Human After All: Using Transformer Based Models to Identify Automatically Generated Text

Jorge Fernández García, Isabel Segura-Bedmar

Computer Science and Engineering Department, Universidad Carlos III de Madrid, Av. de la Universidad, 30, 28911 Leganés, Madrid

Abstract

In this paper, we present our approach to detecting AI-generated texts in a multilingual context as part of the IberAuTexTification 2024 competition. We aimed to accurately classify texts in Spanish, English, Portuguese, Galician, Catalan, and Basque. Our method involves three distinct approaches: Support Vector Machines (SVMs), language-specific transformers, and an ensemble of multilingual transformers with logistic regression at the output. The Ensemble approach, which combines outputs from multiple multilingual transformers using logistic regression, achieved the highest performance, with a Macro-F1 score of 0.8050, securing first place in the competition. Language-specific transformers showed notable results with a Macro-F1 score of 0.7069, while SVMs achieved a Macro-F1 score of 0.6237.

Keywords

Artificial Inteligence, Ensemble models, Transformers, Large Language Models, Machine-Generated text, SVM, Languages, Automatic Text Identification

1. Introduction

With the advancement of large language models (LLMs), these tools have become essential for numerous everyday tasks. LLMs, such as GPT-3 [1] and GPT-4 [2], are increasingly utilized due to their capability to produce coherent and natural text [3]. This evolution has facilitated the automation of content generation, from news articles and social media posts to email drafting and academic tasks. However, this advanced capability presents a significant challenge: distinguishing between AI-generated texts and those written by humans has become increasingly difficult, and in some cases, nearly impossible [4].

This paper addresses this crucial issue by developing models specifically designed to identify AI-generated texts. Our research is framed within the IberAuTexTification competition [5], organized in Spain by the Spanish Society for Natural Language Processing (SEPLN) [6]. The competition aims to tackle the emerging challenge of detecting AI-generated texts in the context of the Iberian Peninsula. Unlike previous editions, the 2024 competition does not distinguish between languages, requiring models to classify texts regardless of the language used. The dataset includes all the languages spoken across the Iberian Peninsula: Spanish, English, Catalan, Galician, Basque, and Portuguese. The competition serves as a rigorous evaluation framework, reflecting real-world conditions where detection systems must be precise and robust to effectively differentiate between human and machine-generated texts. Two tasks were proposed in this competition. The goal of the first task is to determine whether the text has been automatically generated or not. The second task aims to detect the model used to generate an artificial text. Our team only participated in the first task.

Our primary approach leverages an ensemble of multilingual transformer models, incorporating logistic regression at the output layer to calculate the final probability. This method aims to enhance the model's ability to generalize across multiple languages and improve the accuracy of AI-generated text detection.

IberLEF 2024, September 2024, Valladolid, Spain

jorge.fergacria03@gmail.com (J. F. García); isegura@inf.uc3m.es (I. Segura-Bedmar)

0000-0002-0877-7063 (J. F. García); 0000-0001-7116-9338 (I. Segura-Bedmar)

2. IberAutextification 2024

The IberAutextification 2024 competition is dedicated to the challenging task of detecting AI-generated texts in a multilingual context. Participants are provided with a multilingual dataset that includes texts in six languages: Spanish, English, Portuguese, Galician, Catalan, and Basque. This competition emphasizes the importance of developing robust models that can handle the nuances and linguistic features of multiple languages simultaneously.

IberAutextification 2024 is divided into two tasks: (1) a binary classification task for determining whether texts are generated by AI or written by humans, and (2) a multi-classification task aimed at identifying the AI model used to generate a given text.

We have only participated in the first subtask. The main evaluation metric for this subtask is Macro-F1, which ensures a balanced evaluation of model performance across all classes and languages. This subtask presents a significant challenge due to the diversity of languages and the need for models to generalize well across different linguistic contexts.

3. Related Work

This section reviews other works presented for the binary text classification subtask in the AuTexTification 2023 competition [7]. Specifically, it describes the approaches and techniques used by different teams to tackle the task, as well as the results obtained. Unlike the 2024 edition, which proposes the challenge for Iberian languages, the AuTexTification 2023 task only addresses tasks for English and Spanish.

Piotr Przybyła, Nicolau Duran-Silva, and Santiago Egea-Gómez (2023) [8] achieved first place in both languages (Spanish and English). Their approach involved a bidirectional LSTM network trained with a combination of text-based features. This network primarily relied on a probabilistic model predicting the likelihood that the next token in a sequence was generated by a language model. They used DistilGPT2 [9], GPT-2 [10], GPT2-Medium[10], and GPT2-Large[10] for English texts, and GPT2-base-bne [11] and GPT2-large-bne [11] for Spanish texts. This allowed them to obtain a sequence with the probability of each token being generated by a language model. Additionally, they added word-level features such as word frequency, aiming for the LSTM to differentiate between word distributions in human and generated texts. They also studied grammatical errors using tools like LanguageTool [12]. Finally, they included RoBERTa model embeddings fine-tuned on the training dataset and text-level features like Part-Of-Speech and Word-Dependency. They achieved a Macro-F1 of 0.81 in English and 0.71 in Spanish.

