Team Advacheck at PAN: Multitasking Does All the Magic

Notebook for PAN at CLEF 2025

Anastasia Voznyuk[1,†], German Gritsai[1,2,†] and Andrey Grabovoy[1]

[1]Advacheck OÜ, Tallinn, Estonia
[2]Université Grenoble Alpes (UGA), Grenoble, France

Abstract

The paper describes a system designed by Advacheck team to recognise sophisticated compound machine-generated and human-written texts in two subtasks at the Voight-Kampff Generative AI Detection 2025 workshop organised as part of PAN 2025. Our developed system is a multi-task architecture with shared Transformer Encoder between several classification heads. As multiclass heads were trained to distinguish the domains presented in the data, they provide a better understanding of the samples. This approach led us to achieve 99.95% mean metric on validation set in Task 1 and the third place in the official ranking in Task 2 with 60.85% macro F_1-score on the test set and bypass the baseline by 13%.

Keywords

PAN 2025, Voight-Kampff Generative AI Detection, Human-AI Collaborative Text Classification, natural language processing, large language models, multi-task learning, domain adaptation

1. Introduction

As large language models are now firmly embedded in our world, helping researchers and internet users alike, we increasingly interact with machine-generated text. However, as more and more of the web becomes filled with generated content, the need for reliable detection methods grows. Potential misuse includes malicious applications by students [1, 2] or scientists [3, 4, 5]. The mentioned things are encouraging researchers to improve methods for detecting artificial text simultaneously with enhancing generation methods.

The task of detecting machine-generated texts is usually formulated as a binary text classification task [6]. The most common solutions are to fine-tune the Transformer-based model [7] or to use zero-shot approaches with intrinsic statistics of the text [8, 9]. While these methods perform well on in-domain tasks [10], they are not robust to change of the domain, generator model, or language of the texts [11, 12, 13]. Meanwhile, for the detection of AI-content in the wild such a change is, on the contrary, a more realistic setup. Additionally, the quality of available data can be low, introducing noise and making the detection task even harder [14]. The goal, then, is to build a model that can handle noisy inputs and generalize to new domains.

Because of the high coherence and fluency of modern LLMs, it is hard to find a simple, clear-cut feature that separates human-written from machine-generated text. One promising direction is to improve the representation space using multi-task learning (MTL) [15]. MTL has also shown strong results in past competitions [16, 17], which encouraged us to use it in our approach. In this paper we discuss our solution as the Advacheck team at Voight-Kampff Generative AI Detection 2025 [18]. Our method shows that with additional internal data analysis and embedding alignment using MTL, it is still possible to achieve high performance in detecting fragments in cross-domain and cross-generator setups on texts from the advanced LLMs. As we forced model to focus on various domains, it allowed us to form a cluster domain-wise structure for the text representations in the vector space. Additionally, we applied model to a setup where one needs to classify different types of human-AI mixed writing

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain

† Authors contributed equally

voznyuk@advacheck.com (A. Voznyuk); gritsai@advacheck.com (G. Gritsai); grabovoy@advacheck.com (A. Grabovoy)
0000-0002-4031-0025 (A. Grabovoy)

which can also be seen as "domains". In our research, we show that (1) multi-task learning outperforms the default single-task, (2) multi-task can perform well even for non-obvious domain setup such as different ways of mixed human-AI writing.

2. Tasks Definition

2.1. Task 1

The first subtask, *Voight-Kampff AI Detection Sensitivity* is binary AI detection task in that participants are given a text and have to decide whether it was machine-authored (class 1) or human-authored (class 0). The organizers presented new models and different ways of obfuscating texts using LLM tools such as style or mimicry. In addition, the data were augmented with new texts generated by participants in the ELOQUENT task [19]. The statistics of the dataset are summarised in Appendix A. The official evaluation metrics are the following:

- ROC-AUC,
- Brier, the complement of the Brier score (mean squared loss),
- C@1: A modified accuracy score that assigns non-answers (score = 0.5) the average accuracy of the remaining cases,
- F_1 measure,
- $F_{0.5u}$: a modified $F0.5$ measure (precision-weighted F-measure) that treats non-answers (score = 0.5) as false negatives,
- the arithmetic mean of all the metrics above.

