SINAI at AuTexTification in IberLEF 2023: Combining All Layer Embeddings for Automatically Generated Texts.

César Espin-Riofrio[1,*], Jenny Ortiz-Zambrano[1] and Arturo Montejo-Ráez[2]

[1] *University of Guayaquil, Delta Av. s/n, Guayaquil, 090510, Ecuador*

[2] *University of Jaén, Las Lagunillas s/n, Jaén, 23071, Spain*

Abstract

Automatic text generation models have evolved with great advances in recent years. These models mimic human language and can generate convincing texts that can deceive readers, which can be influenced in making the wrong decisions. OpenAI's ChatGPT, based on artificial intelligence, specializes in dialogue and can generate a variety of texts according to the context requested by the user. Detecting whether a text has been written by a human or generated by a machine has aroused great interest in the scientific community, where Natural Language Processing and Machine Learning techniques play a crucial role in identifying whether a text has been produced by a person or not. In this paper, we describe our proposed method for AuTexTification subtask 1 in IberLEF 2023: Human or Generated. We fine-tune a predefined model, using the embeddings of the initial tokens of all BERT-based Transformers model layers, as features. Our prediction with the test dataset was not good, however, our training evaluation metrics were. We will continue experimenting to improve the model.

Keywords

Natural Language Processing, Transformer models, Text classification, Human or Generated

1. Introduction

In today's digital era, automatic text generation has evolved rapidly. Models based on artificial intelligence (AI), known as text generation models (TGM), have been developed [1]. As a result, a considerable amount of fake news, misleading content, fake product reviews, etc., are generated on the Internet. The AutTextTification task [2] at the evaluation forum IberLEF in 2023 encourages participants to explore methods and algorithms to automaticall identify content generated artificially.

Our approach is based in the intuition that style-related information may be encoded throughout all the layers of a transformer model. Therefore, automatically generated text should generate encodings different from those coming from human-written sequences. We have developed a network that combines the outputs of all intermediate layers in a weighted-average

IberLEF 2023, September 2023, Jaén, Spain

[*] Corresponding author.

cesar.espinr@ug.edu.ec (C. Espin-Riofrio); jenny.ortizz@ug.edu.ec (J. Ortiz-Zambrano); amontejo@ujaen.es (A. Montejo-Ráez)

0000-0001-8864-756X (C. Espin-Riofrio); 0000-0001-6708-4470 (J. Ortiz-Zambrano); 0000-0002-8643-2714 (A. Montejo-Ráez)

manner where weights are, themselves, learned from a training dataset of human and non-human labeled texts. Some of the experiments overcome showed that the approach is very promising, but the large differences in style and domain between training a testing datasets turned into a poor performance of our proposal.

The paper is organized as follows: first, the method implemented is detailed. Then, experiments done to evaluate our approach over the training data provided by organizers are reported. Finally, results and conclusions drawn from the official scores are given.

2. Related work

The evolution of automatic text generation models has made great advances, from rule-based systems to sophisticated deep learning models. Early text generation systems were based on predefined rules and templates (rule-based systems). Markov models [3] introduced a probabilistic approach to text generation, using statistical patterns to predict the probability of a particular word or phrase based on prior context [4]. Recurrent Neural Networks (RNN) enabled the generation of more coherent and contextually relevant texts by solving the problem of long-range dependencies. LSTM (Long Short-Term Memory) [5] were able to learn and retain long-term dependencies more effectively, which improved the quality of text generation. GANs (Generative Adversarial Networks) [6] have shown impressive results in generating coherent and diverse text, although they can be difficult to train. The introduction of the Transformer models [7] marked an important milestone in text generation, starting with the Generative Pre-trained Transformer (GPT) [8] model, followed by GPT-2 [9], GPT-3 [10] and later iterations. Recent advances have explored the integration of Reinforcement Learning (RL) techniques in text generation models.

Regarding the use of Transformer models for text classification, [11] uses them together with other classifiers to determine the political affinity of Ecuadorian Twitter users. [12] combine Transformer embeddings with linguistic features to identify complex words. [13] use stylometric characteristics together with a transformer model to determine the gender and profession of Twitter users.

The Turing Test [14] attempted for the first time to detect whether a text came from a robot or a human. [15] develop GLTR, a tool to support humans in detecting whether a text was generated by a model, applies a suite of baseline statistical methods that can detect generation artifacts across common sampling schemes.

In recent studies, [16] study the differences in the ability of humans and automated detectors to identify text generated by TGM. Authors such [17], have used the contextual word embeddings of a BERT model to calculate a quality score for the generated text. [18] explore the originality of content produced by ChatGPT, their results show that ChatGPT has great potential to generate outstanding text output without being well detected by plagiarism checking software.