Abburi et al. [13] used an ensemble of models such as BERT, DeBERTa, RoBERTa, and multilingual RoBERTa with a majority voting classifier, achieving a Macro-F1 of 0.73 in English and 0.65 in Spanish.

Martínez-Murillo et al. [14] used a multilingual BERT model fine-tuned with the whole training dataset (English and Spanish). The best result in English was obtained by training RemBERT (a more powerful version of BERT multilingual) exclusively with English texts, achieving a Macro-F1 of 0.73 . For the Spanish texts they used the model trained with both datasets obtaining a Macro-F1 of 0.67 in Spanish.

Gambini et al. [15] proposed a multi-layer dense neural network that captures three types of features: stylistic properties, embeddings from TwHIN-BERT-base, and Keras Tokenizer with a CNN for text pattern recognition. They achieved a Macro-F1 of 0.715 in English.

Alonso-Simón et al.[16] used a simple approach with TF-IDF representations and machine learning classifiers, with LinearSVC performing best. They achieved a Macro-F1 of 0.68 in English and 0.71 in Spanish.

Preda et al. [17] used an ensemble of fine-tuned language models like multilingual RoBERTa, multilingual BERT, and BERT trained on a Twitter dataset, combined with XGBoost and Virtual Adversarial Training (VAT). This ensemble achieved a Macro-F1 of 0.67 in both English and Spanish.

Claudiu Creanga and Liviu Petrisor Dinu [18] proposed a different approach, using a Convolutional

Neural Network (CNN) for text classification. They used features such as Vader sentiment scores [19], spelling errors, syllables, stop words, and word frequency, along with Google Word2Vec embeddings. The CNN had 5 layers with Batch Normalization and a Dropout layer. This system achieved a Macro-F1 of 0.66 in English.

Overall, transformers were a key component in most works presented in the competition, demonstrating their effectiveness in NLP tasks [20]. Ensemble methods combining various language models also proved highly effective. The winning team used transformers with LSTM networks, achieving the best results in both languages (Macro-F1 0.81 for English and 0.71 for Spanish). Classic approaches like SVM with TF-IDF vectors also yielded good results (around 0.68 in English and 0.71 for Spanish texts).

4. Methodology

This section outlines the approaches and ideas behind the models we presented for the IberAuTexTification 2024 competition. The entirety of this work has been developed in Python, utilising the Google Colab platform. For the training of the transformers, the freely available resources of this platform have been employed, including 13 GB of RAM and a Tesla T4 GPU.

Each task in the competition allowed for up to three submissions, and we decided to utilize this opportunity to propose three different approaches, each employing unique strategies to achieve the best results.

4.1. SVM

One of our proposed approaches in the competition is based on using a Support Vector Machine (SVM) classifier that employs TF-IDF (Term Frequency-Inverse Document Frequency) representations derived from Bag of Words (BOW). This method is particularly suitable due to the linguistic diversity present in the texts, where a specific treatment for each language is needed. Additionally, it does not require a pre-trained transformer model for each language, which may be hard to find. Moreover, this approach is easy to implement, needing only a list of stopwords, a tokenizer, and a lemmatizer for each language.

Due to the need for specific treatment for each language, both the training and evaluation datasets are divided by language. The text preprocessing process is adapted to each language as follows:

1. **Tokenization**: For Spanish, English, and Portuguese, tokenization functions from the NLTK library [21] are used. For Galician, Basque, and Catalan, the Simplemma library [22] with simple_tokenizer is employed, allowing for generic tokenization.
2. **Stopword Removal**: NLTK-provided stopwords are used for Spanish, English, and Portuguese. For the other languages, the stopwordsiso library [23] is used.
3. **Stemming and Lemmatization**: Stemming is performed using NLTK for Spanish, English, and Portuguese. For the other languages, lemmatization is again performed through Simplemma [22].
4. **Removal of Words with Digits or Special Symbols**: This step ensures that text representations are clean and uniform, excluding elements that do not provide significant value to the classification model.

A specific SVM model is trained for each language. Thus, six specialized SVM models are created, one for each language: Spanish, English, Portuguese, Galician, Basque, and Catalan.

