

A Method for Identifying Bot Generated Text : Notebook for IberLEF 2023

Maryam Elamine^{1,*}, Asma Mekki¹ and Lamia Hadrach Belguith¹

¹MIRACL laboratory, Faculty of Economics and Management (University of Sfax), Sfax, Tunisia

Abstract

This paper investigates our proposed method for detecting generated texts and identifying the model that generated them, which were the center of interest in the AuTextification competition. We proposed a deep learning model approach, implementing a Sequential model for the first task and a machine learning approach with the SVM algorithm for the second task. Both methods were tested on multilingual data; specifically a Spanish and English dataset. For the first task, we achieved a macro F1-score of 54.92% for English and 51.38% for Spanish. However, the results for the second task were not satisfactory, with only 14.61% for English and 17.93% for Spanish.

Keywords

Generated text identification, model attribution, AuTextification challenge, text classification

1. Introduction

For a large number of people worldwide, social media platforms have become significant sources of information. Recent advancements in automatic text generation have enabled the creation of brief, coherent text that mimics the style of the human-elicited text used for training the models [1, 2, 3, 4, 5].

Text Generative Models (TGMs) have been shown to perform well in creating text that closely resembles human language [6]. Indeed, automated text generation based on neural language models has reached levels of performance that are almost identical to human-written text [7]. Since modern advances in neural language modeling allow for the quick production of enormous amounts of text, these generated texts appear as if they were written by humans [8]. Moreover, nowadays few-shot learning has been used to demonstrate that large language models can perform admirably across a range of natural language tasks, significantly reducing the amount of task-specific training examples required to adapt the model to a given application [9]. In fact, there has been a lot of research interest in the abilities of humans and automatic discriminators to recognize machine-generated text, but humans and machines use different cues to determine their actions [8].


It is also understandable how some researchers affirm that "Machine-generated text poses a possible risk to both the public and academics since it can discredit legitimate studies by using


IberLEF 2023, September 2023, Jaén, Spain

*Corresponding author.

✉ mary.elamine@gmail.com (M. Elamine); asma.elmekki.ec@gmail.com (A. Mekki); lamia.belguith@fsegs.usf.tn (L. H. Belguith)

ORCID 0000-0001-7272-0224 (M. Elamine); 0000-0003-3140-3171 (A. Mekki); 0000-0002-4868-657X (L. H. Belguith)

 © 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)

persuasive, artificial text" [10]. In some cases, expanding language models does not, in fact, improve their ability to interpret user intent. Large language models, for instance, may produce results that are misleading or insignificant to the user [11].

It has been demonstrated that large language models (LLMs) can complete new tasks based on a few examples or instructions in natural language. Despite the fact that these features have led to widespread adoption, the majority of LLMs are created by resource-rich organizations and often remain inaccessible to the public [12].

The AuTextTification (Automated Text Identification) challenge [13] provided four datasets; two for each task in both English and Spanish. For the first task, participants were given a text and they must decide whether or not it was generated automatically. As for the second, A text that was generated automatically will be given to the participants, and they must identify the model that produced it [14]. As it was more described in their site, a binary classification task with two classes –human and generated– makes up the first subtask. A multiclass classification task with six classes (A, B, C, D, E, and F), each class representing a text generation model, is the second subtask. Since the dataset was gathered from corpora that were made available to the public, certain limitations were placed by the organizers.

- It is possible to use publicly accessible pretrained models from the state-of-the-art. However, we must only use text that was derived from the training data. i.e., only text derived from the training data may be used for data augmentation or any additional self-supervised pretraining,
- It is also permitted to use lexicons, knowledge bases, and other sources of structured data,
- It is prohibited to use data from one subtask in the other subtask.

It is important to note that all proposed methods will be described in detail in the overview paper published by the organizers [15, 16].

In this paper, we present two proposed methods; one that employs a deep learning approach and the other implements a machine learning approach. For each subtask, all files have been read from their respective provided training datasets. For the generated text identification task, we implemented, for both languages, a Sequential model with three layers. With the English language we achieved a macro F1-score of 54.92% with the test dataset and for Spanish we achieved a macro F1-score of 51.38%. Whereas for the model identification task, we experimented three classification models namely, SVM (Support Vector Machine), Random Forest and Naïve Bayes. SVM was the best classification model in this case. For this subtask, the results were disappointing, with a macro F1-score of only 14.61% for English and 17.93% for Spanish with the test datasets.

The remainder of this paper is structured as follows, Section 2 presents the evaluation conditions in which we built our proposed models for both tasks. In section 3, we give a detailed overview of the performed experiments on both the training and test datasets and we report the achieved results . Finally, we finish with some concluding remarks and future work perspectives in Section 4.