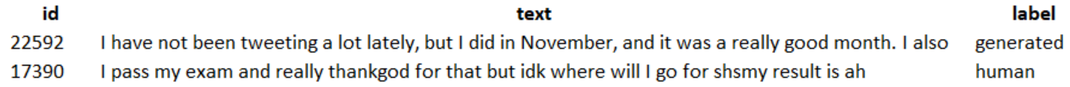
3. Method

In this section we present the dataset and the method used in the experimentation. We tokenize the texts and obtain the embeddings of the start-of-sequence token for each of the 12 layers of

BERT-based models. We stacked the embeddings, and linearly combined all layers. Then, we implemented a Feed Forward Neural (FFN) network as a classification layer, with two linear functions connected by a dropout and an activation function.

3.1. Data

The organizers provided the datasets [19] for the AuTexTification shared task at IberLEF 2023, which include the labeled training and test portions for all proposed subtasks and languages. They considered five different domains including legal documents, how-to articles and social media, to cover a wide variety of writing styles: from the more structured and formal to the less structured and informal. The dataset corresponds to subtask 1: Human or Generated, their structure is shown in Fig. 1 and their sizes in Table 1.



id	text	label
22592	I have not been tweeting a lot lately, but I did in November, and it was a really good month. I also	generated
17390	I pass my exam and really thankgod for that but idk where will I go for shsmy result is ah	human

Figure 1: Sample training dataset with tags.

Table 1
Size of training and test datasets for each language.

	Train_data	Test_data
Spanih	32062	20129
English	33845	21832

3.2. Transformer model for text classification

Text classification is a classic Natural Language Processing (NLP) problem, consisting of assigning predefined categories to a given text [20]. Transformer models are being widely used in many NLP tasks, with very good results. The Transformer architecture is especially conducive to pre-training on large text corpora, allowing for higher accuracy in tasks such as text classification [21].

BERT-based models take an input from a sequence of 512 tokens and outputs the sequence representation. The sequence has one or two segments, The first token of every sequence is always a special classification token [CLS], which contains the special sorting embedding. The final hidden state corresponding to this token is used as the aggregate sequence representation for classification tasks.

Usually, most of the fine-tuning for text classification tasks, take the final hidden layer as the whole text representation and later, pass it to other models for the subsequent task taking the [CLS] token of each sentence. On top, a simple softmax classifier is added to predict the label probability. A BERT-based model contains an encoder with 12 Transformer blocks, 12 self-attention heads, and the hidden size of 768.

We experiment with sequence start token embeddings for each of the 12 layers of BERT-based models.

We have used the DeBERTa (Decoding-enhanced BERT with disentangled attention) [22] model to classify texts in English, and mDeBERTa [23] for texts in Spanish. DeBERTa enhances state-of-the-art pre-trained language models using two novel techniques: an untangled attention mechanism and an improved mask decoder. mDeBERTa is the multilingual version of DeBERTa which use the same structure as DeBERTa and was trained with CC100 multilingual data [24].

3.3. Model tuning

Fine-tuning a Bert-based model, can bring many advantages in classification tasks. It usually involves adding a specific layer on top of the BERT encoder and training the entire end-to-end model with a suitable loss function and optimizer.

For our purposes, we split the datasets into 80% for training and 20% for testing, both for the Spanish and English datasets.

We stacked the sequence start token embeddings from each of the 12 layers of the BERT-based models, combined them linearly, and generated a weighted average. Then, we implemented a FFN as a classification layer, with two linear functions connected by a dropout and an activation function. We use DeBERTa for English texts and mDeBERTa for Spanish texts.

To determine the values for the dropout rate, activation function, and learning rate parameters, we conducted several training experiments by individually varying each parameter across different values. We recorded the value for which the model achieved the best evaluation metric during each experiment. Table 2 shows the values of the hyperparameters explored and, in Table 3, we detail the hyperparameters chosen for the final training of the model, using DeBERTa for English and mDeBERTa for Spanish.

Table 2
Hyperparameters values tested

learning rates	dropout	activation funtion
1e-5	0,1	relu
5e-4	0,2	tanh
5e-5	0,3	gelu
	0,5	

Table 3
Hyperparameters for training

Parameters	DeBERTa	mDeBERTa
drop out	0,3	0,2
activation function	relu	tanh
epoch	3	2
learning rate	5e-5	5e-5
batch_size	16	16

Tables 4 y 5 show the results obtained on each evaluation set, according to the models used for English and Spanish.