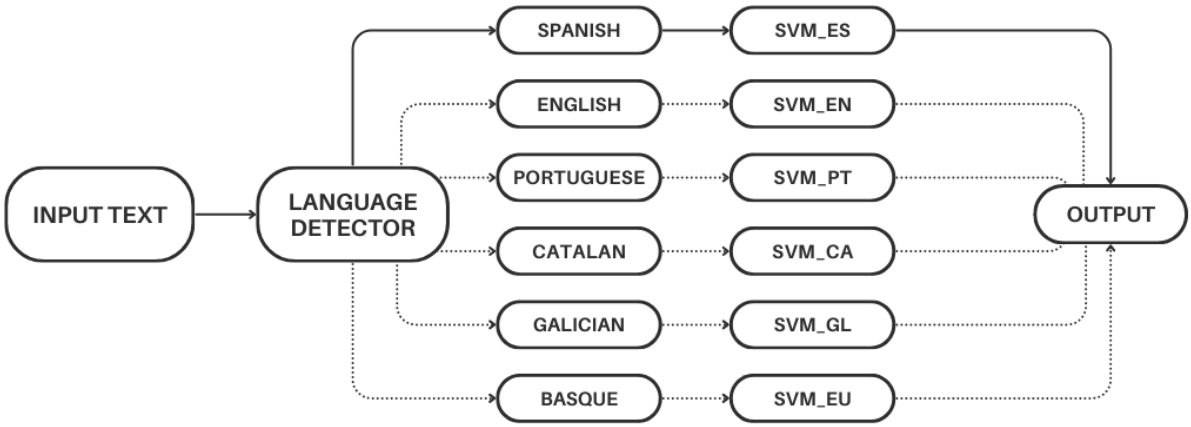

Figure 1: Architecture of the SVM system for SubTask 1 of IberAuTexTification 2024

The classification process for incoming texts (see Figure 1) is carried out as follows:

1. **Language Detection**: Each incoming text is first classified into one of the six dataset languages using the "ftlangdetect" library [24].
2. **SVM Model Application**: The text is processed using the preprocessing pipeline corresponding to its detected language. Subsequently, it is classified using the SVM model specifically trained for that language.

Finally, the classification results from the different SVM models are combined into a single dataset, which is presented as the final result for the competition.

This approach ensures that texts are processed and classified by leveraging the linguistic peculiarities of each language, optimizing the model's ability to detect AI-generated texts in a multilingual environment.

4.2. Transformers Specific to Each Language

To improve text detection we first opted to use pre-trained transformers for each language. Each transformer is fine-tuned with training data corresponding to its language. Below is a brief description of the transformers used.

- **Albert-base-v2 for English**: ALBERT (A Lite BERT) [25] is a lighter and more efficient version of BERT [26], designed to reduce model size and increase training speed without sacrificing performance. The model is available on Hugging Face [27]. The model was trained on the BookCorpus and Wikipedia corpora. The version used consists of 12 layers, 128-dimensional embeddings, 12 attention heads, and 11 million parameters.
- **RoBERTa-base-bne for Spanish**: RoBERTa [28] is a variant of BERT that optimizes pre-training with more data and configuration adjustments. The "base-bne" version was specifically trained on a large corpus of Spanish texts. Available on Hugging Face [29].
- **BERT-base-portuguese-cased for Portuguese**: This is an implementation of BERT trained on a large corpus of Portuguese texts. Available on Hugging Face [30].
- **BERT-galician for Galician**: A BERT-based model specifically adapted to Galician, trained and fine-tuned from a Spanish pre-trained model using a dataset obtained from the Galician Wikipedia. Available on Hugging Face [31].
- **RoBERTa-base-ca for Catalan**: This version of RoBERTa is pre-trained on a corpus of publicly available Catalan texts. Available on Hugging Face [32].
- **RoBasquERTa for Basque**: RoBasquERTa is a version of RoBERTa fine-tuned for the Basque language. Available on Hugging Face [33].

Each transformer model has its own tokenizer, which converts text inputs into a format compatible with the model. This process involves breaking down the text into individual tokens (words, subwords, or other linguistic units) and assigning them numerical IDs. The tokenizer also handles special characters, punctuation, and other aspects of text representation. To prevent overfitting and optimize computational resource usage, we employ an early stopping strategy [34]. This technique involves monitoring a validation metric during training. If the metric does not improve for a specified number of consecutive steps (patience), training is terminated early, and the model with the best validation performance is saved. This approach helps prevent the model from overlearning on the training data and from generalizing poorly to unseen data.

