FI Group at SemEval-2024 Task 8: A Syntactically Motivated Architecture for Multilingual Machine-Generated Text Detection

Maha Ben-Fares[1,2], Urchade Zaratiana[2,3], Simon D. Hernandez [2] and Pierre Holat[2,3]

ETIS, CY Cergy Paris Université - Pontoise, France[1]; FI Group, Puteaux, France[2]
LIPN, Université Sorbonne Paris Nord, Villetaneuse, France[3]
maha.ben-fares@cyu.fr, zaratiana@lipn.fr, {simon.hernandez, pierre.holat}@fi-group.com

Abstract

In this paper, we present the description of our proposed system for Subtask A - multilingual track at SemEval-2024 Task 8, which aims to classify if text has been generated by an AI or Human. Our approach treats binary text classification as token-level prediction, with the final classification being the average of token-level predictions. Through the use of rich representations of pre-trained transformers, our model is trained to selectively aggregate information from across different layers to score individual tokens, given that each layer may contain distinct information. Notably, our model demonstrates competitive performance on the test dataset, achieving an accuracy score of 95.8%. Furthermore, it secures the 2nd position in the multilingual track of Subtask A, with a mere 0.1% behind the leading system.

1 Introduction

The evolution and widespread adoption of Generative Pre-trained Transformers, notably with the release of ChatGPT have significantly influenced the landscape of digital communication and content creation. While these advancements herald a new era of efficiency and creativity, enabling applications ranging from sophisticated writing aids to advanced conversational agents, they simultaneously introduce significant challenges and ethical concerns. In fact, the proliferation of AI-generated texts has raised alarm over issues like the dissemination of misinformation, the facilitation of academic fraud, and the potential erosion of trust in digital content. This underscores the urgent requirement for robust solutions to identify AI-generated content, safeguarding the integrity of information while embracing the benefits of AI advancements.

In this paper, we aim to develop a reliable detection system by participating in the SemEval Task 8 on Machine-Generated Text Detection. This task is notable for its complexity, as it involves Multi-generator, Multidomain, and Multilingual text, making it a highly challenging endeavor. Furthermore, the evaluation is conducted on unseen domains and languages, establishing it as a robust benchmark for evaluating AI text detectors. This requires the model to effectively generalize across different domains and languages. We focus our efforts on the binary detection, which aims to determine whether a text has been generated by an AI or not. To tackle this challenge, we propose a syntactically motivated architecture. Our approach is primarily inspired by the realization that texts generated by AI and humans are semantically similar, as they are derived from comparable topical distributions. Hence, we argue that the distinction between them lies in their syntax and writing style.

Typically, transformer-based text classification relies on information from the last layer for classification. However, our model takes a different approach by dynamically aggregating information from all layers of the transformer (a.k. *multi-layer fusion* Shi et al., 2022). This method is intentionally designed to harness the diverse linguistic information present at various levels of the transformer, as noted in previous studies (Peters et al., 2018; Jawahar et al., 2019; Tenney et al., 2019). These studies reveal the uneven distribution of linguistic features across the transformer's architecture, with syntactic details predominantly in the initial layers and complex semantic information in the deeper layers. By utilizing insights from all layers, our model aims to capture the entire range of linguistic cues, enhancing its capability to accurately differentiate between human and AI-generated content. Additionally, our model moves beyond the standard practice of using just the [CLS] token for classification in BERT-based classifiers. It applies sequence labeling to classify each token in the text as either Human or AI. We believe that

this approach enables the capture of more complex phrasal structures, which helps in more effectively distinguishing the style and syntax of a text.

Our proposed model obtains competitive performance on the test leaderboard of the shared task subtask A, securing the 2nd best position on binary multilingual detection, using a much smaller model than other approaches often using finetuned LLMs.

2 Related Works

Since the introduction of large-scale pre-trained models like GPT-3, capable of generating high-quality text, the detection of machine-generated text has attracted considerable interest. The most common and straightforward strategy for addressing this task involves training models on a labeled dataset comprising both human and AI-generated text. This approach is utilized by well-known models such as the OpenAI ai text detector and commercial models such as GPTZero (Tian and Cui, 2023). While these models achieve strong in-domain results, they often require labeled datasets from a wide range of sources and domains to achieve generalization. An alternative approach involves zero-shot detectors, which do not necessitate any model training. For example, DNA-GPT (Yang et al., 2024) assesses N-Gram divergence between the continuation distribution of re-prompted text and the original text for making predictions, while Detect-GPT (Mitchell et al., 2023) employs a curvature-based criterion to determine if a passage is generated by a specific LLM.