We save the fine-tuned model, which we then use with the test dataset to make the prediction.

Table 4
DeBERTa's evaluation metrics for English

Epoch	Training Loss	F1-macro	Accuracy	Precision Macro	Precision Weighted
1	0,2639	0,8941	0,8948	0,9032	0,9024
2	0,1580	0,9182	0,9185	0,9211	0,9207
3	**0,0658**	**0,9232**	**0,9233**	**0,9243**	**0,9241**
4	0,0384	0,9090	0,9094	0,9154	0,9148
5	0,0153	0,9047	0,9053	0,9130	0,9122
6	0,0047	0,9076	0,9081	0,9149	0,9142
7	0,0051	0,9164	0,9167	0,9211	0,9205

Table 5
mDeBERTa's evaluation metrics for Spanish

Epoch	Training Loss	F1-macro	Accuracy	Precision Macro	Precision Weighted
1	0,3110	0,8311	0,8358	0,8693	0,8673
2	**0,2112**	**0,9281**	**0,9281**	**0,9284**	**0,9282**
3	0,1374	0,8749	0,8768	0,8953	0,8938
4	0,0749	0,9071	0,9077	0,9148	0,9139
5	0,0489	0,9129	0,9133	0,9185	0,9177
6	0,0227	0,9037	0,9044	0,9126	0,9116
7	0,0111	0,8880	0,8893	0,9035	0,9022

4. Experiment

We proceed to extract the embeddings from the test dataset. We load the fine-tuned model and execute its prediction function by inputting the obtained embeddings. This way, we obtain the predicted tags to be submitted for the task evaluation.

In Tables 6 and 7, we show the results of our participation in AuTexTification subtask 1, for English and Spanish texts, respectively.

Table 6
Participation results in AuTexTification subtask 1 for English texts

Rank	Team	Run	Macro-F1	Confidence Interval
1	TALN-UPF	Hybrid_plus	80,91	(80.4, 81.38)
2	TALN-UPF	Hybrid	74,16	(73.68, 74.7)
3	CIC-IPN-CsCog	run2	74,13	(73.53, 74.72)
…				
47	**SINAI**	**run1**	**57,78**	**(57.24, 58.45)**
…				
74	penguinz	run1	33,89	(33.62, 34.16)
75	UAEMex	run3	33,87	(33.6, 34.17)
76	UAEMex	run1	33,87	(33.6, 34.11)

Table 7
Participation results in AuTexTification subtask 1 for Spanish texts

Rank	Team	Run	Macro-F1	Confidence Interval
1	TALN-UPF	Hybrid_plus	70,77	(70.21, 71.35)
2	Linguistica_F-P_et_al	run1	70,6	(69.85, 71.18)
3	RoBERTa (BNE)	baseline	68,52	-
...				
40	**SINAI**	**run1**	**54,95**	**(54.31, 55.53)**
...				
51	LKE_BUAP	run2	33,02	(32.57, 33.53)
52	LKE_BUAP	run3	31,6	(31.24, 31.99)

5. Conclusions and future work

The different layers of BERT capture different levels of semantic and syntactic information. We experimented with the [CLS] token for each of the 12 layers of the BERT-based models.

In our final presentation, we showcased the prediction systems for English and Spanish datasets using the DeBERTa and mBERTa models, respectively. The best performance for English texts was achieved in epoch 3 with an F1 score of 0.9232. For Spanish texts, the best performance was obtained in epoch 2 with an F1 score of 0.9281. All the details of the AuTexTification workshop, including information about the different participants and the obtained results, can be found in the official overview [25].

Regarding our participation in AuTexTification subtask 1, we ranked 47th out of 76 participants for English texts with an f1-macro of 0.5778, and 40th out of 52 participants for Spanish texts with f1-macro of 05495. We did not achieve good results in prediction, considering that domain generalization was one of the key aspects of Subtask 1.

This participation has helped us to propose a system in which we expect the style to be somewhat encoded throughout all the layers of the Transformer but, it seems that we still do not obtain results that allow us to reach adequate conclusions. It seems that the method we propose does not represent an improvement and presents problems of application to the data set. Our model demonstrates good prediction performance within the domain it was trained on, but it does not effectively generalize to other unseen domains. As future work, we propose to add stylometric features of the text to the embedding of the tokens obtained from the text, in order to reinforce the learning of the model.

We will continue working now that we have the ground truth tags from the test, we plan to perform further experiments to see if this method has really contributed to the proposed task.

