SINAI at SemEval-2024 Task 8: Fine-tuning on Words and Perplexity as Features for Detecting Machine Written Text

Alberto J. Gutiérrez-Megías, L. Alfonso Ureña-López, Eugenio Martínez-Cámara
SINAI Research Group, Advanced Studies Center in ICT (CEATIC)
University of Jaén, Spain

Abstract

This work describes the system submitted by the SINAI team to the subtask A of Task 8 of SemEval 2024, as well as two additional systems evaluated during the training phase of the shared task. We claim that the perplexity score of a text may be used as a classification signal. Accordingly, we conduct a study on the utility of perplexity for discerning text authorship, and we perform a comparative analysis of the results obtained on the datasets of the task. The results of this study motivated us to use as classification features the word embeddings vectors of the input texts and its corresponding perplexity score. Likewise, the submitted system is a fine-tuning version of the XLM-RoBERTa-Large model. The analysis of the results of the evaluation shows large differences among the language probability distribution of the training and test sets. Nonetheless, the results show that perplexity can be used as feature for identifying machine generated text, hence our claim holds.

1 Introduction

In recent years, the use of generative models has increased considerably. The capabilities of this multifaceted tool include summarizing texts, retrieving information through searches, rephrasing texts for specific purposes and so on. However, it is important to recognize the potential threats associated with their application in certain contexts. For example, hallucinations in Natural Language Generation (NLG) models present significant problems, as they damage performance, raise safety issues for its use in the real world and hallucinations introduce privacy violation risks (Ji et al., 2023). Likewise, the very ability to generate natural language represents a threat, since it is increasingly indistinguishable from natural language. Therefore, the development of systems with the capacity to discern the authority of a given text, determining whether it is of human origin or generated by a generative model, is arising peremptory.

The language used by humans follows a probability distribution that differs so far from the distribution of the automated generated language (Rosenfeld et al., 1996). Perplexity measures the uncertainty value of a sample in a probability distribution. Accordingly, it is used in Natural Language Processing (NLP) as a metric to evaluate the effectiveness of linguistic models, for instance in text generation and machine translation tasks (Geluykens et al., 2021; Vaswani et al., 2018; Wang et al., 2019). Hence, it can be used to assess whether a text was generated by a machine, whose perplexity would be low, or written by a human, whose score would be large, since it may differ from that text automatically generated. We thus claim that perplexity can be used as a classification signal to enhance the finding of machine-generated texts (Meister and Cotterell, 2021).

In this work, we present the model submitted by the SINAI team to subtask A of task 8 of SemEval 2024 (Wang et al., 2024). Our proposal is based on the fuse of the word embeddings vectors stemmed from the fine-tuning of XLM-RoBERT-Large language model and the perplexity score of the input text. We use the Multimodal-Toolkit library (Gu and Budhkar, 2021) to fuse this two set of features.

After obtaining relevant results in the training phase and finding a clear difference between machine-generated and human-written texts, the results obtained have not been satisfactory. In part, this is due to the difference between the training and test datasets, which is analyzed later. Nevertheless, the exploration of the results obtained by merging textual content and associated perplexity raises the idea of using more novel linguistic models to calculate textual perplexity (see section 7).

The rest of the paper is organized as what follows: section 2 presents the works that support our proposal. Section 3 describes the data of the task, and section 4 is focused on the systems evaluated

Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), pages 1505–1510

and the submitted one. Section 5 presents the results reached during the training phase, and Section 6 the official reached ones as well as an analysis of them. We summarize the conclusions in Section 7. We release the source code at GitHub.[1]

2 Related Work

The language generation capacity of large language models is unceasing improving (Crothers et al., 2023). Hence, machine-generated text detection is key as a fundamental countermeasure to mitigate the misuse of NLG models, accompanied by notable technical challenges and a multitude of unresolved issues.

We find a wide range of strategies to differentiate human-written text from machine-generated text (Jawahar et al., 2020) from the most simple ones based on bag-of-words models to the latest ones grounded in the fine-tuning of linguistic models.

