TaiDepZai999_UIT_AIC at IberLEF 2024 DETEST-Dis task: Classification of racial stereotypes in Spanish With Ensemble Learning Methods and BERT-based Adapter Head

Le Duc Tai*

[1]*University of Information Technology-VNUHCM, Quarter 6, Linh Trung Ward, Thu Duc District, Ho Chi Minh City, Vietnam*

Abstract

This paper presents our participation in the IBERLEF 2024 Task - DETESTS-Dis in Spanish, focusing on sub-task 1: determine whether a comment or sentence contains at least one stereotype or none, considering the full distribution of labels provided by the annotators. To address task 1 in the shared task, we implement and investigate different solutions, including (1) machine learning algorithms(SVM, RandomForest, LSTM, etc); and (2) Ensemble learning on the three best BERT-based models. All the experiments were conducted on the two T4 GPUs from the Kaggle platform. To enhance the context of the input review, we concat the contextual information to the language models like [Level1][Level2][Text input][Level3] "level1,2,3" are the names of columns in the dataset, and the contextual information is arranged in that order so it constructs a meaningful paragraph before feeding it to pre-trained language models to get the hidden state representation for the given input and we also use Adapter-Head with XLM-RoBERTa. The experimental results validate the effectiveness of our approach. Our best results on the test set are 63.0, 62.4, and 60.8 in terms of F1-score on the gold standard in percentage.

Keywords

Stereotype text classification, Spanish language, text classification, Ensemble learning, BERT-based models, Adapter Head,

1. Introduction

The shared-task IberLEF 2024 [1] Task Detection and classification of racial stereotypes in Spanish aims to classify the targeted stereotype related to specific groups(Women, immigrants, etc) or racism. In this shared task, two sub-tasks were proposed for participants. The first challenge, called the stereotype Identification task, aims to classify whether or not the targeted text contains stereotypes in the content of tweets or comments. For example, given a new tweet, “Las inmigrantes tienen más derechos que nosotras” the output of this task is “Stereotype” for the targeted class. On the other hand, another task is called Implicitness Identification, which aims to classify how the stereotype is expressed explicitly or implicitly only if the input contains a stereotype. Using the same input as above, the output of this task would be “Explicit”. The

IberLEF 2024, September 2024, Valladolid, Spain

*Corresponding author.

23521374@uit.edu.vn (L. D. Tai)

assigned classes in two sub-tasks 1 and 2 are “Stereotype“ - “NoStereotype“ and “Explicit“ - “Implicit“, respectively.

With the burst out of social media, the amount of posts, and comments have been dramatically escalated. Therefore, sensitive content verification has become a crucial problem on the internet. With the power of pre-trained BERT-based transformers from HuggingFace[2] (BETO [3], XLM-RoBERTa [4], mDeBERTa-v3 [5], etc), Adapters [6] and various ensemble methods, many natural language processing tasks have been successfully addressed as text classification. Therefore, this paper presents a multi-BERT-based model with an ensemble approach. Subsequently, we refine and optimize the pre-trained models by processing and extracting input with different methods to address sub-task 1 within an end-to-end framework effectively. Moreover, we design and test on three different ensemble learning methods namely, Hard-voting, Soft-voting, and Stacking.

2. Related work

Social media has taken over the internet which results in a large battery of posts, tweets, and comments within a day. Subsequently, this situation poses a pressing problem for social media companies like Facebook, Twitter, etc to control and ban toxic, racist, and sensitive content. While the concept of taking advantage of Artificial Intelligence to tackle such obstacles is not brand-new, Natural Language Processing has had a firm influence on this field.