Acknowledgments

This work has been partially supported by Big Hug project (P20_00956, PAIDI 2020) and WeLee project (1380939, FEDER Andalucía 2014-2020) funded by the Andalusian Regional Government, and LIVING-LANG project (RTI2018-094653-B-C21) funded by MCIN/AEI/10.13039/501100011033.

References

[1] D. I. Adelani, H. Mai, F. Fang, H. H. Nguyen, J. Yamagishi, I. Echizen, Generating sentiment-preserving fake online reviews using neural language models and their human- and machine-based detection, in: Advanced Information Networking and Applications: Proceedings of the 34th International Conference on Advanced Information Networking and Applications (AINA-2020), Springer, 2020, pp. 1341–1354.

[2] 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.

[3] O. Ibe, Markov processes for stochastic modeling, Newnes, 2013.

[4] O. C. Ibe, 14 - hidden markov models, in: O. C. Ibe (Ed.), Markov Processes for Stochastic Modeling (Second Edition), second edition ed., Elsevier, Oxford, 2013, pp. 417–451. URL: https://www.sciencedirect.com/science/article/pii/B9780124077959000141. doi:https://doi.org/10.1016/B978-0-12-407795-9.00014-1.

[5] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (1997) 1735–1780.

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks, Communications of the ACM 63 (2020) 139–144.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).

[8] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, et al., Improving language understanding by generative pre-training (2018).

[9] 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.

[10] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, Advances in neural information processing systems 33 (2020) 1877–1901.

[11] C. Espin-Riofrio, J. L. Charco, J. Z. Gamboa, V. M. Morán, A. Montejo-Ráez, B. D. Campoverde, I. En Sistemas, Y. Gilson, T. Molina, Determination of political affinity of ecuadorian twitter users using machine learning techniques for authorship attribution (????). doi:10.18687/LACCEI2022.1.1.535.

[12] J. A. Ortiz-Zambrano, C. Espin-Riofrio, A. Montejo-Ráez, Combining transformer embeddings with linguistic features for complex word identification, Electronics 12 (2022) 120.

[13] C. Espin-Riofrio, M. Pazmiño-Rosales, C. Aucapiña-Camas, V. Mendoza-Morán, A. Montejo-Ráez, Spanish stylometric features to determine gender and profession of ecuadorian twitter users, in: Smart Technologies, Systems and Applications: 3rd International Conference, SmartTech-IC 2022, Cuenca, Ecuador, November 16–18, 2022, Revised Selected Papers, Springer, 2023, pp. 161–172.

[14] A. M. Turing, Computing machinery and intelligence, Mind 59 (1950) 433–60. doi:10.1093/mind/lix.236.433.

[15] S. Gehrmann, H. Strobelt, A. M. Rush, Gltr: Statistical detection and visualization of generated text, arXiv preprint arXiv:1906.04043 (2019).

[16] D. Ippolito, D. Duckworth, C. Callison-Burch, D. Eck, Automatic detection of generated text is easiest when humans are fooled, arXiv preprint arXiv:1911.00650 (2019).

[17] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi, Bertscore: Evaluating text generation with bert, arXiv preprint arXiv:1904.09675 (2019).

[18] M. Khalil, E. Er, Will chatgpt get you caught? rethinking of plagiarism detection, arXiv preprint arXiv:2302.04335 (2023).

[19] A. Sarvazyan, J. Ángel González, M. Franco, F. M. Rangel, M. A. Chulvi, P. Rosso, Autextification dataset (full data), 2023. URL: https://doi.org/10.5281/zenodo.7956207. doi:10.5281/zenodo.7956207.

[20] C. Sun, X. Qiu, Y. Xu, X. Huang, How to fine-tune bert for text classification?, in: Chinese Computational Linguistics: 18th China National Conference, CCL 2019, Kunming, China, October 18–20, 2019, Proceedings 18, Springer, 2019, pp. 194–206.

[21] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, X. Le Q, generalized autoregressive pretraining for language understanding. arxiv 2019; 1906.08237, 1906.

[22] P. He, X. Liu, J. Gao, W. Chen, Deberta: Decoding-enhanced bert with disentangled attention, arXiv preprint arXiv:2006.03654 (2020).

[23] P. He, J. Gao, W. Chen, Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing, arXiv preprint arXiv:2111.09543 (2021).

[24] G. Wenzek, M.-A. Lachaux, A. Conneau, V. Chaudhary, F. Guzmán, A. Joulin, E. Grave, Ccnet: Extracting high quality monolingual datasets from web crawl data, arXiv preprint arXiv:1911.00359 (2019).

[25] 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).