2. Evaluation Conditions

In this competition, the organizers provided four datasets; two sets for each subtask in Spanish and English. They also provided baseline models that participants could use. However, there were certain constraints defined by the organizers. Participants were allowed to use publicly accessible pre-trained models, but only text derived from the training data could be used for data augmentation or additional self-supervised pre-training. Lexicons, knowledge bases, and other sources of structured data were permitted as well. It was explicitly prohibited to use data from one subtask in the other. As for the baseline models at our disposal, they were as follows:

- Classical machine learning models: Logistic Regression with bag of n-grams at word and character levels and Low-Dimensionality Statistical Embeddings (LDSE) [17].
- Pre-trained deep learning models:
 - For the Spanish language: Symanto Brain [18], XLM-RoBERTa [19], MDeBERTa [20], RoBERTa-Large-BNE [21] and BETO [22].
 - For the English language: Symanto Brain, XLM-RoBERTa [19], MDeBERTa [20], RoBERTa [23] and DeBERTa [24].

While participants were allowed to experiment with different hyper-parameter configurations and explore new methodologies and models using these baselines. Our proposed approach considers the defined constraints and introduces its own approach.

For the first task, we proposed a deep learning model. Indeed, we tokenized the dataset and vectorized the labels. Then we defined our model. We implemented, for both languages, a Sequential model with three layers. We worked with an embedding layer to learn the dense representations of words, This layer displays randomly initialized embeddings, where the embeddings were created during the training of our neural network. Then we flattened the output into a 1-dimensional array. Finally, in order to build our binary classification, we added a fully connected layer with a sigmoid activation function. With the English language we achieved a macro F1-score of 54.92% with the test dataset and for Spanish we achieved a macro F1-score of 51.38%.

Whereas for the model identification task, we first preprocessed the data, then we tokenized the text into a set of words and lemmatized the text. Afterwards, we vectorized our data using TF-IDF. Finally, we experimented three classification models namely, SVM (Support Vector Machine), Random forest and Naïve Bayes. SVM was the best classification model in this case so we submitted our run with this model only. For this subtask, the results were very disappointing, indeed, we achieved on the test data for English a macro F1-score of 14.61% . For Spanish, we managed to achieve on the test data a macro F1-score of 17.93%.

3. Experiments and Results

In this section, we provide a detailed presentation of the various experiments performed for both tasks on the training and testing datasets.

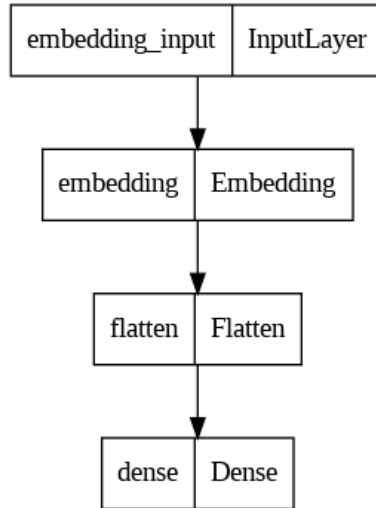


Figure 1: The architecture of the proposed model for bot classification task.

3.1. Results on the training dataset

For the first task, we proposed a deep learning model. Indeed, we tokenized the dataset and vectorized the labels. Then we defined our model. We implemented, for both languages, a Sequential model with three layers. In our work we experimented the Tensorflow’s Keras API. We worked with an embedding layer to learn the dense representations of words. This layer was utilized to capture word representations within our model. Then we flattened the output into a 1-dimensional array. Finally, in order to build our binary classification, we added a fully connected layer with a sigmoid activation function. With the English language we achieved an F-score of 77.5% with the training dataset and for Spanish we achieved an F-score of 80.1% with 15% split for testing, a batch_size of 256, 10 epochs and a Dense layer equal to 32. Table 1 gives a summary of the results from the training dataset while experimenting different parameters, mainly the activation function. We also experimented different optimizers, however, adam optimizer was our best choice for the task at hand.

For the second subtask, we wanted to continue on the same footsteps and test the deep learning model on the dataset in both languages, however the results were very disappointing. Table 2 gives an overview on the obtained results with the training dataset in both languages.

As shown in Table 2, we achieved really bad results with a deep learning approach, which made us change into classic Machine Learning models. We believe these results are so low because compared to the first subtask, this classification problem is a multi-class one (5 classes in total) which made it difficult for the model to adequately learn the patterns of each class (a language model in the case of this classification problem).