We implemented early stopping by monitoring the loss function on the validation set after every 100 training steps. If the loss function does not improve for a consecutive 500 steps (patience), the training is terminated, and the best model is saved. To ensure our Transformers are fine-tuned on each language, we first split the text by language using the "ftlangdetect" library. Then, we fine-tuned the Transformers on each language's corresponding text data. To monitor progress, we used 10% of the text in each language to calculate validation metrics every 100 training steps.

Once training is complete, the fine-tuned models are stored on Hugging Face [35], a public platform for sharing NLP models. This ensures that the models are accessible to the community, promoting reproducibility, verification of results, and the advancement of research by providing resources for other researchers and developers. The described approach ensures that Transformer models are trained efficiently and effectively, maximizing the use of available resources and ensuring the quality of the final models.

The text classification process (see Figure 2) is straightforward: first, the text's language is classified using the "ftlangdetect" tool, and it is assigned to the pre-trained and fine-tuned transformer for that specific language. Finally, the corresponding transformer infers the class of the text, determining whether it was generated by AI or not.

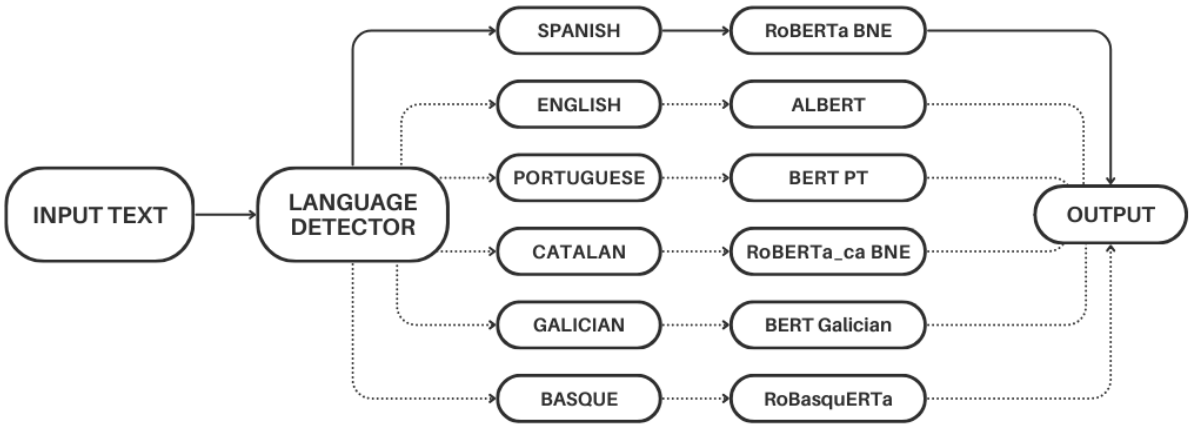

Figure 2: Architecture of the transformer system with language detector for SubTask 1 of IberAuTexTification 2024

This approach offers greater customization compared to the previous one. By studying the overall results of all the models developed for the 2023 competition, we observed a common issue: the inferred classes are highly imbalanced, with all models predicting a majority of generated texts. Since the output of the transformers is a probability, the classification threshold (typically 0.5) can be adjusted to favor predictions of the *Human* class.

4.3. Multilingual Transformer Ensemble

We implemented an Ensemble that combines the output of three multilingual transformers with a logistic regression algorithm at the output. Below is a brief description of the transformers used:

1. **DistilBERT-base-multilingual-cased**: Multilingual version of DistilBERT [36], trained on the concatenation of Wikipedia from up to 104 different countries. Model publicly available on Hugging Face [37].
2. **mDeBERTa-v3-base**: A variant of DeBERTa (Decoding-enhanced BERT with disentangled attention) [38] designed for multilingual tasks. Trained using the CC100 dataset [39], a massive multilingual corpus. Model publicly available on Hugging Face [40].
3. **XLM-RoBERTa-base**: A version of RoBERTa [28] trained in 100 different languages. It also uses the CC100 multilingual dataset. Available on Hugging Face [41].

Building upon the early stopping technique described earlier (Section 4.2), we split the data into three portions. Seventy percent (70%) goes towards training the corresponding multilingual Transformer. Fifteenth percent (15%) is used for validation during training. This portion helps us monitor performance and trigger early stopping if the validation loss function doesn't improve over 500 steps. The remaining Fifteenth percent (15%) is used to train a separate logistic regression model that will eventually combine the Transformers' outputs.

The text classification process (see Figure 3) is a combination of the three Transformer outputs. Each text is passed through the three multilingual transformers (DistilBERT, mDeBERTa, XLM-RoBERTa), and their output probabilities are obtained. These probabilities are used as input features for a logistic regression model, which infers the final class of the text.