2.2. Task 2

The second subtask, *Human-AI Collaborative Text Classification* is stated as multiclassification task, however, the classes are novel:

- **Fully human-written**: the document is entirely authored by a human without any AI assistance;
- **Human-initiated, then machine-continued**: a human starts writing, and an AI model completes the text;
- **Human-written, then machine-polished**: the text is initially written by a human but later refined or edited by an AI model;
- **Machine-written, then machine-humanized (obfuscated)**: an AI generates the text, which is later modified to obscure its machine origin;
- **Machine-written, then human-edited**: the content is generated by an AI but subsequently edited or refined by a human;
- **Deeply-mixed text**: the document contains interwoven sections written by both humans and AI, without a clear separation;

The official metric is Macro Recall, with additional metrics of accuracy and Macro F_1.

3. Multitask Learning

Similar to the previous competition on detection, COLING 2025, training set contained dozens of generators and several distinct domains. Moreover, organisers claimed there will be more domains and obfuscations in the test set. In the training data, we observed a large amount of valuable information such as the name of a particular model, genres, which can sharpen the representations of each text in the latent space and regularise the model. Therefore, following the best performed approach from COLING [17], we decided to utilise multitask learning once again, as multitask learning may help the model focus on those features that actually matter by shaping representations from all subtasks. Our

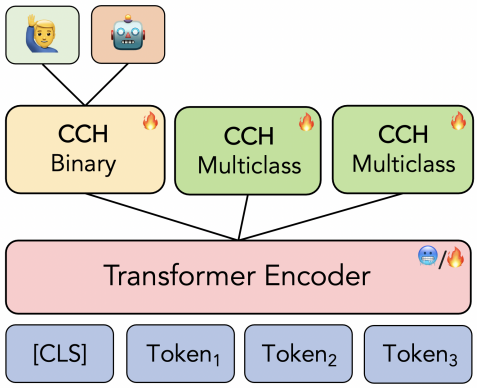

Figure 1: Overview of the proposed multi-task architecture. Modules marked only with 🔥 are trainable at all stages. The weights of the Transformer Encoder are frozen 🥶 at the first stage of training and trainable 🔥 at the second one. The Custom Classification Head (CCH) described in Appendix B is used for predictions.

final goal was to obtain fine-grained representations of the data that would ignore data-dependent noise and generalises well. Since different tasks involve distinct noises, a model trained on multiple tasks simultaneously is able to learn a more general representation. Furthermore, it reduces the risk of overfitting.

3.1. Model for Task 1

We propose an MTL architecture with hard parameter sharing (HPS), it is depicted in Figure 1. In HPS, a common Transformer-based encoder is used for multiple tasks. After several variations of set of parallel heads, we focused on three custom classification heads (CCH) for simultaneous training:

- Binary CCH for solving the initial task [2 classes]
- Multiclass CCH to define a genre [3 classes]
- Multiclass CCH determining model family belonging [4 classes]

In this setup, the data with genres for the second head were taken from the original data proposed by the authors, while the data for the third head passed a slight preprocessing. If we look at the list of models proposed by the organisers, we can cluster most of them into families. We performed such clustering by selected families of models, detailed description in Appendix A. These families did not include rows with all the models represented in the original dataset, due to the small amount in some of them. The idea of families should help to bring new insights into the representations, but not confuse the model too much, as would be the case if we set this head to the task of classifying into all available models without separating into families. Experiments with this sort of thing have been done as part of our previous work [17]. We chose DeBERTa-v3 base for the baseline and the backbone in our system, as it is currently state-of-the-art model for supervised fine-tuning for binary classification [20].

The model was trained in two phases: 1) fine-tuning assigned classifiers with frozen 🥶 shared encoder weights and 2) fine-tuning the complete model with all weights unfrozen 🔥. Therefore, at the first stage, only the weights of the classifiers are updated, while at the second stage, all the weights in the model are updated. These learning stages help to shift the distribution of the encoder weights in the right direction and avoid overfitting [21]. At the inference stage, only encoder with binary CCH predictions used for final classification.