In recent years, many researchers have brought innovation to Linguistic Features such as Named Entity Recognition and part-of-speech. However, little research was done to utilize categorical, numerical features besides text features. A study conducted by Poth et al. [6] demonstrated a novel way to improve model performance and processing time by adding several parameters in the head of model structure which since we only need to train an Adapter-Head [6] instead of a whole model, so it takes less time for training and achieved some promising results in text classification. More specifically, the head of the model i.e. Adapter-Head acts as an altered part that we can inject into models without modifying model code directly. Another popular approach for text classification is Ensemble Learning, Al-Omari et al. [7] has shown a significant improvement by constructing BERT(cased), BERT(uncased), BiLSTM, and XGBoost into ensemble structure that achieved high accuracy in propaganda detection. Another similar approach has been conducted by García-Díaz et al [8], they adopted soft, average voting and knowledge integration with four distinct feature sets that resulted in 69.9 f1-score in subtask 1 "Stereotype Classification" at DETEST 2022. Vázquez et al. [9] team also attained a 70.4 f1-score by applying back-translation and UnderBagging to tackle unbalanced data then they ensemble three models to attain such impressive result.

3. Approach

3.1. Traditional Machine Learning

First, we experimented in three sets as raw, clean, and proved context(1) ("level1,2,3" are the names of columns in the dataset) data shown below. Specifically, we have attempted to use

lemmatization, removing punctuation on data cleaning. However it failed to improve accuracy, so we cleaned the data by only removing emojis and URLs. Since the dataset is unbalanced, we also apply stratified K-fold cross-validation [10] with k = 10 to minimize the unbalanced data effect on models. Precisely, cross-validation helps reduce bias in performance estimation as the distribution of classes may not be equal in the random splitting dataset, so cross-validation mitigates this issue by repeatedly splitting data into different train and test sets. This helps in obtaining a more reliable estimate of model performance across different subsets of the data. Our simplest approach is extracting features with TF-IDF and then feeding it to SVM, Random Forest model in Scikit-Learn [11] and using grid-search to find the most optimal parameters. As seen in Table 1, Random Forest outperforms SVM on both raw and clean datasets, and clean data seem to increase F1-score to SVM as long as Random Forest.

$$[Level1][Level2][Text\,input][Level3] \tag{1}$$

Table 1
SVM and Random Forest result

Model	data	F1-score
SVM	Raw	74.0
Random Forest	Raw	77.5
SVM	Clean	75.5
Random Forest	Clean	78.5

3.2. LSTM

We made use of deep learning model by using LSTM with structure as shown in figure 1, we extract feature by utilize Word Embedding with pretrained Spanish Word2vec. We define Adam as optimizer, cross-entropy as loss function, and LSTM was trained in 50 epochs. Despite the fact that we have employed deep learning, our LSTM best result is 78.0 F1-score on raw dataset.

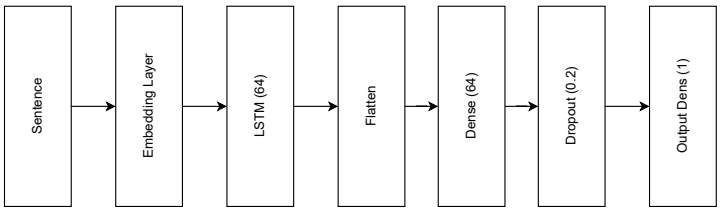

Figure 1: The architecture of LSTM model.

3.3. BERT-based models

BERT-based models have succeeded in many NLP tasks that why we took the BERT-based model as our main core and applied other methods to them. We have tested in six different BERT

models, namely BETO [3] (BERT model that has been trained on large Spanish corpus), RoBERTa (uncased), XLM-RoBERTa (uncased), mDeBERTa-v3 [5] (Multi-lingual DeBERTa). in addition to finetuning models, we also aim to use Transfer Learning with dehateBERT-mono-spanish [12], XLM-RoBERTa-twitter-hate which was trained on HaterNet dataset from SemEval-2019 [13] task 5, so it can reduce training time and computational resource. Before feeding data to our models, we tokenized data by using a pretrained tokenizer according to its model weights from HuggingFace Hub [2]. Given data was encoded into input-ids, attention-mask, and token-type-ids(if needed). Afterward, the feature vector was generated by BERT Word embedding from encoded data. HuggingFace Pipeline has handled inferences that return both hard labels and soft labels of each prediction. After finishing training and evaluating 6 models on raw, clean, and proved context datasets, we have observed that the three best models are BETO on the clean dataset and XLM-RoBERTa, XLM-RoBERTa-twitter-hate on the raw dataset. Unlike the traditional machine learning algorithm that we have present in 3.1, BERT-based models often seem to perform well on raw data instead of clean data.