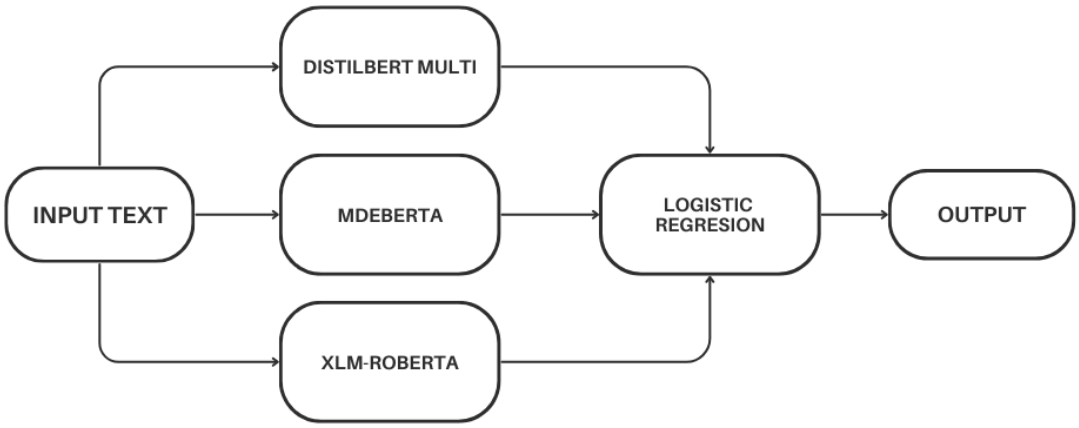

Figure 3: Architecture of the multilingual transformer Ensemble for SubTask 1 of IberAuTexTification 2024

This approach, with a multilingual transformer ensemble followed by logistic regression, leverages the combined strength of multiple transformers, optimizing accuracy and robustness in detecting AI-generated texts.

5. Results

This section discusses the results achieved in the competition. As of today (June 19, 2024), the labeled test dataset has not yet been released, so we only have the evaluation metric, *Macro-F1*, for each model and its ranking in the competition. With this limited data, we will attempt to analyze the results obtained by each system.

RANK	TEAM	RUN	MACRO-F1
1	jor_isa_uc3m	1	0.8050
2	gmc_fosunlp	1	0.7663
3	telescope_team	2	0.7579
4	iimasNLP	2	0.7188
5	gmc_fosunlp	2	0.7155
6	telescope_team	3	0.7118
7	jor_isa_uc3m	2	0.7069
8	iimasNLP	3	0.7051
	llmixtic_llama	baseline	0.6984
9	telescope_team	1	0.6965
10	iimasNLP	1	0.6793
	logistic-regression-word-1_2-char-2_6	baseline	0.6767
	me5-base	baseline	0.6349
30	jor_isa_uc3m	3	0.6237
	mdeberta-v3-base	baseline	0.6147
	multilingual-dec-512-shots	baseline	0.5862
	llmixtic_gpt	baseline	0.5317
	logistic-regression-readability	baseline	0.5095
	xlm-roberta-base	baseline	0.4997
	random	baseline	0.4972
	multilingual-dec-zero-shot	baseline	0.4402
	majority	baseline	0.3556

Table 1: Ranking and Macro-F1 of the models in the competition

Table 1 shows the ranking of the participating systems. Out of a total of **54** models submitted, we achieved first place with the multilingual transformer Ensemble, obtaining a *Macro-F1* score of 0.805. The second best model scored 0.7663, and the best baseline, 0.6984.

In the competition, our **SVM**-based model achieved a relatively low *Macro-F1* score of 0.6237, ranking 30th. Analyzing the model's predictions, we can infer the problem: the imbalance of the predicted classes. In this case, the SVMs predicted 70% of the samples as AI-generated. Considering that this is a competition and the test set contains many texts, it is likely that the classes in the test set are more or less balanced, and our classifier has a clear tendency to infer texts as generated, similar to what happened with the models in last year's competition. This imbalance may significantly impact the *Macro-F1* metric, as many human texts are being misclassified as generated.

The approach using **Transformers Specific to Each Language** showed significant improvement compared to the SVM. This method benefited from the transformers' ability to capture nuances and contexts specific to each language. Despite this improvement, the complexity of training and fine-tuning multiple models for different languages with limited data (between 10,000 and 20,000 texts) may have limited the model. In the competition, the model based on language-specific transformers achieved a *Macro-F1* score of 0.7069, ranking seventh.

Initially, this model exhibited the same issue as the SVMs, with inferred classes being highly imbalanced (80% generated texts). Unlike the previous model, the output of the transformers is directly a probability, making it easy to modify the decision threshold to prioritize the *Human* class. This adjustment resulted in the model having the following proportions: 49.8% *Human* and 50.2% *Generated*.