3.2. Model for Task 2

Model for Task 2 was trained in the same way. We hypothesized that what really matters is who is the *initial* author of the text and clustered the data based on this. After that we came up with the following architecture:

- Multiclass CCH head for solving the initial task [6 classes];
- Multiclass CCH head for machine-based writing for classes *Machine-written, then machine-humanized* and *Machine-written, then human-edited* into 3 domains related to the datasets [3 classes];
- Multiclass CCH head for human-based writing for classes *fully human-written* and *human-written, then machine-polished* into 4 domains related to the datasets [4 classes];
- Multiclass CCH head for mixed writing for classes *deeply-mixed text* and *human-initiated, then machine-continued* [6 classes];

4. Results

In Table 1 we show the preliminary results obtained on the validation set for the Task 1, together with official baselines. In Table 2 we demonstrate the final results on the test set for the Task 2. Our approach took **3rd** place and performed much better than the suggested baseline.

Table 1
Overview of the various methods' performance on validation set on PAN 2025 (Voight-Kampff Generative AI Authorship Verification) Task 1. We report ROC-AUC, Brier, C@1, F_1, $F_{0.5u}$ and their mean. Results are obtained with TIRA [22].

Approach	ROC-AUC	Brier	C@1	F_1	$F_{0.5u}$	Mean
watery-bag	0.994	**0.996**	**0.996**	**0.997**	**0.995**	**0.995**
Baseline Binoculars	0.918	0.867	0.844	0.872	0.882	0.877
Baseline PPMd Compression-based Cosine	0.786	0.799	0.757	0.812	0.778	0.786
Baseline Linear SVM with TF-IDF features	**0.996**	0.951	0.984	0.980	0.981	0.978

Table 2
Overview of the various methods' performance on test set on PAN 2025 (Human-AI Collaborative Text Classification) Task 2. We report Recall (Macro), F_1 (Macro) and Accuracy.

Approach	Recall (Macro)	F_1 (Macro)	Accuracy
Advacheck (anastasiya.vozniuk)	0.6016	0.6085	0.6904
1st place on leaderboard	0.6446	0.6506	0.7409
Baseline RoBERTa	0.4832	0.4782	0.5709

5. Conclusion

In this paper we described the system by the Advacheck team for both tasks at Voight-Kampff Generative AI Detection 2025. We proposed solution with multi-task learning architecture that consists of shared Transformer Encoder and composition of one binary and two multiclass Custom Classification Heads. This approach led us to achieve 99.95% mean metric on validation set in Task 1 and outperformed all the baselines on the test set. The described multi-task way of learning also allowed us to take the third place in the official ranking in Task 2 with 60.85% macro F_1-score on the test set and bypass the baseline by 13%. Adding tasks for training in parallel reveal the formation of a cluster structure in the space of embeddings, helping to achieve high results despite the presence of a large amount of noisy data.

6. Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT and Writefull in order to check grammar and spelling and to paraphrase some sentences to improve clarity. After using this tool/service, the

author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] Z. Zeng, L. Sha, Y. Li, K. Yang, D. Gašević, G. Chen, Towards automatic boundary detection for human-ai collaborative hybrid essay in education, 2023. arXiv:2307.12267.

[2] R. Koike, M. Kaneko, N. Okazaki, Outfox: Llm-generated essay detection through in-context learning with adversarially generated examples, in: AAAI Conference on Artificial Intelligence, 2023.

[3] Y. Ma, J. Liu, F. Yi, Q. Cheng, Y. Huang, W. Lu, X. Liu, Ai vs. human – differentiation analysis of scientific content generation, 2023. arXiv:2301.10416.

[4] G. Gritsay, A. Grabovoy, A. Kildyakov, Y. Chekhovich, Artificially generated text fragments search in academic documents, in: Doklady Mathematics, volume 108, Springer, 2023, pp. S434–S442.

[5] K. Avetisyan, G. Gritsay, A. Grabovoy, Cross-lingual plagiarism detection: Two are better than one, Program. Comput. Softw. 49 (2023) 346–354. doi:10.1134/S0361768823040138.

[6] G. Jawahar, M. Abdul-Mageed, L. Lakshmanan, V.S., Automatic detection of machine generated text: A critical survey, in: D. Scott, N. Bel, C. Zong (Eds.), Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 2296–2309. doi:10.18653/v1/2020.coling-main.208.

