

ELiRF-VRAIN at PoliticES-IberLEF2023: Dealing with Long Texts in Transformer-based Systems for User Profiling

Vicent Ahuir, Lluís Felip Hurtado, Fernando García-Granada* and Emilio Sanchis

Valencian Research Institute for Artificial Intelligence (VRAIN), Universitat Politècnica de València, Camino de Vera s/n, 46022 Valencia. Spain

Abstract

In this paper, we present our approach to the PoliticES 2023 task. This shared task aims to extract user information from tweets in Spanish. We have developed systems based mainly on Deep Neural Networks (Transformers) to address the problem of binary and multiclass classification. Using pre-trained Transformer-based language models in this shared task poses an input length challenge because the amount of text per user significantly exceeds the input capabilities of common Transformer-based models. Our systems deal with input length problems by dividing the input into subsamples and performing the classification using a voting scheme. The results show the adequacy of our systems for the proposed task.

Keywords

Transformers, User Profiling, Voting Classification

1. Introduction

PoliticES 2023 is a shared task that aims to extract user information from tweets in Spanish. There is a growing interest in this type of analysis of user profiles and the correlation between some personality traits and political ideology, especially in the field of social networks. This task was initiated last year in IberLEF 2022, and was called PoliticES 2022 [1]; although a previous dataset was generated in 2020, the PoliCorpus 2020 dataset [2]. For this new edition of the shared task of 2023 [3], the participants will work with clusters of texts written by different users, but with the same traits. As the clusters can be considered a kind-of meta-users, the organization posed a user profiling challenge that consisted of political ideology identification (binary and multiclass classification), gender identification (binary), and profession identification (multiclass) of a set of texts that belong to users with the same traits. In political ideology, it is distinguished between left and right in binary classification, and, for multiclass classification,

IberLEF 2023, September 2023, Jaén, Spain

*Corresponding author.


✉ viahes@dsic.upv.es (V. Ahuir); lhurtado@dsic.upv.es (L. F. Hurtado); fgarcia@dsic.upv.es (F. García-Granada); esanchis@dsic.upv.es (E. Sanchis)

🌐 <https://vrain.upv.es/elirf/> (V. Ahuir); <https://vrain.upv.es/elirf/> (L. F. Hurtado); <https://vrain.upv.es/elirf/> (F. García-Granada); <https://vrain.upv.es/elirf/> (E. Sanchis)

🆔 0000-0001-5636-651X (V. Ahuir); 0000-0002-1877-0455 (L. F. Hurtado); 0000-0003-2213-4213 (F. García-Granada); 0000-0002-6737-4723 (E. Sanchis)



© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

left/right or a moderate stand for each wing. In gender, it is distinguished between men and women. Lastly, politicians, journalists, and celebrities are distinguished on the identification of the profession.

2. Dataset

The dataset[4] defined for this task is an extension of the PoliCorpus 2020 dataset [2], and the corpus used for the PoliticES 2022 shared task [1]. The data was collected from 2020, and 2022 from the Twitter accounts of politicians, political journalists, and celebrities in Spain using the UMUCorpusClassifier [5].

The politicians' accounts were selected from:

- (1) members of the government of Spain,
- (2) members of the Congress and Senate of Spain,
- (3) mayors of some important cities in Spain,
- (4) presidents of the autonomous communities,
- (5) former politicians, and
- (6) collaborators affiliated with political parties.

Journalists were selected from different Spanish news media, such as ABC, El País, ElDiario, El Mundo, or La Razón, among others.

The organizers created clusters of texts, mixing some of these extracted tweets. Each cluster groups 80 tweets written by different users that share all the traits under evaluation and are selected, favoring the diversity, including texts from different dates and topics. Every cluster is labeled with gender (male, female), profession (politician, journalist), and two types of political ideology: binary (left, right) and multiclass (left, moderate_left, moderate_right, right).

The dataset is composed of approximately 2800 different clusters. The training and test sets will be released (80%-20%), that is, 2250 clusters for training and 547 for testing. Cluster size approximately varies between 6000 and 21 600 words for the training set and 8100 and 21 600 words for the test set.