The highest performance was achieved with the **Multilingual Transformer Ensemble** approach, reaching first place in the competition with a *Macro-F1* score of **0.8050**. This result highlights the effec-

tiveness of combining multiple multilingual transformers to leverage their complementary capabilities. Given that this model achieved first place in the competition, it is worth providing some details about the process that led to the development of the Ensemble. To illustrate our model development process, we present in Table 2 the results of the individual multilingual transformers compared to the Ensemble on a validation set.

Model	Accuracy	Macro-F1	AUC-ROC
mDeBERTa	0.9573	0.9572	0.9582
XLM-RoBERTa	0.9209	0.9195	0.9161
Multi DistilBERT	0.9262	0.9253	0.9230
Ensemble	**0.9634**	**0.9632**	**0.9633**

Table 2: Classification Metrics in Validation Set for Multilingual Transformers and the Ensemble

As illustrated in the table, the Ensemble model is the most effective of the four, although mDeBERTa achieves comparable results, suggesting that it could also perform well on the test set. All models demonstrate exceptionally high performance, which may indicate potential overfitting to the training dataset. It is important to note that class imbalance was not an issue in the predictions over the validation dataset. This is a notable contrast to the situation observed when testing the Ensemble model with the evaluation dataset. As observed with SVM and Transformers for each language, the Ensemble exhibited a significant class imbalance in its predictions over the test set (30% human, 70% generated). This discrepancy may be attributed to the existing difference between the training and evaluation domains, as the models were trained on texts of significantly different nature from those in the evaluation dataset.

To address this issue, we prioritized the human class through hyperparameter tuning, as detailed in Table 3, with the objective of producing balanced predictions (approximately 50% Human and 50% Generated). Specifically, the `class_weight` parameter was adjusted to modify the distribution of weights assigned to the classes during training, thereby prioritizing *Human* predictions. The final selected parameters using a Grid Search CV algorithm were: `class_weight` = {0: 50, 1: 1}, `C` = 10, and `solver` = `liblinear`.

Hyperparameter	Values	Description
`class_weight`	{0:50, 1:1}, {0:60, 1:1}, {0:70, 1:1}, {0:80, 1:1}, {0:90, 1:1}, {0:100, 1:1}, {0:150, 1:1}, {0:200, 1:1}	Class weights to balance the sample. 0: *Human*, 1: *Generated*
`C`	0.001, 0.01, 0.1, 1, 10, 100, 1000	Regularization parameter
`solver`	`newton-cg`, `lbfgs`, `liblinear`	Optimization algorithms

Table 3: Hyperparameter search for the logistic regression model used in the competition Ensemble 2024

In summary, the multilingual model Ensemble has proven to be the most consistent model in the competition for classifying automatically generated texts in different languages of the peninsula. Several factors contribute to the success of this system, including the fact that transformers require a large amount of data for training on a task. Compared to other approaches with fewer texts per language, each of these models is trained on approximately 60,000 texts from all languages. Another important factor could be the use of early stopping to prevent overfitting to the training set. Since we know that the test set texts belong to different domains not present in the training dataset, avoiding overfitting is crucial.

6. Conclusions

This work has demonstrated that an ensemble of transformers with a logistic regression algorithm in the output can solve the task of detecting AI-generated texts with high precision and confidence. The results obtained surpass the rest of the models presented in this year's competition.

Our accumulated experience enabled us to secure first place in the IberAuTexTification 2024 competition, which addresses the identification of AI-generated texts in a multilingual context. The dataset includes texts in Spanish, English, Portuguese, Galician, Catalan, and Basque. We presented three different approaches. Two of these approaches employed specialized models (SVMs and transformers) for each language, using a language detector at the input to direct the texts to the corresponding model. These systems achieved a *Macro-F1* score of 0.7069 with specialized transformers and 0.6237 with SVMs. However, the most successful approach was an ensemble of multilingual transformers with a logistic regression model at the output. This model achieved a *Macro-F1* score of 0.8050, surpassing the second-place team, which achieved a *Macro-F1* score of 0.7663. As of the date this document is written, the proceedings of the competition have not yet been published, so we do not have information about the approaches used by other teams.

For future projects, there are several promising lines of development. Outside the competitive arena, it would be beneficial to explore expanding the dataset, as it is relatively easy to obtain new and varied data, both human and generated. However, this was not allowed for the competition.

The time and resource limitations during this project prevented us from conducting deep and comprehensive training (such as hyperparameter selection) of the transformers with the competition data. Performing these experiments on more powerful hardware could improve the results. Additionally, we used very similar transformer models, all of which were variations of BERT. It could be interesting to incorporate more diverse models such as GPT [42] or Llama [43] into the Ensemble.