[7] D. Macko, R. Moro, A. Uchendu, J. Lucas, M. Yamashita, M. Pikuliak, I. Srba, T. Le, D. Lee, J. Simko, M. Bielikova, MULTITuDE: Large-scale multilingual machine-generated text detection benchmark, in: H. Bouamor, J. Pino, K. Bali (Eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 9960–9987. URL: https://aclanthology.org/2023.emnlp-main.616/. doi:10.18653/v1/2023.emnlp-main.616.

[8] E. Mitchell, Y. Lee, A. Khazatsky, C. D. Manning, C. Finn, Detectgpt: Zero-shot machine-generated text detection using probability curvature, 2023. arXiv:2301.11305.

[9] A. Hans, A. Schwarzschild, V. Cherepanova, H. Kazemi, A. Saha, M. Goldblum, J. Geiping, T. Goldstein, Spotting llms with binoculars: Zero-shot detection of machine-generated text, 2024. arXiv:2401.12070.

[10] A. Uchendu, Z. Ma, T. Le, R. Zhang, D. Lee, TURINGBENCH: A benchmark environment for Turing test in the age of neural text generation, in: M.-F. Moens, X. Huang, L. Specia, S. W.-t. Yih (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2021, Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021, pp. 2001–2016. doi:10.18653/v1/2021.findings-emnlp.172.

[11] Y. Wang, J. Mansurov, P. Ivanov, J. Su, A. Shelmanov, A. Tsvigun, O. Mohammed Afzal, T. Mahmoud, G. Puccetti, T. Arnold, A. Aji, N. Habash, I. Gurevych, P. Nakov, M4GT-bench: Evaluation benchmark for black-box machine-generated text detection, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 3964–3992. doi:10.18653/v1/2024.acl-long.218.

[12] L. Kushnareva, T. Gaintseva, D. Abulkhanov, K. Kuznetsov, G. Magai, E. Tulchinskii, S. Barannikov, S. Nikolenko, I. Piontkovskaya, Boundary detection in mixed AI-human texts, in: First Conference on Language Modeling, 2024. URL: https://openreview.net/forum?id=kzzwTrt04Z.

[13] K. Kuznetsov, E. Tulchinskii, L. Kushnareva, G. Magai, S. Barannikov, S. Nikolenko, I. Piontkovskaya, Robust ai-generated text detection by restricted embeddings, 2024. arXiv:2410.08113.

[14] G. Gritsai, A. Voznyuk, A. Grabovoy, Y. Chekhovich, Are ai detectors good enough? a survey on quality of datasets with machine-generated texts, 2024. arXiv:2410.14677.

[15] M. Crawshaw, Multi-task learning with deep neural networks: A survey, 2020. arXiv:2009.09796.

[16] Z. Guo, K. Jiao, X. Yao, Y. Wan, H. Li, B. Xu, L. Zhang, Q. Wang, Y. Zhang, Z. Mao, USTC-BUPT at SemEval-2024 task 8: Enhancing machine-generated text detection via domain adversarial neural networks and LLM embeddings, in: A. K. Ojha, A. S. Doğruöz, H. Tayyar Madabushi, G. Da San Martino, S. Rosenthal, A. Rosá (Eds.), Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 1511–1522. doi:10.18653/v1/2024.semeval-1.217.

[17] G. Gritsai, A. Voznyuk, I. Khabutdinov, A. Grabovoy, Advacheck at GenAI detection task 1: AI detection powered by domain-aware multi-tasking, in: F. Alam, P. Nakov, N. Habash, I. Gurevych, S. Chowdhury, A. Shelmanov, Y. Wang, E. Artemova, M. Kutlu, G. Mikros (Eds.), Proceedings of the 1stWorkshop on GenAI Content Detection (GenAIDetect), International Conference on Computational Linguistics, Abu Dhabi, UAE, 2025, pp. 236–243.

