True	(1, 2)
Iceland (is)	svm	tfidf	False	(1, 2)
Italy (it)	random_forest	tfidf	True	(1, 2)
Latvia (lv)	svm	tfidf	False	(1, 2)
The Netherlands (nl)	svm	tfidf	True	(1, 1)
Norway (no)	svm	tfidf	False	(1, 1)
Poland (pl)	svm	tfidf	True	(1, 2)
Portugal (pt)	svm	tfidf	False	(1, 1)
Serbia (rs)	svm	tfidf	True	(1, 2)
Sweden (se)	svm	tfidf	False	(1, 1)
Slovenia (si)	logistic_regression	tfidf	False	(1, 2)
Turkey (tr)	svm	tfidf	False	(1, 2)
Ukraine (ua)	logistic_regression	tfidf	False	(1, 2)

bility for the publication's content.

References

[1] T. Taulli, Artificial Intelligence Basics: A Non-Technical Introduction, primera ed., Apress, Berkeley, CA, 2019. URL: https://link.springer.com/book/10.1007/978-1-4842-5028-0. doi:10.1007/978-1-4842-5028-0, publicado el 2 de agosto de 2019.

[2] M. Iyyer, P. Enns, J. Boyd-Graber, P. Resnik, Political ideology detection using recursive neural networks, in: K. Toutanova, H. Wu (Eds.), Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Baltimore, Maryland, 2014, pp. 1113–1122. URL: https://aclanthology.org/P14-1105/. doi:10.3115/v1/P14-1105.

[3] M. Andruszak, A. Alhamzeh, E. Egyed-Zsigmond, A. Carlsson, J. Leydet, Y. Otiefy, Team insa passau at touché: Multi-lingual parliamentary speech classification, in: Conference and Labs of the Evaluation Forum, 2024. URL: https://api.semanticscholar.org/CorpusID:271843942.

[4] T. Erjavec, M. Kopp, N. Ljubešić, T. Kuzman, P. Rayson, P. Osenova, M. Ogrodniczuk, Ç. Çöltekin, D. Koržinek, K. Meden, et al., Parlamint ii: advancing comparable parliamentary corpora across europe, Language Resources and Evaluation (2024) 1–32.

[5] M. Fröbe, M. Wiegmann, N. Kolyada, B. Grahm, T. Elstner, F. Loebe, M. Hagen, B. Stein, M. Potthast, Continuous integration for reproducible shared tasks with tira.io, in: J. Kamps, L. Goeuriot,

F. Crestani, M. Maistro, H. Joho, B. Davis, C. Gurrin, U. Kruschwitz, A. Caputo (Eds.), Advances in Information Retrieval. 45th European Conference on IR Research (ECIR 2023), Lecture Notes in Computer Science, Springer, Berlin Heidelberg New York, 2023, pp. 236–241. doi:10.1007/978-3-031-28241-6_20.