Furthermore, although we focused primarily on the use of NLP models, we overlooked some classic text analysis techniques. Within the context of this Ensemble, we can incorporate additional features alongside the outputs of the transformer models into the logistic regression model. When increasing the number of features, it is also worth exploring the use of convolutional deep neural networks to more effectively capture the information that each feature contributes. Some tools and techniques that could be useful include:

- **N-gram Analysis**: Evaluating the frequency of word sequences (bigrams, trigrams, etc.) to capture common patterns in AI-generated texts.
- **Topic Modeling**: Using techniques such as Latent Dirichlet Allocation (LDA) to identify predominant themes in the texts and how these may differ between human and generated texts.
- **Sentiment Analysis**: Evaluating the tone and emotion expressed in the texts, as automatic texts may exhibit different patterns in sentiment expression.
- **Grammatical and Syntactic Features**: Analyzing the grammatical and syntactic structure of the texts, such as the use of verb tenses, sentence structures, and grammatical errors that may be more common in AI-generated texts.
- **Stylometric Tools**: Using techniques to analyze writing style, such as sentence length, vocabulary variability, and the use of certain types of words or phrases.

Implementing these additional techniques can provide valuable insights and complement the transformer models, thereby improving the accuracy in detecting AI-generated texts. With the appropriate integration of these methodologies, future models could achieve even higher levels of accuracy, providing more robust and effective solutions for identifying AI-generated texts.

Acknowledgments

This work was supported by ACCESS2MEET project (PID2020-116527RB-I0) supported by MCIN AEI/10.13039/501100011033/

References

[1] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, arXiv preprint arXiv:2005.14165 (2020).

[2] OpenAI, Gpt-4 technical report, arXiv preprint arXiv:2303.08774 (2023).

[3] Y. Yao, J. Duan, K. Xu, Y. Cai, Z. Sun, Y. Zhang, A survey on large language model (llm) security and privacy: The good, the bad, and the ugly, High-Confidence Computing (2024) 100211.

[4] Communications of the ACM, The science of detecting llm-generated text, https://cacm.acm.org/research/the-science-of-detecting-llm-generated-text/, 2024. Online; Accessed: 17/06/2024.

[5] A. M. Sarvazyan, J. Á. González, F. Rangel, P. Rosso, M. Franco-Salvador, Overview of iberautextification at iberlef 2024: Detection and attribution of machine-generated text on languages of the iberian peninsula, Procesamiento del Lenguaje Natural 73 (2024).

[6] 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.

[7] A. M. Sarvazyan, J. Á. González, M. Franco-Salvador, F. Rangel, B. Chulvi, P. Rosso, Overview of autextification at iberlef 2023: Detection and attribution of machine-generated text in multiple domains, Procesamiento del Lenguaje Natural. 2023 71 (2023) 275–288.

[8] P. Przybyła, N. Duran-Silva, S. Egea-Gómez, IâĂŹve seen things you machines wouldnâĂŹt believe: Measuring content predictability to identify automatically-generated text, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2023). CEUR Workshop Proceedings, CEUR-WS, Jaén, Spain, 2023.

[9] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilgpt2: A distilled version of openai's gpt-2, arXiv preprint arXiv:1910.01108 (2019).

[10] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al., Language models are unsupervised multitask learners, OpenAI blog 1 (2019) 9.

[11] A. G. FandiÃśo, J. A. EstapÃĺ, M. PÃămies, 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). URL: https://upcommons.upc.edu/handle/2117/367156#.YyMTB4X9A-0.mendeley. doi:10.26342/2022-68-3.

[12] D. Naber, LanguageTool, LanguageTool Development Team, 2023. URL: https://languagetool.org, available at https://languagetool.org.

[13] H. Abburi, M. Suesserman, N. Pudota, B. Veeramani, E. Bowen, S. Bhattacharya, Generative ai text classification using ensemble llm approaches, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2023). CEUR Workshop Proceedings, CEUR-WS, Jaén, Spain, 2023.

[14] I. Martínez-Murillo, R. Sepúlveda-Torres, E. Saquete, E. Lloret, M. Palomar, Team gplsi at autextification shared task: Determining the authorship of a text, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2023). CEUR Workshop Proceedings, CEUR-WS, Jaén, Spain, 2023.

[15] M. Gambini, M. Avvenuti, F. Falchi, M. Tesconi, T. Fagni, Detecting generated text and attributing language model source with fine-tuned models and semantic understanding, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2023). CEUR Workshop Proceedings, CEUR-WS, Jaén, Spain, 2023.