[18] J. Bevendorff, D. Dementieva, M. Fröbe, B. Gipp, A. Greiner-Petter, J. Karlgren, M. Mayerl, P. Nakov, A. Panchenko, M. Potthast, A. Shelmanov, E. Stamatatos, B. Stein, Y. Wang, M. Wiegmann, E. Zangerle, Overview of PAN 2025: Voight-Kampff Generative AI Detection, Multilingual Text Detoxification, Multi-Author Writing Style Analysis, and Generative Plagiarism Detection, in: J. C. de Albornoz, J. Gonzalo, L. Plaza, A. G. S. de Herrera, J. Mothe, F. Piroi, P. Rosso, D. Spina, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Sixteenth International Conference of the CLEF Association (CLEF 2025), Lecture Notes in Computer Science, Springer, Berlin Heidelberg New York, 2025.

[19] J. Bevendorff, Y. Wang, J. Karlgren, M. Wiegmann, M. Fröbe, A. Tsivgun, J. Su, Z. Xie, M. Abassy, J. Mansurov, R. Xing, M. N. Ta, K. A. Elozeiri, T. Gu, R. V. Tomar, J. Geng, E. Artemova, A. Shelmanov, N. Habash, E. Stamatatos, I. Gurevych, P. Nakov, M. Potthast, B. Stein, Overview of the "Voight-Kampff" Generative AI Authorship Verification Task at PAN and ELOQUENT 2025, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 – Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2025.

[20] D. Macko, R. Moro, A. Uchendu, J. Lucas, M. Yamashita, M. Pikuliak, I. Srba, T. Le, D. Lee, J. Simko, M. Bielikova, Multitude: Large-scale multilingual machine-generated text detection benchmark, in: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2023, p. 9960–9987. URL: http://dx.doi.org/10.18653/v1/2023.emnlp-main.616. doi:10.18653/v1/2023.emnlp-main.616.

[21] S. M. Xie, T. Ma, P. Liang, Composed fine-tuning: Freezing pre-trained denoising autoencoders for improved generalization, in: M. Meila, T. Zhang (Eds.), Proceedings of the 38th International Conference on Machine Learning, volume 139 of *Proceedings of Machine Learning Research*, PMLR, 2021, pp. 11424–11435.

[22] M. Fröbe, M. Wiegmann, N. Kolyada, B. Grahm, T. Elstner, F. Loebe, M. Hagen, B. Stein, M. Potthast, Continuous Integration for Reproducible Shared Tasks with TIRA.io, in: Advances in Information Retrieval. 45th European Conference on IR Research (ECIR 2023), Lecture Notes in Computer Science, Springer, Berlin Heidelberg New York, 2023, pp. 236–241.

[23] D. Hendrycks, K. Gimpel, Gaussian error linear units (gelus), 2023. URL: https://arxiv.org/abs/1606.08415. arXiv:1606.08415.

A. Data description

This appendix reports the statistics within the provided training dataset for Subtask 1. Statistics are presented for the target task of binary classification, genres and author attribution. More detailed in Figure 2.

In the shape of the architectural features that we came up with, we needed to cluster the models that generated the examples. We were able to do this in 4 classes, some of the models were not included in