Another holistic approach is using Adapter-Head [6], this is a training method for BERT-based models that not only can reduce the computational cost for the training process but also achieve notable accuracy. We decided to choose XLM-RoBERTa-twitter-hate to apply this method as XLM-RoBERTa-twitter-hate has achieved the most remarkable F1-score out of six models that I have tested. Initially, we add a pretrained adapter named xlm-roberta-base-es-wiki_pfeiffer [14] which is a base adapter for XLM-RoBERTa and has been pretrained on wiki Spanish corpus. Then, We ran inference on xlm-roberta-base-es-wiki_pfeiffer [14], and the result was promising. Therefore, we continued training that adapter from scratch with "Lora" configured on our dataset, this method has succeeded in obtaining even higher accuracy and less training time. Our final procedure is integrating our best models and strategies into only one robust method. We have analyzed and evaluated the ensemble learning approach but in three methods (Hard Voting, Soft Voting, and Stacking), and our ensemble architecture needs to have an odd number of models which is described in Figure 2. Accordingly, we promoted with three best models such as BETO, XLM-RoBERTa (uncased), and XLM-RoBERTa-twitter-hate (Adapter-based). Hard and soft Voting started by feeding data to three models and afterward utilizing three outputs from each model then unifying those to the final output by taking the mode of the three outputs (only in Hard Voting) or picking a class that has the highest probability out of all classes. Taking advantage of different models helps improve final prediction accuracy significantly.

In like manner, Stacking is a technique in which we utilize two levels of models i.e. level 0, and level 1 to produce a prediction. To be more precise, level 0 models are the three models that we have used in the voting technique above then a new training set was generated by taking each model's outputs. Level 1 (meta model) training juncture takes place after a new training set has been created, and the new training set is composed of 3 feature columns and 1 target column. Each feature columns are the predictions of the three-level 0 models. Additionally, we decided to choose Logistic Regression and Random Forest as our metamodel as they play a crucial role in generalizing and capturing the level 0 outputs rule instead of the original dataset. Nevertheless, we also apply cross-validation [10] to tackle Over-fitting because promoting the stacking technique could increase complexity. Consequently, we examined that stacking ensemble approach with Logistic Regression brought about more comparable advances in overall accuracy than leverage Random Forest as a classifier. Our Stacking architecture is depicted in

Figure 3 below:

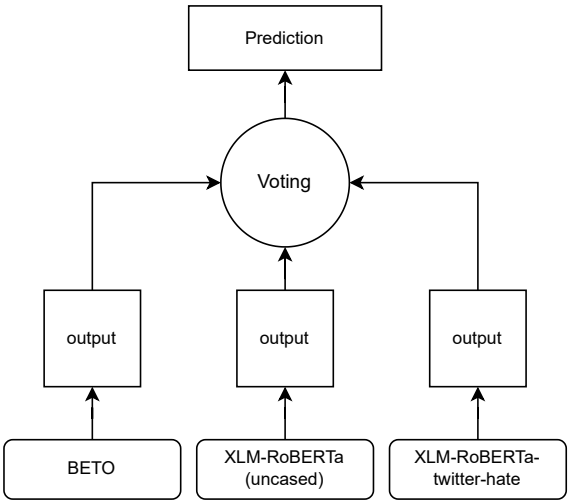

Figure 2: The structure of Hard and Soft Voting method.