3. System architecture and Fine-tuning model process

In this work, we wanted to evaluate the capabilities of Transformers-based [6] systems for the task of user profiling. However, as mentioned in the previous section, each training sample is made up of the text of 80 tweets from users with the same traits. This involves much more text than pre-trained models based on Transformers can handle. To deal with this problem, it was decided to divide each training sample into a set of subsamples with a size that would fit in the input layer of the Spanish pre-trained models that have shown better performance, MarIA [7] and BETO [8] available at HuggingFace [9] public hub. This can lead to mislabeling of some training subsamples, but we thought the benefits of using the whole dataset outweigh the drawbacks. In the inference stage, from a test sample a set of subsamples are generated that are individually labeled by the system. The final decision is made using a simple voting scheme.

For each task, the most voted class is selected. Due to time limits, it has not been possible to test other more prominent voting techniques.

The decision of which pre-trained model to use and the strategy to generate the subsamples was made through a previous validation process.

For choosing the pre-trained model for each classification subtask, we fine-tuned MarIA and BETO pre-trained models for the different classification subtasks, and measured the performance of each classification model with the validation set. All the BETO-based models obtained slightly better results than MarIA-based ones. For this reason, it was decided to use BETO for all the ongoing experimentation. Regarding the strategy to split each sample, to generate each subsample, the largest number of complete tweets that do not exceed a previously determined maximum number of words are grouped sequentially. We tested the maximum word length in the range of 300 to 500. The best results were achieved with subsamples that did not exceed 450 words.

In most of the models trained for this shared task, a search process for the best hyperparameter configuration was carried out. To do this hyperparameter optimization, the optuna library was used. Table 1 summarizes the hyperparameters search space.

Table 1
Ranges of values considered in the hyperparameter search.

Hyperparameter	Range
Training epochs	15 - 30
Learning rate	2e-5 - 1e-4
Batch size	[4, 8, 16, 32]
Gradient accumulation	[2, 4, 8, 16, 32]
Weight decay	0.001 - 0.015
Learning rate scheduler	[constant, linear]
Hidden dropout	0.05 - 0.15
Attention dropout	0.08 - 0.15

4. Run configuration details

During the challenge, we published a total of 7 runs. Most of those runs followed the system architecture detailed in Section 3. However, *Run2* was created with a classic approach using a Linear Support Vector Machine (SVM), and the input text was vectorized as a Bag of Words of up to 4-grams weighted with TF-IDF. This run was developed using the library Scikit-learn [10] and can be seen as a baseline for classical machine learning approaches.

To perform the fine-tuning of the pre-trained models, a stratified split of the training corpus was made. 90% of the corpus was used to fine-tune the models and the remaining 10% was used for the selection of the best epoch and the best hyperparameter configurations. Due to the size of the corpus, the stratified split could only be done taking into account the gender, profession and ideology-binary labels. In addition, to have greater variability in the learned models, two different random partitions were performed using the same stratified strategy but two different random seeds. In this way, it was possible to learn different models with two different partitions.

- Run1: BETO models fine/tuned individually for each task with hyperparameter search only in multiclass tasks, profession and ideology-m, using the first random partition.
- Run4: BETO models fine-tuned individually for each task with hyperparameter search in all tasks using the second random partition.
- Run6: For each task, the best model from Run1 and Run4 is selected.

Table 2 shows the hyperparameters used for Run1 and Run4. Since only these two runs introduce different fine-tuned models, we chose only to list them for simplicity. The rest of the runs were based on selecting a set of models from Run1 and Run4.

Table 2

Hyperparameters for each run and classification task. The hyperparameters are the following: Seed (S), Training epochs (TE), Learning rate (LR), Batch size (BS), Gradient accumulation (GA), Weight decay (WD), Learning rate scheduler (LRS), Hidden dropout (HD), and Attention dropout (AD).