Ergo, for the identification task, we proposed a machine learning model. To do so, we first preprocessed the data by removing blank spaces, lowercasing all the text, then we tokenized the

text into a set of words, removing stopwords and numerical characters and lastly lemmatizing the text. Afterwards, we vectorized our data using TF-IDF. As a reminder, we experimented three classification models; SVM, Random forest and Naïve Bayes, however SVM gave us the highest F1-score. For this subtask, we achieved on the training data for English an F1-score of 32.10%, 37.41% and 33.47% with Naïve Bayes, SVM and Random Forest, respectively. For Spanish, we managed to achieve on the training data an F1-score of 39.39%, 46.04% and 40.97% with Naïve Bayes, SVM and Random Forest, respectively. Table 3 gives an overview on the obtained results on the training dataset in terms of accuracy score with the various models using different parameters.

Table 1

Results on the Training data of the first subtask with the proposed Deep Learning model

Model + Parameters	English F1-score	Spanish F1-score
batch_size=256, epochs=10, activation='sigmoid', optimizer='adam', loss='binary_crossentropy'	77.50%	80.10%
batch_size=256, epochs=10, activation='softmax', optimizer='adam', loss='binary_crossentropy'	35.50%	31.60%
batch_size=256, epochs=10, activation='relu', optimizer='adam', loss='binary_crossentropy'	76.10%	79.90%

Table 2

Results on the Training data of the second subtask with a deep learning model

Model + Parameters	English F1-score	Spanish F1-score
dense=16, batch_size=16, epochs=4, activation='relu', optimizer='adam', loss='binary_crossentropy'	4.60%	5.00%
dense=16, batch_size=16, epochs=4, activation='sigmoid', optimizer='adam', loss='binary_crossentropy'	4.50%	4.00%
dense=16, batch_size=16, epochs=4, activation='softmax', optimizer='adam', loss='binary_crossentropy'	4.60%	5.10%

3.2. Results on the test dataset

Following the obtained results on the training data, we tested the proposed models on the provided test data. This data was provided without any labels, so we could not determine exactly what the outcome of our prediction will accomplish. In fact, we submitted for each subtask and each language the results of a single run each, which were evaluated by the organizers based on the macro F1-score. We managed to achieve a macro F1-score values of 54.92%, 51.38%, 14.61% and 17.93% for the English and Spanish languages in the first task and for English and Spanish in the second subtask, respectively. Table 4 shows our obtained results for each subtask as they were published in the competition Website [25].

Table 3

Results on the Training data for the second subtask with the tested Machine Learning models

Model + Parameters	English - Accuracy	Spanish - Accuracy
Naïve Bayes	55.90%	58.73%
SVM: C=1.0, kernel='linear', degree=3, gamma='auto'	62.89%	67.66%
SVM: C=1.0, kernel='rbf', degree=3, gamma='auto'	18.15%	17.31%
SVM: C=1000.0, kernel='linear', degree=3, gamma='auto'	28.30%	32.20%
Random Forest: n_estimators=100, random_state=42	34.50%	42.72%
Random Forest: n_estimators=100, max_depth=6, random_state=0	29.86%	36.25%

Indeed, we experimented different parameters for the models as well as in the training process. We experimented different values for the TF-IDF feature size from 5000 to 5500 and managed to obtain the best results when the feature size was equal to 5500. Those results were then submitted to the competition.

Table 4

Results on the Test data for all subtasks.

Subtask	Macro-F1	Confidence interval
run1 - Subtask1 (EN)	54.92%	(54.4, 55.63)
run1 - Subtask1 (ES)	51.38%	(50.74, 52.07)
run1 - Subtask2 (EN)	14.61%	(13.9, 15.45)
run1 - Subtask2 (ES)	17.93%	(16.92, 18.84)

4. Conclusion

In this paper we have discussed our participation in the AuTextTification competition for the bot classification and model identification subtasks. For the first task, we proposed a deep learning model and tested it on both the English and Spanish corpora. For the second task, we experimented with three machine learning algorithms: SVM, Naïve Bayes and Random Forest, again on both corpora. The implemented models achieved a macro F1-score values of 54.92%, 51.38%, 14.61% and 17.93% for the English and Spanish languages in the respective tasks. While our achieved results may not be the best, this competition provided us with a valuable opportunity to explore new challenges and methodologies. It has also shed light on potential future directions for our work, where we can further refine and enhance our approaches to address the tasks at hand.