3 Preliminary study

In this section, we detail a preliminary study that provided essential insights, guiding us towards our final model design.

Motivation Our aim was to assess the efficacy of semantic embeddings, particularly sentence-BERT, in differentiating between machine-generated and human-authored texts. Figure 1 illustrates the embeddings of texts from both humans and various language models, visualized using sentence-BERT embeddings (Reimers and Gurevych, 2019) and UMAP for dimensionality reduction (McInnes et al., 1802).

Analysis The visualization in Figure 1 reveals that texts generated by humans and various language models occupy similar positions in the latent semantic space, with data points from different sources blending together, lacking distinct separation. Given the limited utility of semantic features for discriminating human and ai-generated text, we argue that the key to distinguishing between these texts may lie at the syntactic level.

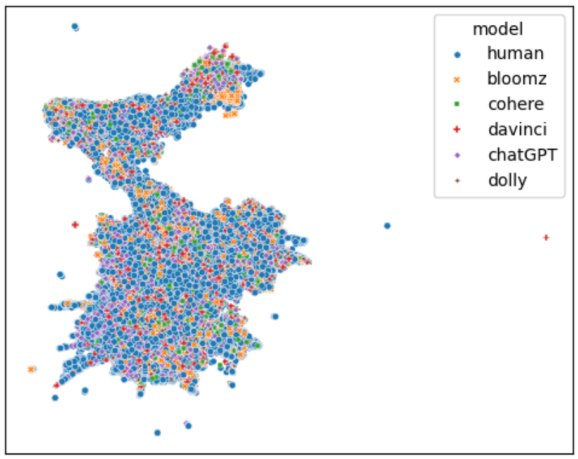

Figure 1: Visualization of UMAP-projected Sentence-BERT embedding of documents generated by human and different large language models

Model Motivated by our analysis, we propose a text classification model that take into account syntactic information. More specifically, our approach introduces two main innovations: 1) the integration of information from all layers of the transformer for classification, referred to as *layer fusion* (Shi et al., 2022). This method leverages the rich linguistic information embedded across the transformer's layers to compute classification scores (Peters et al., 2018; Tenney et al., 2019; Jawahar et al., 2019). 2) The usage of sequence labeling for text classification, which could enhance the model's ability to capture complex phrasal structures, potentially improving its ability to differentiate texts based on style and syntax.

4 Architecture

In this section, we provide a detailed overview of our proposed model's architecture illustrated in Figure 2.

4.1 Representation

Given a text input $X = \{x_1, \ldots, x_N\}$, the model first computes embeddings for each word using a multi-layer pre-trained transformer encoder such as BERT:

$$H = \texttt{transformer}(X) \in \mathbb{R}^{N \times L \times D} \quad (1)$$

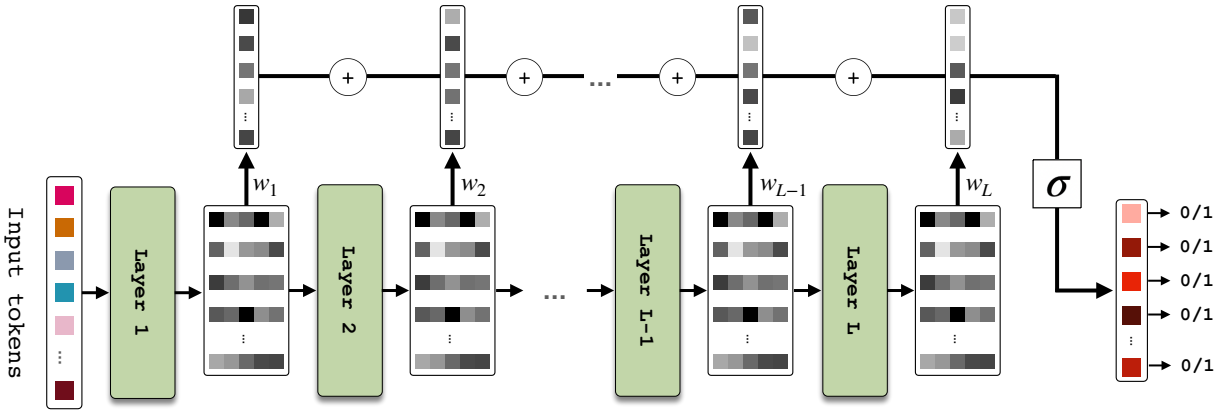

Figure 2: Architecture of Our Proposed Model. A pre-trained transformer receives a sequence of tokens as input and generates token embeddings at each layer. Token scores are computed for each layer, and the final score for each token is derived from the sum of scores across all layers. The probability of each token being AI-generated is determined by applying a sigmoid activation function to its score.

Here, N is the number of words in the input, L is the number of transformer layers, and D is the model dimension.

4.2 Scoring

The model then computes a score for each word, integrating information across all transformer layers, similar to the proposed *multi-layer fusion* by Shi et al. (2022). The score s_i for a word at position i is computed as follows:

$$s_i = \sum_{l=1}^{L} \boldsymbol{w}_l^\top \boldsymbol{h}_i^l \in \mathbb{R} \tag{2}$$

In this equation, $\boldsymbol{h}_i^l \in \mathbb{R}^D$ represents the embedding of the i-th word at the l-th layer. $\boldsymbol{w}_l \in \mathbb{R}^D$ is a learned weight vector specific to layer l. This scoring mechanism allows the model to weigh the contributions of different layers differently for each token, potentially emphasizing certain linguistic features over others.

4.3 Classification

For the classification, we employ a sequence labeling approach, where each word is classified based on its computed score. For this, a sigmoid function is applied to convert the token scores into probabilities, and a threshold is used to make a binary decision:

$$y_i = \begin{cases} 1 & \text{if } \sigma(s_i) > 0.5, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

This step results in a binary classification for each word, indicating its belonging to the positive class. Finally, the classification of the entire sentence is determined by averaging these binary decisions:

$$y = \frac{1}{N} \sum_{i=1}^{N} y_i \tag{4}$$

This average represents the probability of the sentence belonging to the positive class, synthesizing the word-level classifications into an overall sentence-level prediction. Finally, given an input X, we consider it as being machine-generated if its computed probability is superior to 0.5, i.e $y > 0.5$.

4.4 Training

To train our model, we focus on maximizing the likelihood of the correct label for each token by minimizing the binary cross-entropy loss at the token level. The binary cross-entropy loss for an input text of length N can be formulated as follows:

$$\mathcal{L} = -\sum_{i=1}^{N} \left(y^* \log(p_i) + (1 - y^*) \log(1 - p_i)\right) \tag{5}$$

Here, y^* represents the true label of the input (1 for human-generated and 0 for AI-generated), p_i denotes the predicted probability of the i-th token being human-generated (computed by applying the sigmoid function to the score s_i).

5 Experimental setup

5.1 Data

For our experiments, we used the dataset provided at SemEval-2024 Task 8, more details can be found in the task description (Wang et al., 2024a). It is

based on the benchmark M4 dataset (Wang et al., 2024b), which is a large-scale multi-generator, Multi-domain, and Multi-lingual corpus containing human-written and machine-generated texts. The machine-generated texts were produced by prompting several LLMs, including ChatGPT, textdavinci-003, Cohere, Dolly-v2 and BLOOMz from different sources such as Wikipedia, WikiHow, Reddit, arXiv, PeerRead for English, Baike and Web question answering for Chinese, news for Urdu, RuATD for Bulgarian and news for Indonesian.

5.2 Hyperparameters

In our experiments, we utilized the `xlm-roberta-large` model as the backbone for our architecture. The model was trained with a batch size of 12 across a maximum of 2 epochs, as we found that training further harms the validation results. More specifically, we observed that while training longer always improves in-domain performance measured on a held-out subset of the training set, it harms performance on out-of-domain validation (Kumar et al., 2022). We hypothesize this is due to overfitting on in-domain data, making long training harms the generalization of the model. Due to this, we evaluated our model on the out-of-domain validation set every 500 gradient steps and kept the best-performing model for testing. We employed different learning rates for the backbone (pre-trained transformer model) parameters and the added projection parameters: the learning rate for the backbone was set to `1e-5`, and the learning rate for the projection weights (randomly initialized) was set higher at `3e-4`. This distinction allows for delicate fine-tuning of the pre-trained model (to not distort the pre-trained representation too much), while more aggressively updating the newly introduced parameters to adapt to the task-specific features. During training, we use a maximum sequence length of 128 subwords to allow faster training, but we compute test set prediction using the maximum size of 512 tokens. The experiments were conducted with a runtime limit of 2 hours and 30 minutes for each experiment, utilizing an NVIDIA V100 GPU.