The paper (Mitrović et al., 2023) has shown that different observable patterns make up generative models of language, either grammatically or through the meaning of sentences. For example, perplexity is usually lower in texts generated by artificial intelligence and their texts rather express feelings and use unusual words. This paper also shows a difference in performance between perplexity-based and machine learning-based classification, the latter being better than perplexity-based classification. However, it shows the capacity of perplexity score to distinguish among natural language text and machine-generated text.

3 Data and Task Description

Task 8 is focused on the identification of machine-generated text. In this work, we manage the subtasks of monolingual (English) and multilingual classification.

The dataset for the monolingual English task consists of 119,757 training instances, complemented by another 5,000 evaluation instances (Wang et al., 2023). In the multilingual task, the corpora comprise a total of 172,417 instances, with an allocation of 4,000 instances for the evaluation phase. This multilingual dataset is composed of 77.48% English text, with Bulgarian as a secondary language. The rest of the training dataset also incorporates languages such as Chinese, Indonesian, and Urdu. In addition, the evaluation dataset includes texts in Russian, German, and Arabic.

Each instance includes the *text*, along with its corresponding *source* according to five categories: *Wikihow*, *Wikipedia*, *Reddit*, *Arxiv*, *Peerread*. In the multilingual task, we can find additional sources: *Bulgarian*, *Urdu*, *Indonesian*, and *Chinese*. Also has a category that attributes the text to a specific large language model: *ChatGPT*, *Cohere*, *Bloomz*, *Davinci*, *Dolly*, or *Human* in another case. The *gold label* is 1, if the text is machine-generated and 0 otherwise. The dataset presents an even distribution, with cases annotated as human or machine being approximately equal in the training and development corpora.

4 System Description

Our proposed system to subtask A of task 8 is based on the wide success of fine-tuning methods on language models and in our claim of using perplexity as a feature to separate texts written by humans from machine-generated texts. (Min et al., 2023).

We use the XLM-RoBERTa-Large base as a language model, and we first assess its performance by fine-tuning the training data on it. Then, we evaluate the use of the perplexity as a classification signal, and the third one, which we submit to the shared-task, is based on joint use the resulting features of the fine-tuning phase and the perplexity score of each sentence.

In the next subsections, we argue the use of perplexity as feature in Section 4.1, we present all the systems studied in Section 4.2 and we describe all the implementation details in Section 4.3.

4.1 Perplexity as Feature

According to (Mitrović et al., 2023), the perplexity of human-written text tends to be higher than the one of machine-generated text. We evaluate this assertion by calculating the perplexity of the documents from the training and development sets. Table 1 shows the perplexity of texts written by humans and machines. The results for monolingual and multilingual subtasks confirm a substantial gap in perplexity in both classes, which entails that perplexity can be used as a classification signal. We use the python Language Model Perplexity library (LM-PPL)[2] to calculate the perplexity. From all the large language models available to calculate the perplexity, we use GPT2 (Radford et al., 2019).

[1] https://github.com/sinai-uja/SemEval-2024-Task-8-Identification-of-machine-written-text/tree/main