References

- [1] J. Tourille, B. Sow, A. Popescu, Automatic detection of bot-generated tweets, 2022.
- [2] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, Language models are unsupervised multitask learners, 2019.
- [3] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, Q. V. Le, Xlnet: Generalized autoregressive pretraining for language understanding, 2020. [arXiv:1906.08237](https://arxiv.org/abs/1906.08237).
- [4] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners, 2020. [arXiv:2005.14165](https://arxiv.org/abs/2005.14165).
- [5] N. S. Keskar, B. McCann, L. R. Varshney, C. Xiong, R. Socher, Ctrl: A conditional transformer language model for controllable generation, 2019. [arXiv:1909.05858](https://arxiv.org/abs/1909.05858).
- [6] G. Jawahar, M. Abdul-Mageed, L. V. S. Lakshmanan, Automatic detection of machine generated text: A critical survey, 2020. [arXiv:2011.01314](https://arxiv.org/abs/2011.01314).
- [7] V. Liyanage, D. Buscaldi, A. Nazarenko, A benchmark corpus for the detection of automatically generated text in academic publications, 2022. [arXiv:2202.02013](https://arxiv.org/abs/2202.02013).
- [8] D. Ippolito, D. Duckworth, C. Callison-Burch, D. Eck, Automatic detection of generated text is easiest when humans are fooled, 2020. [arXiv:1911.00650](https://arxiv.org/abs/1911.00650).
- [9] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, N. Fiedel, Palm: Scaling language modeling with pathways, 2022. [arXiv:2204.02311](https://arxiv.org/abs/2204.02311).
- [10] J. Rodriguez, T. Hay, D. Gros, Z. Shamsi, R. Srinivasan, Cross-domain detection of GPT-2-generated technical text, in: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Seattle, United States, 2022, pp. 1213–1233. URL: <https://aclanthology.org/2022.naacl-main.88>. doi:10.18653/v1/2022.naacl-main.88.
- [11] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, R. Lowe, Training language models to follow instructions with human feedback, 2022. [arXiv:2203.02155](https://arxiv.org/abs/2203.02155).
- [12] B. Workshop, T. L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A. S. Luccioni, F. Yvon, M. Gallé, J. Tow, A. M. Rush, S. Biderman, A. Webson, et al., Bloom: A 176b-parameter open-access multilingual language model, 2023. [arXiv:2211.05100](https://arxiv.org/abs/2211.05100).
- [13] AuTexTification site, Homepage, <https://sites.google.com/view/autextification/home>, 2023.
- [14] AuTexTification description, Task description, <https://sites.google.com/view/>

- autextification/task-description, 2023.
- [15] 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, in: *Procesamiento del Lenguaje Natural*, Jaén, Spain, 2023.
 - [16] S. M. Jiménez-Zafra, F. Rangel, M. Montes-y Gómez, Overview of IberLEF 2023: Natural Language Processing Challenges for Spanish and other Iberian Languages, *Procesamiento del Lenguaje Natural* 71 (2023).
 - [17] F. Rangel, M. Franco-Salvador, P. Rosso, A low dimensionality representation for language variety identification, 2017. [arXiv:1705.10754](https://arxiv.org/abs/1705.10754).
 - [18] Symanto-brain, Symanto Brain Website, <https://www.symanto.com/nlp-tools/symanto-brain/>, 2010.
 - [19] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, 2020. [arXiv:1911.02116](https://arxiv.org/abs/1911.02116).
 - [20] P. He, J. Gao, W. Chen, Deberv3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing, 2023. [arXiv:2111.09543](https://arxiv.org/abs/2111.09543).
 - [21] A. Gutiérrez-Fandiño, J. Armengol-Estapé, M. Pàmies, J. Llop-Palao, J. Silveira-Ocampo, C. P. Carrino, C. Armentano-Oller, C. Rodríguez-Penagos, A. Gonzalez-Agirre, M. Villegas, Maria: Spanish language models, *Procesamiento del Lenguaje Natural* (2022) 39–60. URL: <https://doi.org/10.26342/2022-68-3>. doi:10.26342/2022-68-3.
 - [22] J. Cañete, G. Chaperon, R. Fuentes, J.-H. Ho, H. Kang, J. Pérez, Spanish pre-trained bert model and evaluation data (2020) 1–10.
 - [23] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, 2019. [arXiv:1907.11692](https://arxiv.org/abs/1907.11692).
 - [24] P. He, X. Liu, J. Gao, W. Chen, Deberta: Decoding-enhanced bert with disentangled attention, 2021. [arXiv:2006.03654](https://arxiv.org/abs/2006.03654).
 - [25] AuTexTification, Results, <https://sites.google.com/view/autextification/results>, 2023.