5.3 Other approaches

In this section, we provide an overview of the methodologies adopted by participants based on their description[1] in the shared task (Wang et al., 2024a). The baseline approach involved fine-tuning an XLM-Roberta-base model specifically for this task. The team *USTC-BUPT* presented the top-performing system, where English texts were processed using the Llama-2-70b model to generate average embeddings. These embeddings were then classified using a two-stage CNN. For texts in languages other than English, they treated classification as a next-token prediction task utilizing the mT5 model. Another notable participant, the *KInIT* team, employed an ensemble strategy that combined fine-tuned large language models (LLMs), including Mistart and Falcon, with zero-shot statistical methods to improve performance. Lastly, the *L3i++* team opted to fine-tune a LLaMA-2–7b model for the task. In comparison to the top-performing participants, only our approach uses small-scale transformer models.

[1]Note that we do not have access to entire articles.

Rank	Team	Accuracy (%)
1	USTC-BUPT	95.9
2	FI Group (*Ours*)	95.8
3	KInIT	95.0
4	priyansk	93.8
5	L3i++	92.9
–	*Baseline*	80.9

Table 1: Test leaderboard results.

6 Results

6.1 Test leaderboard results

Table 1 shows the top 5 scores from the leaderboard obtained using the test dataset, which includes domains and languages never seen during training.

Our team achieved the second-highest score, with an accuracy of 95.8%, narrowly trailing the top system by only 0.1%. There were 69 participants in the multilingual track in subtask A, out of a total of 159 participants across all SemEval-2024 Task 8.

6.2 Ablation study

In this section, we conduct an ablation study to examine the impact of various components of our model, including the backbone, layer fusion, and sequence labeling. The outcomes of this analysis are reported in Table 2.

Results Regarding the backbone, our findings indicate that `XLM-R-large` achieves better per-

Model	Accuracy (%)	F1 (%)
Ours (`XLM-R-base`)	87.3	87.1
Ours (`XLM-R-large`)	87.6	87.5
- w/o sequence labeling	81.2	81.1
- w/o layer fusion	78.1	77.4
Baseline (`XLM-R-base`)	75.0	–

Table 2: Ablation performance on the validation set. We perform ablation of our proposed model to see the influence of sequence labeling and layer fusion.

formance than `XLM-R-base`, suggesting that our method scales effectively. Moreover, our analysis reveals that both sequence labeling and layer fusion significantly contribute to the model's performance. Specifically, omitting sequence labeling—which involves aggregating the scores of the CLS token across layers—results in a 6-point decrease in accuracy. Similarly, excluding layer fusion leads to a more pronounced decline, with over a 10-point drop in F1 score and a 9-point decrease in accuracy. These findings underscore the critical roles that token-level prediction and layer fusion play in enhancing the overall effectiveness of our model.

6.3 Learned Weight Analysis

Motivation Figure 3 visualizes the norm of the learned weight vector for each layer of our model, denoted as w_l in equation 2. We hypothesize that the magnitude of these projection weights reflects the significance of each layer in contributing to the final prediction, with higher weights suggesting a more substantial influence on the token scores.