[2] https://pypi.org/project/lmppl/

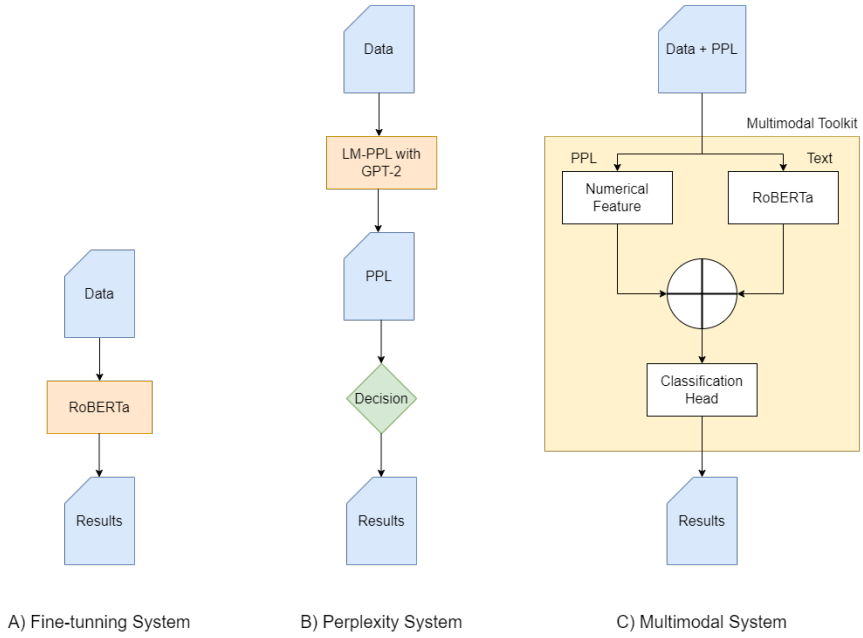

Figure 1: System diagrams used in developing phase.

Perplexity by classification	Mean
Human Monolingual	32.3613
Machine-Generated Monolingual	16.1494
Human Multilingual	35.8514
Machine-Generated Multilingual	20.9105

Table 1: Text perplexity for different classifications.

The results of Table 1 entails that the perplexity score can be use as classification signal, since it separates samples from the two classes. Accordingly, we propose a system based on the joint use of word embedding vectors and perplexity as feature.

4.2 Machine-Generated Text Detection systems

We have developed and evaluated three different systems. The first system is only based on fine tuning (system one), the second one only uses the perplexity score of the input sentences (system two) and the third one fuses the two set of features (system three). Figure 1 depicts the three systems.

System one based on fine-tuning and system two based on the use of perplexity as a classifier have been developed for the monolingual data only. Once the results of them have been obtained, the final system proposed for the task has been tested for the monolingual and multilingual subtasks.

System one - fine-tuning It is based on fine-tuning the XLM-RoBERTa-Large language model on the data of the task.

System two - perplexity For one of our systems only used the perplexity as a classifier, we established a threshold range based on the average perplexity of the dataset. The final choice of this strategy is to assign texts with perplexity at or below 20 as machine-generated, while those above this threshold are considered to be human-written.

Proposed system - fine-tuning and perplexity It is built upon two distinct features: word embedding vectors and the perplexity score of the input texts. The textual data is processed using the XLM-RoBERTa-Large transformer, and the result is sent to the combination module with the numerical features, in our case the perplexity.

The combination module uses the MultimodalToolkit library, and in particular the option[3] that separately encodes the two set of features and concatenate them before the final classification layer.

We submitted the classification results of our system based on fine-tuning and perplexity for the monolingual and multilingual subtasks.

4.3 Training and Implementation Details

We use Python to develop the proposed system and all the models evaluated during the development phase of the task. Likewise, we use the Transformers HuggingFace library (Wolf et al., 2020).

[3]Option name: *individual_mlps_on_cat_and_numerical_feats_then_concat*.

Optimized models	Epochs	Learning Rate	Weight Decay	Adam Epsilon	PP threshold
System one - fine-tuning	10	1.19e-05	6.18e-03	1.98e-07	-
System two - perplexity	-	-	-	-	20
Proposed system - Monolingual	10	6.89e-06	4.99e-02	1.13e-10	-
Proposed system - Multilingual	1	1.28e-05	8.67e-12	2.67e-07	-

Table 2: The values used for the hyperparameters of each model.

.

We optimize all the hyperparameters that drive the training of the models which involve transformers using the Optuna library (Akiba et al., 2019) following a grid search approach. This search has been performed using the English dataset, and once the optimized hyperparameters for each of the systems have been obtained, the same hyperparameters have been used for the multilingual task. For the sake of the reproducibility of the experiments, we describe the value exploration strategy of the values of the hyperparameters as what follows:

- Epochs [8, 16]: They represent the count of iterations required to traverse the entire training dataset for model training within a single cycle.
- Learning Rate [5e-6, 5e-5]: They govern the rate at which an algorithm updates or learns the parameter estimate values.
- Weight Decay [1e-12, 1e-1]: It constitutes a regularization technique that introduces a minor penalty term to the loss function.
- Adam Epsilon [1e-10, 1e-6]: It is a short positive value to forestall division by zero during the optimization process.