[16] L. Alonso Simón, J. A. Gonzalo Gimeno, A. M. Fernández-Pampillón Cesteros, M. Fernández Trinidad, M. V. Escandell Vidal, Using linguistic knowledge for automated text identification, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2023). CEUR Workshop Proceedings, CEUR-WS, Jaén, Spain, 2023.

[17] A.-A. Preda, D.-C. Cercel, T. Rebedea, C.-G. Chiru, Upb at iberlef-2023 autextification: Detection of machine-generated text using transformer ensembles, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2023). CEUR Workshop Proceedings, CEUR-WS, Jaén, Spain, 2023.

[18] C. Creanga, L. P. Dinu, Automated text identification using cnn and training dynamics, in: Pro-

ceedings of the Iberian Languages Evaluation Forum (IberLEF 2023). CEUR Workshop Proceedings, CEUR-WS, Jaén, Spain, 2023.
[19] C. Hutto, E. Gilbert, Vader: A parsimonious rule-based model for sentiment analysis of social media text, in: Proceedings of the international AAAI conference on web and social media, volume 8, 2014, pp. 216–225.
[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Å. Kaiser, I. Polosukhin, Attention is all you need, Advances in Neural Information Processing Systems (2017).
[21] S. Bird, E. Loper, E. Klein, Natural language toolkit, https://www.nltk.org/, 2009. Online; Accessed: 17/06/2024.
[22] H. Falk, Simplemma: Simple multilingual lemmatizer for python, https://github.com/henrikfalk/simplemma, 2020. Online; Accessed: 17/06/2024.
[23] A. BarrÃşn-CedeÃśo, stopwords-iso: A collection of stopwords for multiple languages, https://github.com/stopwords-iso/stopwords-iso, 2018. Online; Accessed: 17/06/2024.
[24] S. Montabone, ftlangdetect: A fasttext-based language detector for python, https://fasttext.cc/blog/2017/10/02/blog-post.html, 2020. Online; Accessed: 17/06/2024.
[25] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, R. Soricut, Albert: A lite bert for self-supervised learning of language representations, arXiv preprint arXiv:1909.11942 (2019).
[26] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).
[27] H. Face, Albert-base-v2, https://huggingface.co/albert/albert-base-v2, 2024. Online; Accessed: 17/06/2024.
[28] 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, arXiv preprint arXiv:1907.11692 (2019).
[29] H. Face, Roberta-base-bne, https://huggingface.co/PlanTL-GOB-ES/roberta-base-bne, 2023. Online; Accessed: 17/06/2024.
[30] H. Face, Bert-base-portuguese-cased, https://huggingface.co/neuralmind/bert-base-portuguese-cased, 2022. Online; Accessed: 17/06/2024.
[31] H. Face, Bert-galician, https://huggingface.co/fpuentes/bert-galician, 2023. Online; Accessed: 17/06/2024.
[32] H. Face, Roberta-base-ca, https://huggingface.co/PlanTL-GOB-ES/roberta-base-ca, 2022. Online; Accessed: 17/06/2024.
[33] H. Face, Robasquerta, https://huggingface.co/mrm8488/RoBasquERTa, 2022. Online; Accessed: 17/06/2024.
[34] L. Prechelt, Early stopping — but when?, in: Neural Networks: Tricks of the Trade, Springer, 1998, pp. 55–69.
[35] H. Face, Perfil de huggingface: jorgefg03, https://huggingface.co/jorgefg03, 2024. Online; Accessed: 17/06/2024.
[36] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, arXiv preprint arXiv:1910.01108 (2019).
[37] H. Face, Distilbert-base-multilingual-cased, https://huggingface.co/distilbert/distilbert-base-multilingual-cased, ???? Online; Accessed: 17/06/2024.
[38] P. He, X. Liu, J. Gao, W. Chen, Deberta: Decoding-enhanced bert with disentangled attention, arXiv preprint arXiv:2006.03654 (2020).
[39] G. Wenzek, M.-A. Lachaux, A. Conneau, V. Chaudhary, F. GuzmÃąn, A. Joulin, E. Grave, Cc100: A monolingual dataset collection for many languages, https://data.statmt.org/cc-100/, ???? Online; Accessed: 17/06/2024.
[40] H. Face, mdeberta-v3-base, https://huggingface.co/microsoft/mdeberta-v3-base, ????. Online; Accessed: 17/06/2024.
[41] H. Face, Xlm-roberta-base, https://huggingface.co/FacebookAI/xlm-roberta-base, ????. Online; Accessed: 17/06/2024.
[42] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, Improving language understanding by generative pre-training, OpenAI 12 (2018) 212–223.

[43] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. RoziÃĺre, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, G. Lample, Llama: Open and efficient foundation language models, 2023. URL: https://arxiv.org/abs/2302.13971. arXiv:2302.13971.