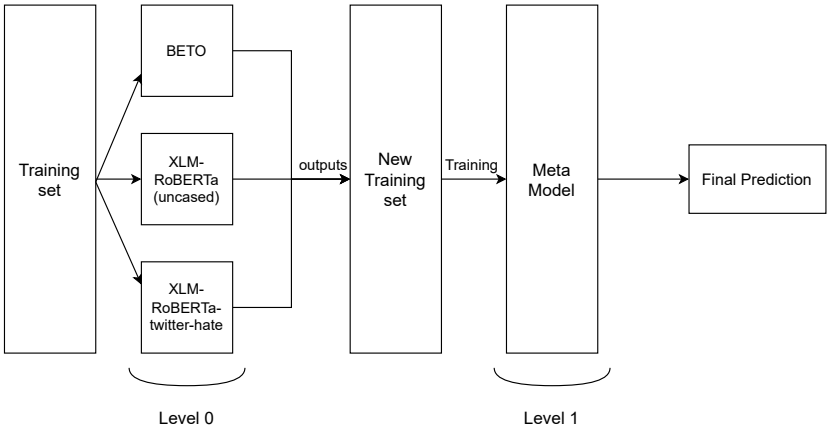

Figure 3: The structure of Stacking method.

4. Experimental Setup

4.1. Data and Evaluation Metrics

We only experimented on a dataset that was provided by the organizer for training models and testing approaches in this shared task. Table 2 illustrated the summary information about the

training and testing dataset. Meanwhile, Table 3 depicted general statistics and distribution among two classes in the training dataset. By observing class polarity in Table 3, we could see that the ratio between the "Stereotype" and "NoStereotype" number of labels is in the neighborhood of 1:3, which is an imbalanced polarity between the two classification classes. Consequently, imbalanced data was one of the main challenges that competitors must tackle while implementing distinct techniques to achieve the most optimal result possible. Furthermore, to face imbalanced class labels our team has taken advantage of cross-validation [10] which is one of the most popular ways to deal with such a problem. Cross-validation helps reduce bias in performance estimation as the distribution of classes may not be equal in the random splitting dataset, so cross-validation mitigates this issue by repeatedly splitting data into different train and test sets. This helps in obtaining a more reliable estimate of model performance across different subsets of the data. As shown in Table 2, in the training dataset the number of samples is much higher than in testing dataset records, which allows us to train and our models can easily generalize data.

Table 2
The general information of datasets from competition

Information	Training set	Testing set
Number of samples	9 906	2 205
Number of tokens	389 247	90 955
The average length	39.29	49.25
The maximum length	291	249

Table 3
The statistic of class distribution in training dataset

class samples	Stereotype	NoStereotype
Whole Dataset	2 605	7 301
Training set	2 070	5 855
Validation set	535	1 446

4.2. System Settings

We operate our training process with HuggingFace [2], and all of our BERT-based models were trained in 10 epochs. The AdamW optimizer was leveraged to optimize our models. We decided to choose 5e-5 as learning rate for BETO, RoBERTa (uncased), XLM-RoBERTa (uncased), mDeBERTa-v3 [5] (Multi-lingual DeBERTa) and 3e-5 for dehateBERT-mono-spanish [12], XLM-RoBERTa-twitter-hate. Additionally, the batch size was set to 16, 32 and the random seed is 221, and we set the max length token according to the longest token of the dataset. We have attempted to use lemmatization, removing punctuation on data cleaning. However it failed to improve accuracy, so we cleaned data by only removing emoji, and URLs and shortening multiple spaces. Our models were evaluated with an F1-score on both Scikit-Learn [11] and the Gold standard (which was given by the organizer). Regarding computational resources, Our

team made use of two GPU T4s which are provided for 30 hours free each week from Kaggle.

5. Main results

The official submission results are shown in Table 4, The Soft Voting technique registered the highest F1-score out of our three main approaches which are 0.6 and 2.2 higher in comparison to the Adapter and Stacking techniques, respectively. As we could observe, the ensemble learning method worked pretty well since it takes advantage of benefits and drawbacks from each model to strengthen the learning process and ability to classify stereotypes. Furthermore, XLM-RoBERTa-twitter-hate was the best model out of all the models we tested because XLM-RoBERTa-twitter-hate was previously trained on a Twitter hate dataset which is a close topic to our problem. Additionally, the official ranking and score are presented in Table **??**, our team ranked in the top 9 and achieved such promising results on task 1: Stereotype Identification (hard label) with 63.000 F1-score. Our best result is lower than the F1-score of the Top 1 and Top 2 teams, which are -9.4 and -8.2, in turn.