Table 2 shows the selected values, for the three systems that we evaluated during the development phase. We clarify that we independently optimized them, since they differ in their architecture (system one vs. proposed system) and the training objective (monolingual vs. multilingual).

5 Development results

Once we optimized the hyperparameters of each model, we assessed the performance of each system on the development data. We use accuracy as an evaluation measure, since it is the evaluation measure of the shared-task. Table 3 shows the results of this initial evaluation.

As detailed previously, the results of systems one and two are based on monolingual data, while the proposed system has been tested on both sets (see section 4.2).

System	Accuracy
System one - fine-tuning	0.8002
System two - Perplexity	0.6894
Proposed system - Monolingual	0.8698
Proposed system - Multilingual	0.6789
Task baseline monolingual	0.7400
Task baseline multilingual	0.7200

Table 3: Results in train phase with the dev. dataset.

As we indicated below, system one is only grounded in fine-tuning the XML-RoBERTa-Large model on the training data. The results are over the task baseline, which means that the incorporation of knowledge from the domain with the optimal values of the hyperparameters may reach competitive results and to overcome the task baseline.

The system that only uses perplexity as a classification signal reached poorer results than the baseline system. However, its performance is close to 70% accuracy, which means that the perplexity may be used as a feature to discriminate among human written text and machine-generated text, as we claim.

The proposed system jointly uses words and perplexity as features. In the monolingual scenario, this fusion of features reaches strong results far away from the task baseline. Nonetheless, the results for the multilingual scenario were not as strong as the monolingual one. According to the results of the monolingual data, we use as proposed system the one based on the fusion of words and perplexity as features.

6 Analysis and Discussion

The final results revealed unexpected differences compared to the results observed during the development phase. In the system that jointly uses

perplexity and text, the accuracy achieved in the monolingual task was 0.744631, which was lower than expected based on performance during development. In contrast, a poorer result was observed on the multilingual task, which achieved an accuracy of 0.801689 on the final test data set. Both results are lower than the baseline obtained by the competitors, with 0.8846 accuracy for the monolingual test and 0.8088 for the multilingual test.

Recognizing these discrepancies, our initial action consisted of an examination to determine the causes. The main advantage of our system resides in the incorporation of perplexity as classification signal along with the textual data. Consequently, our analysis primarily focused on examining the perplexity to elucidate possible factors contributing to the observed errors, as well as observing which class is more difficult to recognize, human-written or machine-generated texts.

Perplexity Performance Analysis The main limitation of our system is the use of perplexity. This metric depends on the characteristics of the reference large language model used for its calculation. Hence, we argue that there is a large disparity between the large language model used to generate the training and development sets and the documents of the test set.

We analyzed the perplexity of the documents of the test set, and we show them in Table 4. The results show a large discrepancy between the perplexity reached on the training dataset and the ones obtained on the test data. We stand out for the unexpectedly high perplexity of the machine-generated text, which is also over the human-written text. This is a sign that the large language model used to generate the documents of the test set may be more sophisticated than the one used to prepare the training dataset, or at least it was not the same large language model. This unexpected tendency to perplexity between the training and test data is behind the degradation of the performance of our proposed system on the test data, since the perplexity is a relevant feature of our porposed system.

We also explain the better performance of our proposed system in the multilingual subtask with the behavior of the perplexity on the test data. Although machine-generated text reaches again higher perplexity than human-written text, the difference is thin. Hence, the behavior of the perplexity is nearer to our claim, and the performance of our proposed system is thus stronger on multilingual data.