Run	Task	S	TE	LR	BS	GA	WD	LRS	HD	AD
Run1	f1_gender	33	26/30	5.000e-05	32	1	0.0	linear	0.100	0.100
	f1_profession	33	5/11	3.350e-05	4	2	1.193e-2	linear	0.075	0.090
	f1_ideology_b	33	19/30	5.000e-05	32	1	0.0	linear	0.100	0.100
	f1_ideology_m	33	18/26	4.998e-05	32	2	5.119e-3	constant	0.119	0.130
Run4	f1_gender	11	7/16	6.422e-05	16	4	1.230e-2	linear	0.092	0.100
	f1_profession	11	12/24	2.990e-5	16	32	2.607e-3	constant	0.110	0.138
	f1_ideology_b	11	15/20	2.717e-05	16	4	4.770e-3	constant	0.098	0.123
	f1_ideology_m	11	17/21	5.102e-05	32	2	7.351e-3	linear	0.136	0.120

Table 3 summarizes the performance of the three runs for each task at subsample level in terms of macro-f1, the average of the four values is also added in the last column. In bold, it can be seen the best result of each task in Run1 and Run4 that is used for Run6. Note that *Run1* and *Run4* are based on different models since the randomness in the random partition and in the fine-tuning process is based on different seeds. It also can be observed, that the hyperparameter search process appreciably improves the results; the binary tasks of Run1, where no hyperparameter search was made, present a considerably lower performance than the equivalent ones in Run4.

Table 3

Results on the validation set at subsample level in terms of macro-f1 for each task and the average of all four values.

	f1_gender	f1_profession	f1_ideology_b	f1_ideology_m	avg. macro_f1
Run1	94.48	97.39	96.83	96.83	96.38
Run4	96.51	98.43	98.19	96.33	97.37
Run6	96.51	98.43	98.19	96.83	97.49

The use of different models for the ideology-binary and ideology-multiclass tasks can produce discrepancies in the labeling. For instance, for the same sample, the binary classifier can label a sample with the *right* label while the multiclass classifier can label this sample as

left or *moderated_left*. To avoid this, we create an Ideology Discrepancy Correction (IDC) procedure, where the political wing (left or right) is synchronized between the binary and the multiclass labels. Thus, the binary classifier was prioritized on the ideology identification, and the multiclass label was overwritten by the binary one when there were discrepancies between them. For instance, if a sample is binary classified with the *right* label, and multiclass identified as *left*, IDC procedure changes the multiclass label to *moderated_right*. Three additional runs were obtained by applying the IDC procedure to the previous runs. By doing it this way, we can quantify the impact of the IDC on the system’s performance. The additional runs were:

- Run3: Applying the IDC procedure to the output of Run1.
- Run5: Applying the IDC procedure to the output of Run4.
- Run7: Applying the IDC procedure to the output of Run6.

5. Experiments and Results

Table 4 shows the results obtained in the seven runs we published in this challenge. The first noticeable detail on the numbers is the significant reduction of the performance compared with the results obtained in the validation set (Table 3). This could indicate that the validation sets do not represent the whole nature of the samples in test dataset.

By analyzing the results in Table 4, we notice that the Transformer-based runs outperform in every classification task the SVM-based system baseline. If we compare *Run1* and *Run4*, we observe the critical aspect of hyperparameter search in the fine-tuning process. Performing the hyperparameter search in all the tasks (*Run4*) instead of just on multiclass tasks (*Run1*) increased the average performance of the system by 2.7%. Additionally, combining the best models in *Run6* increased the performance by 1.7% (if we compare the result with *Run4*). Finally, due to the higher performance of the ideology binary classifier compared to the multiclass ideology classifier, we could apply the IDC procedure in our final run and increased the final system (*Run7*) performance by 1.1%.