Analysis The Figure 3 indicates that layer 0, the embedding layer, has the lowest norm value. Given that this layer does not incorporate contextual information, its minimal contribution suggests that mere word appearance is insufficient for determining whether a text is produced by a human or an AI, aligning with the findings of Gallé et al. (2021). Interestingly, layer 24, which is the final layer, also shows a relatively low norm value. This observation resonates with analyses indicating that the last layer tends to be rich in semantic content yet sparse in syntactic details. We believe this explains the lower norm value for the last layer, as semantic aspects alone are inadequate for distinguishing between human and AI writing. Conversely, the highest norm values are predominantly found in layers 3 to 6 and 20 to 22, suggesting these layers

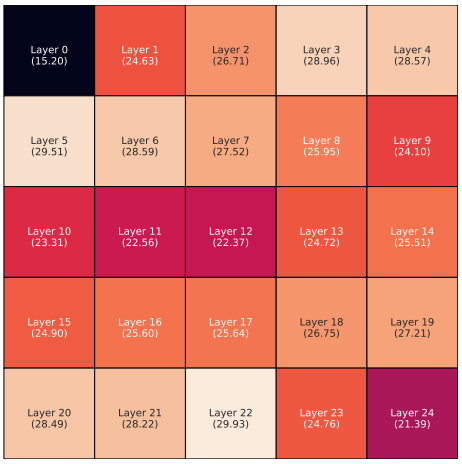

Figure 3: Norm (L1) of the weight vector w_l for each layer in our model.

play a pivotal role in the model's decision-making process.

7 Conclusion

In this paper, we presented our system submitted to SemEval-2024 Task 8 for detecting human-written and machine-generated text, achieving 2nd place for subtask A on multilingual texts. Our system relies on a hierarchical fusion strategy that adaptively fuses representations from transformer's layers, with a focus on syntactic rather than semantic information. By leveraging syntactic features, particularly through sequence labeling, we captured more phrasal structures of text, thereby enhancing our ability to distinguish text styles and syntax. Our system achieved robust performance across diverse unseen domains and languages, demonstrating its adaptability and generalization capability, notably considering that we used a smaller model compared to other proposed systems often reliant on fine-tuned LLMs.

References

Matthias Gallé, Jos Rozen, Germán Kruszewski, and Hady Elsahar. 2021. Unsupervised and distributional detection of machine-generated text. ArXiv: 2111.02878 [cs.CL].

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What Does BERT Learn about the Structure of Language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Ananya Kumar, Aditi Raghunathan, Robbie Matthew Jones, Tengyu Ma, and Percy Liang. 2022. Fine-

Tuning can Distort Pretrained Features and Underperform Out-of-Distribution. In *International Conference on Learning Representations*.

Leland McInnes, John Healy, and James Melville. 1802. Umap: Uniform manifold approximation and projection for dimension reduction. arXiv 2018. *arXiv preprint arXiv:1802.03426*.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. 2023. DetectGPT: Zero-Shot Machine-Generated Text Detection using Probability Curvature. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 24950–24962. PMLR.

Matthew E. Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018. Dissecting Contextual Word Embeddings: Architecture and Representation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509, Brussels, Belgium. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-Networks. In *Proceedings of the 2019 conference on empirical methods in natural language processing*. Association for Computational Linguistics.

Han Shi, JIAHUI GAO, Hang Xu, Xiaodan Liang, Zhenguo Li, Lingpeng Kong, Stephen M. S. Lee, and James Kwok. 2022. Revisiting over-smoothing in BERT from the perspective of graph. In *International conference on learning representations*.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT Rediscovers the Classical NLP Pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Edward Tian and Alexander Cui. 2023. GPTZero: Towards detection of AI-generated text using zero-shot and supervised methods.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, jinyan su, Artem Shelmanov, Akim Tsvigun, Osama Mohammed Afzal, Tarek Mahmoud, Giovanni Puccetti, Thomas Arnold, Chenxi Whitehouse, Alham Fikri Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024a. SemEval-2024 task 8: Multidomain, multimodel and multilingual machine-generated text detection. In *Proceedings of the 18th international workshop on semantic evaluation (SemEval-2024)*, pages 2041–2063, Mexico City, Mexico. Association for Computational Linguistics.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Chenxi Whitehouse, Osama Mohammed Afzal, Tarek Mahmoud, Toru Sasaki, Thomas Arnold, Alham Fikri Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024b. M4: Multi-generator, multi-domain, and multi-lingual black-box machine-generated text detection. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics*, Malta.

Xianjun Yang, Wei Cheng, Yue Wu, Linda Ruth Petzold, William Yang Wang, and Haifeng Chen. 2024. DNA-GPT: Divergent N-Gram Analysis for Training-Free Detection of GPT-Generated Text. In *The Twelfth International Conference on Learning Representations*.