	Mean Perplexity	
	Human	M. Generated
Train Monolingual	32.3613	16.1494
Train Multilingual	35.8514	20.9105
Test Monolingual	35.8071	44.7824
Test Multilingual	58.4526	59.0258

Table 4: Mean perplexity in the test set for each task in comparison with the train datasets

Before generating the final results, we analyzed the prediction distribution to determine whether our system showed any tendency to predict a class in particular. In the monolingual tasks, we observed 17,978 instances of correct predictions, with only 22 false positives, with false positives being texts written by humans predicted to be machine-generated. Similarly, in the multilingual tasks, we found only 19 false negatives, the main finding was the occurrence of many false positives. Most of the predictions were obtained as machine-generated. We also highlight that the proposed system does not have any false negatives, which means that it is able to identify all the machine-generated text. However, since the disparity among the large language models to generate the training and test sets, we will keep working on reducing the false positives.

7 Conclusion

We have described the system submitted to subtask A of task 8 of SemEVAL. The system is grounded in the claim that perplexity may be a discriminant feature in identifying machine-generated texts. Hence, our submitted system is built upon the fine-tuning of a XML-RoBERTa-Large language model on a fusion of words and perplexity as features. The results reached during the development phase convinced us that our claim holds.

The official results show that the fusion of words and perplexity as features were not as good as the assessment on the development set. According to our analysis results, it may be caused by the use of a different large language model to generate the text documents. It pushes us to study the influence of the reference large language model used for the calculation of the perplexity and also to analyze the possibility of combining different perplexity calculated using a wide diverse set of large language models.

Acknowledgements

This work has been partially supported by projects CONSENSO (PID2021-122263OB-C21), MODERATES (TED2021-130145B-I00), SocialTOX (PDC2022-133146- C21) and FedDAP (PID2020-116118GA-I00) funded by MCIN/AEI/10.13039/501100011033 and by the "European Union NextGenerationEU/PRTR".

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *The 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2623–2631.

Evan Crothers, Nathalie Japkowicz, and Herna L Viktor. 2023. Machine-generated text: A comprehensive survey of threat models and detection methods. *IEEE Access*.

Joppe Geluykens, Sandra Mitrović, Carlos Eduardo Ortega Vázquez, Teodoro Laino, Alain Vaucher, and Jochen De Weerdt. 2021. Neural machine translation for conditional generation of novel procedures.

Ken Gu and Akshay Budhkar. 2021. A package for learning on tabular and text data with transformers. In *Proceedings of the Third Workshop on Multimodal Artificial Intelligence*, pages 69–73, Mexico City, Mexico. Association for Computational Linguistics.

Ganesh Jawahar, Muhammad Abdul-Mageed, and Laks VS Lakshmanan. 2020. Automatic detection of machine generated text: A critical survey. *arXiv preprint arXiv:2011.01314*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Clara Meister and Ryan Cotterell. 2021. Language model evaluation beyond perplexity. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5328–5339, Online. Association for Computational Linguistics.

Bonan Min, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth. 2023. Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Comput. Surv.*, 56(2).

Sandra Mitrović, Davide Andreoletti, and Omran Ayoub. 2023. Chatgpt or human? detect and explain. explaining decisions of machine learning model for detecting short chatgpt-generated text. *arXiv preprint arXiv:2301.13852*.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Ronald Rosenfeld et al. 1996. A maximum entropy approach to adaptive statistical language modelling. *Computer speech and language*, 10(3):187.

Ashish Vaswani, Samy Bengio, Eugene Brevdo, Francois Chollet, Aidan N Gomez, Stephan Gouws, Llion Jones, Łukasz Kaiser, Nal Kalchbrenner, Niki Parmar, et al. 2018. Tensor2tensor for neural machine translation. *arXiv preprint arXiv:1803.07416*.

Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F Wong, and Lidia S Chao. 2019. Learning deep transformer models for machine translation. *arXiv preprint arXiv:1906.01787*.

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. 2024. 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, Alham Fikri Aji, and Preslav Nakov. 2023. M4: Multi-generator, multi-domain, and multilingual black-box machine-generated text detection. *arXiv:2305.14902*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.