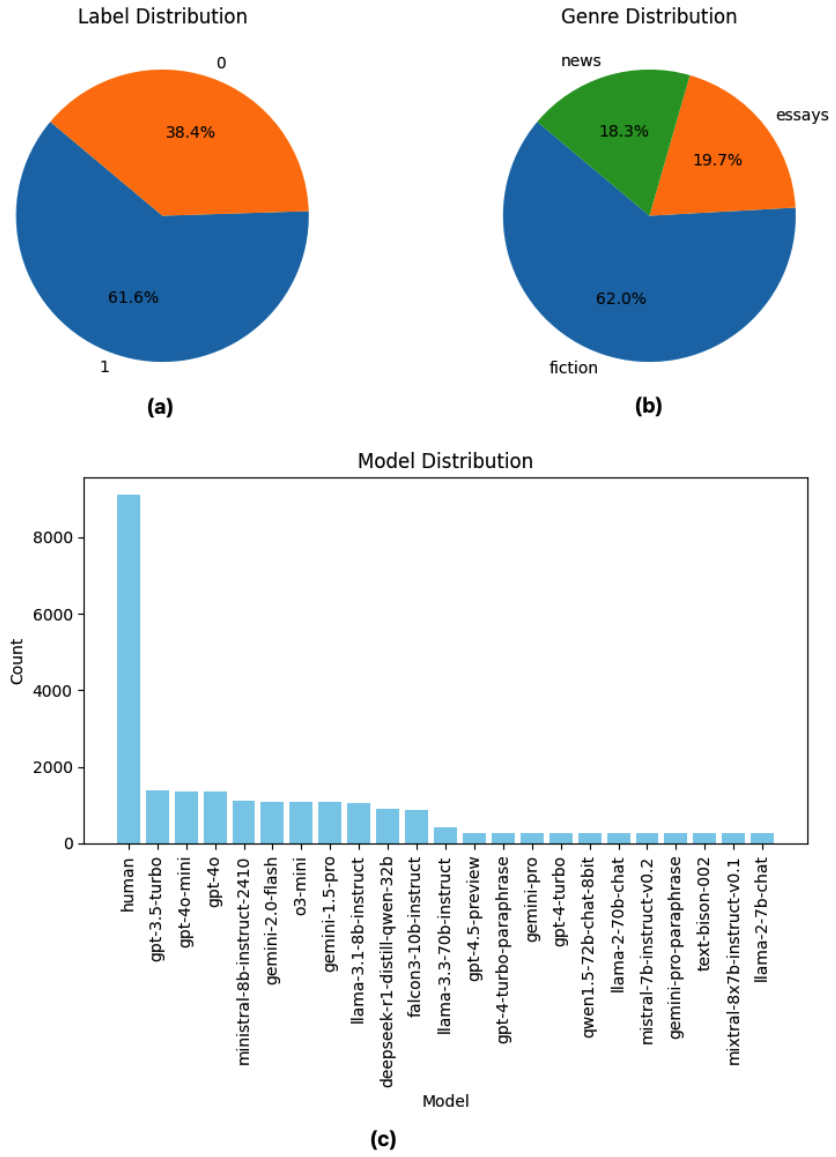

Figure 2: Analysing the training dataset for subtask 1. Figure (a) depicts the class ratio of the target task, Figure (b) depicts the distribution of genres among the available samples and Figure (c) depicts the distribution by text authors.

this data because their occurrences were not numerous and adding them to one of the existing groups could confuse the model. In the merging we maintained the following families and their components:

- **GPT family**: gpt-3.5-turbo, gpt-4o-mini, gpt-4o, o3-mini, gpt-4.5-preview, gpt-4-turbo-paraphrase, gpt-4-turbo
- **Mistral family**: ministral-8b-instruct-2410, mistral-7b-instruct-v0.2, mixtral-8x7b-instruct-v0.1
- **Gemini family**: gemini-2.0-flash, gemini-1.5-pro, gemini-pro, gemini-pro-paraphrase
- **LLaMA family**: llama-3.1-8b-instruct, llama-3.3-70b-instruct, llama-2-70b-chat, llama-2-7b-chat

B. Custom Classification Head

In our approach, we replaced the default one-layer linear classifier with a more extended version by adding multiple layers, the final structure of Custom Classification Head (CCH) is shown in Figure 3. We chose GELU [23] as the activation feature and added dropout. In earlier experiments, when

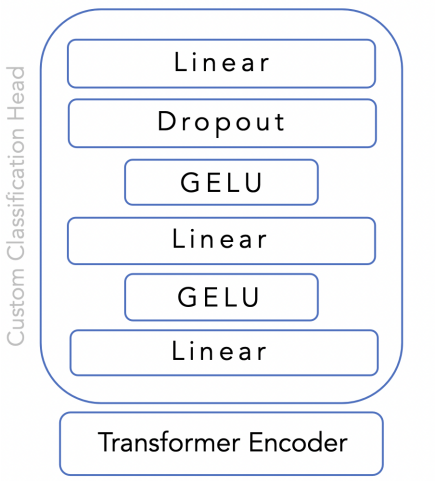

Figure 3: The architecture of the custom classification head used in the described approach. Value of `dropout` is equal to 0.5.

compared with the base head, this adaptation gives a higher quality therefore we used it in all subsequent experiments.