Table 4
All submission official results

Models	Method	F1-score
BETO + XLM-RoBERTa (uncased) + XLM-RoBERTa-twitter-hate	Soft Voting	63.0
XLM-RoBERTa-twitter-hate	Adapter	62.4
BETO + XLM-RoBERTa (uncased) + XLM-RoBERTa-twitter-hate	Stacking	60.8

Ranking	Team	F1-score
Top 1	Brigada Lenguaje_1	72.4
Top 2	I2C-Huelva_1	71.2
Top 3	I2C-Huelva_2	70.1
Top 4	EUA_2	69.1
Top 5	EUA_3	68.5
Ours (Top 9)	**TaiDepZai999_UIT_AIC_1**	**63.0**

Table 5
Official Results for Task 1: Stereotype Identification with hard label

Figure 4 presents three confusion matrices of our official submission techniques: Soft Voting, Adapter [6], and Stacking, respectively on the evaluation set with Scikit-Learn [11] confusion matrix function in sub Task 1: Stereotype Identification. As depicted in Figure 4, our models perform pretty well on this task overall. However, even though the Stacking method outperforms the other two techniques in the evaluation set which register the smallest error rate in classifying stereotypes in both classes, in the test set Stacking method had the lowest F1-score out of the three methods. Moreover, we could observe that our Soft-Voting and Stacking approach has a descent error ratio in class 1 due to a lack of data in a particular class. Except for the Adapter-Head approach, the class 1 error is higher than the class 0 error because we can see that FN is lower than FP.

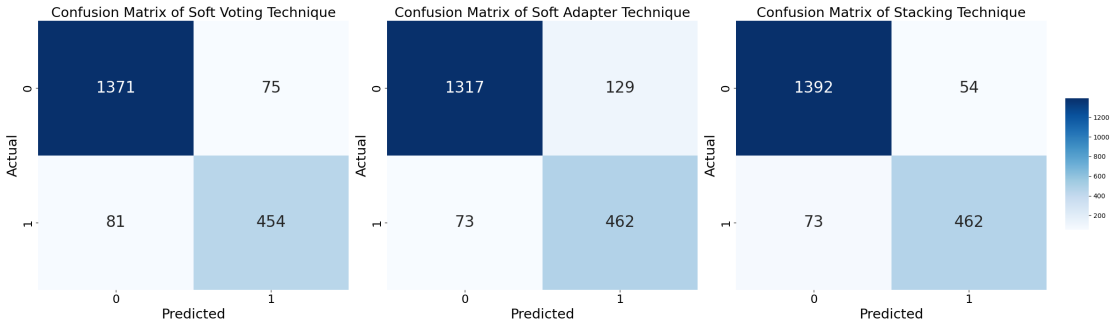

Figure 4: Confusion Matrix of three techniques on our submission.

6. Conclusion and Future Work

In this paper, we illustrate our approaches in DETEST-Dis IberLEF 2024 Task 1 [15]: Stereotype Identification, which achieved top 9th in official Task 1 Stereotype Identification with hard label ranking. We introduced different ensemble learning methods and Adapter-Head integrated with BERT-based models on classification tasks. By experimenting and analyzing, our approaches have achieved such promising results on task 1, and we believe that our methods could apply to reality tasks because of low computational cost in comparison to the large language models approach. Furthermore, also by investigating results, we realize that pre-processing and testing data on different formats can boost performance.

References

[1] L. Chiruzzo, S. M. Jiménez-Zafra, F. Rangel, Overview of IberLEF 2024: Natural Language Processing Challenges for Spanish and other Iberian Languages, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2024), co-located with the 40th Conference of the Spanish Society for Natural Language Processing (SEPLN 2024), CEUR-WS.org, 2024.