Table 4

Results of the seven runs. [T] means Transformers-based. [T+IDC] means Transformers-based and IDC procedure applied. [L-SVM] system based on Linear Support Vector Machines. In *Run7*, values in parentheses indicate the team’s global position in the challenge. “Best results” row shows the best result achieved by any participant in the competition.

	f1_gender	f1_profession	f1_ideology_b	f1_ideology_m	avg. macro_f1
Run1 [T]	79.23	75.93	89.31	64.94	77.35
Run2 [L-SVM]	75.40	70.55	86.09	63.56	73.90
Run3 [T+IDC]	79.23	75.93	89.31	68.41	78.22
Run4 [T]	82.96	82.76	89.67	62.46	79.47
Run5 [T+IDC]	82.96	82.76	89.67	63.09	79.62
Run6 [T]	82.96	82.76	89.67	64.94	80.08
Run7 [T+IDC]	82.96 (1)	82.76 (3)	89.67 (1)	69.131 (2)	81.13 (1)
Best results	82.96	86.08	89.67	69.133	81.13

Overall, our solution achieved the best performance of the challenge in two of the four classification tasks and in the average system performance. Interestingly, our approach had the best results in the binary classification tasks (Genre and Ideology) but achieved second or third place in multiclass tasks (Profession and Ideology). In the Ideology task, our best run achieved nearly the same score as the best run. However, the differences between our run and the best run were more significant in the Profession task. Only by the results, we can not conclude whether our approach has some limitations in multiclass classification tasks posed by this challenge or whether it is just circumstantial.

6. Conclusions

In this paper, we have presented a user profiling approach to infer gender, profession, and political ideology from tweets written in Spanish. Our system contains Spanish Transformer-based models that were fine-tuned for each classification subtask. The fine-tuning process was done with a hyperparameter search, which helped to increase the overall system performance. A difficulty posed by the challenge was the overall text length of the user's text when Transformers-based models are used due to the input length limitations of these models. We addressed these limitations by splitting the input into subsamples, classifying them, and performing a voting process to determine the final label for each sample. Also, we resolved discrepancies between the binary ideology classification and the ideology multiclass one, which further increased the system's performance. Our solution took first place in most classification tasks and reached the best overall performance in the challenge, indicating our approach's adequacy for the proposed tasks. In future work, we will explore some variations of the architecture, for example including adapter, and in the fine-tuning process. It is also interesting study other possibilities for tackling with the length of the input texts, for example, by modifying the voting mechanism.

Acknowledgments

This work is partially supported by MCIN/AEI/10.13039/501100011033, by the "European Union" and "NextGenerationEU/MRR", and by "ERDF A way of making Europe" under grants PDC2021-120846-C44 and PID2021-126061OB-C41. It is also partially supported by the Spanish Ministerio de Universidades under the grant FPU21/05288 for university teacher training.

References

- [1] 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 2022: Spanish Author Profiling for Political Ideology, *Procesamiento del Lenguaje Natural* 69 (2022) 265–272. doi:<https://doi.org/10.26342/2022-69-23>.
- [2] 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:<https://doi.org/10.1016/j.future.2021.12.011>.

- [3] 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.
- [4] 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).
- [5] J. A. García-Díaz, Almela Sánchez-Lafuente, G. Alcaraz Mármol, R. Valencia García, UMUCorpusClassifier: Compilation and evaluation of linguistic corpus for Natural Language Processing tasks, *Procesamiento del Lenguaje Natural* 65 (2020) 139–142. doi:<https://doi.org/10.26342/2020-65-22>.
- [6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, I. Polosukhin, Attention is all you need, in: I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, volume 30, Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.
- [7] A. G. Fandiño, J. A. Estapé, M. Pàmies, J. L. Palao, J. S. Ocampo, C. P. Carrino, C. A. Oller, C. R. Penagos, A. G. Agirre, M. Villegas, MarIA: Spanish Language Models, *Procesamiento del Lenguaje Natural* 68 (2022) 39–60. URL: <https://upcommons.upc.edu/handle/2117/367156#YyMTB4X9A-0.mendeley>. doi:10.26342/2022-68-3.
- [8] J. Cañete, G. Chaperon, R. Fuentes, J.-H. Ho, H. Kang, J. Pérez, Spanish pre-trained bert model and evaluation data, in: PML4DC at ICLR 2020, 2020.
- [9] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, A. M. Rush, Transformers: State-of-the-art natural language processing, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Association for Computational Linguistics, Online, 2020, pp. 38–45. URL: <https://www.aclweb.org/anthology/2020.emnlp-demos.6>.
- [10] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, *Journal of Machine Learning Research* 12 (2011) 2825–2830.