[2] 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. Le Scao, S. Gugger, M. Drame, Q. Lhoest, A. 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://aclanthology.org/2020.emnlp-demos.6. doi:10.18653/v1/2020.emnlp-demos.6.

[3] 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.

[4] F. Barbieri, L. Espinosa Anke, J. Camacho-Collados, XLM-T: Multilingual language models in Twitter for sentiment analysis and beyond, in: Proceedings of the Thirteenth Lan-

guage Resources and Evaluation Conference, European Language Resources Association, Marseille, France, 2022, pp. 258–266. URL: https://aclanthology.org/2022.lrec-1.27.

[5] P. He, X. Liu, J. Gao, W. Chen, Deberta: Decoding-enhanced bert with disentangled attention, in: International Conference on Learning Representations, 2021. URL: https://openreview.net/forum?id=XPZIaotutsD.

[6] C. Poth, H. Sterz, I. Paul, S. Purkayastha, L. Engländer, T. Imhof, I. Vulić, S. Ruder, I. Gurevych, J. Pfeiffer, Adapters: A unified library for parameter-efficient and modular transfer learning, arXiv preprint arXiv:2311.11077 (2023).

[7] H. Al-Omari, M. Abdullah, O. AlTiti, S. Shaikh, JUSTDeep at NLP4IF 2019 task 1: Propaganda detection using ensemble deep learning models, in: A. Feldman, G. Da San Martino, A. Barrón-Cedeño, C. Brew, C. Leberknight, P. Nakov (Eds.), Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda, Association for Computational Linguistics, Hong Kong, China, 2019, pp. 113–118. URL: https://aclanthology.org/D19-5016. doi:10.18653/v1/D19-5016.

[8] J. A. García-Díaz, S. M. J. Zafra, R. Valencia-García, Umuteam at iberlef-2022 detests task: Feature engineering for the identification and categorization of racial stereotypes in spanish, in: IberLEF@SEPLN, 2022. URL: https://api.semanticscholar.org/CorpusID:252015691.

[9] J. M. Vázquez, V. Pachón, C. T. Taybi, P. P. Sánchez, I2c at iberlef-2022 detests task: Detection of racist stereotypes in spanish comments using underbagging and transformers, in: IberLEF@SEPLN, 2022. URL: https://api.semanticscholar.org/CorpusID:252015923.

[10] S. Bates, T. Hastie, R. Tibshirani, Cross-validation: what does it estimate and how well does it do it?, Journal of the American Statistical Association (2023) 1–12.

[11] 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.

[12] S. S. Aluru, B. Mathew, P. Saha, A. Mukherjee, Deep learning models for multilingual hate speech detection, arXiv preprint arXiv:2004.06465 (2020).

[13] V. Basile, C. Bosco, E. Fersini, D. Nozza, V. Patti, F. M. Rangel Pardo, P. Rosso, M. Sanguinetti, SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter, in: J. May, E. Shutova, A. Herbelot, X. Zhu, M. Apidianaki, S. M. Mohammad (Eds.), Proceedings of the 13th International Workshop on Semantic Evaluation, Association for Computational Linguistics, Minneapolis, Minnesota, USA, 2019, pp. 54–63. URL: https://aclanthology.org/S19-2007. doi:10.18653/v1/S19-2007.

[14] J. Pfeiffer, I. Vulić, I. Gurevych, S. Ruder, MAD-X: An Adapter-based Framework for Multi-task Cross-lingual Transfer, arXiv preprint (2020). URL: https://arxiv.org/pdf/2005.00052.pdf.

[15] W. S. Schmeisser-Nieto, P. Pastells, S. Frenda, A. Ariza-Casabona, M. Farrús, P. Rosso, M. Taulé, Overview of DETESTS-Dis at IberLEF 2024: DETEction and classification of racial STereotypes in Spanish - Learn with Disagreement, Procesamiento del Lenguaje Natural 69 (2024).