Team OSINT at GenAI Detection Task 1: Multilingual MGT Detection: Leveraging Cross-Lingual Adaptation for Robust LLMs Text Identification

Anonymous ACL submission

Abstract

Detecting AI-generated text has become increasingly prominent. This paper presents our solution for the DAIGenC Task 1 Subtask 2, where we address the challenge of distinguishing human-authored text from machine-generated content, especially in multilingual contexts. We introduce MLDet, a model that leverages Cross-Lingual Adaptation and Model Generalization strategies for Multilingual Machine-Generated Text (MGT) detection. By combining language-specific embeddings with fusion techniques, MLDet creates a unified, language-agnostic feature representation, enhancing its ability to generalize across diverse languages and models. Our approach demonstrates strong performance, achieving macro and micro F1 scores of 0.7067 and 0.7187, respectively, and ranking 15th in the competition [1]. We also evaluate our model across datasets generated by different distinct models in many languages, showcasing its robustness in multilingual and cross-model scenarios.

1 Introduction

Large Language Models (LLMs) have been quickly adopted in mainstream, making machine-generated content readily available across various platforms, such as news, social media, Q&A forums, education and academics. Recent models, ChatGPT, GPT-4 and Llama, can deliver quality responses to diverse prompts. However, the ease with which these models can articulate human-like text has raised concerns about potential misuse and information integrity (Liao, 2020). Given that humans have very slim chance of distinguishing machine-generated text from human-written content, there is an urgent need for automated detection systems. Many researches are underway developing various methods and models to address the challenge of distinguish MGT from human-authored content. Most of these works focus on English text, but struggle to differentiate text in other languages effectively. While the overall accuracy is generally high, the class-wise F1 scores remain low.

To tackle these issues, The COLING 2025 Workshop on DAIGenC (Wang et al., 2025) *"Task 1: Binary Multilingual Machine-Generated Text (MGT) Detection (Human vs. Machine)"* aim to refresh training and testing data with generations from novel LLMs and include new languages. The task is framed as—*"determining whether a given text is generated by a machine or authored by a human"*— and is divided into two sub-tasks: Subtask A: English-only MGT detection. Subtask B: Multilingual MGT detection. This paper focuses on Subtask B.

Our approach for multilingual MGT detection includes Cross-Lingual Adaptation and Model Generalization strategies. This methodology leverages language-specific embeddings to improve generalization across languages and models. Through this our model, **MLDet** aims to balanced performance on both macro and micro F1 scores.

2 Background

Over the last few years, numerous approaches have been proposed to tackle the task of Machine-generated text detection. Detecting machine-generated text is primarily formulated as a binary classification task (Zellers et al., 2019; Gehrmann et al., 2019; Ippolito et al., 2019), naively distinguishing between human-written and machine-generated text. In general, there are three main approaches: the supervised methods (Wang et al., 2023; Uchendu et al., 2021; Zellers et al., 2019; Zhong et al., 2020; Liu et al., 2023, 2022), the unsupervised ones, such as zero-shot methods (Solaiman et al., 2019; Ippolito et al., 2019; Mitchell

[1] https://github.com/mbzuai-nlp/COLING-2025-Workshop-on-MGT-Detection-Task1/tree/main

et al., 2023; Su et al., 2023; Hans et al.; Shijaku and Canhasi, 2023) and Adversarial measures on detection accuracy (Susnjak and McIntosh, 2024; Liang et al., 2023), especially within the education domain. For example, (Antoun et al., 2023) evaluates the robustness of detectors against character-level perturbations or misspelled words, focusing on French as a case study. (Krishna et al., 2024) train a generative model (DIPPER) to paraphrase paragraphs to evade detection. Although supervised approaches yield relatively better results, they are susceptible to overfitting (Mitchell et al., 2023; Su et al., 2023).

There are few Multilingual MGT Detection techniques which are mainly based on finetuned models (Macko et al., 2023, 2024; Hashmi et al., 2024; Bahad et al., 2024).

3 Proposed Model

In this section, we outline our approach for multilingual MGT detection.

3.1 Dataset Description

There are three datasets provided by (Wang et al., 2025): Train, Dev, and Test. Training and development data with 7 columns id, source, sub_source, language, model, label and text for the development phase. Testing data for the Evaluation phase. The AI and Human text distribution is tabulated in Table 1.

Data	AI	Human	Total
Train	674,083	257,968	932,051
Dev	178,728	110,166	288,894
Test	77,791	73,634	151,425

Table 1: Data for AI and Human across three datasets.

Table 2 includes different AI text generation models, languages, and domains. Specifically, the text in the training and development datasets are generated using 43 distinct models, while the training dataset uses 20 different models. Additionally, the training dataset includes data in 9 languages, whereas the testing dataset contains text in 20 languages. These variations in models and languages are essential for training and evaluation processes. Detail of dataset mention in Section A.2

3.2 Language-Specific Embedding Extraction

Given the input text $x^{(l)}$ from the "text" column in language l, we obtain a feature vector $h^{(l)}$ using

	lang	model	domain
Train	9	43	36
Dev	9	43	36
Test	16	20	27

Table 2: Table showing the different type of unique lang, model, and domain.

the pre-trained embedding model M_l specialized for the language l (e.g. *Chinese-BERT* (Sun et al., 2021) for Chinese, and *AraBERT* (Antoun et al., 2020) for Arabic) as $h^{(l)} = M_l(x^{(l)})$. Detail of Embedding models mention in Section A.3.1

This produces a feature vector that captures both language-specific and general semantic features. For handling unknown languages, we detect the language of input text and either use a default language model such as *XLM-RoBERTa* or fall back to an "unknown" embedding model, ensuring robustness across languages not explicitly included in the training set.

3.3 Cross-Lingual Fusion for Unified Representation

We combine embeddings from different languages in the dataset to create a unified representation as language-agnostic. Let $H = \{h^{(l_1)}, h^{(l_2)}, \ldots, h^{(l_n)}\}$ represent feature embeddings across languages.

Concatenation Fusion combines embeddings from various languages, as shown in equation 1. We then apply a weighted summation, where each language embedding $h^{(l)}$ is scaled by a learnable weight $w^{(l)}$, as shown in equation 2. The resulting fused embedding, $\mathbf{h}_{\text{fusion}}$, is passed through a Language Prediction Network, which predicts the language of the text. The output of this network is $\hat{y}_{lang}$ which is the Language model label, as described in equation 3.

$$h_{\text{fusion}} = [h^{(l_1)}; h^{(l_2)}; \ldots; h^{(l_n)}] \in \mathbb{R}^{n \times d} \quad (1)$$

$$h_{\text{fusion}} = \sum_{l \in \mathcal{L}} w^{(l)} h^{(l)} \quad (2)$$

$$\hat{y}_{lang} = f_{\text{language}}(\mathbf{h}_{\text{fusion}}) \quad (3)$$

3.4 Cross-Lingual Consistency Loss

To enforce consistency across languages, we introduce a cross-lingual consistency loss that encourages similarity between embeddings of the same

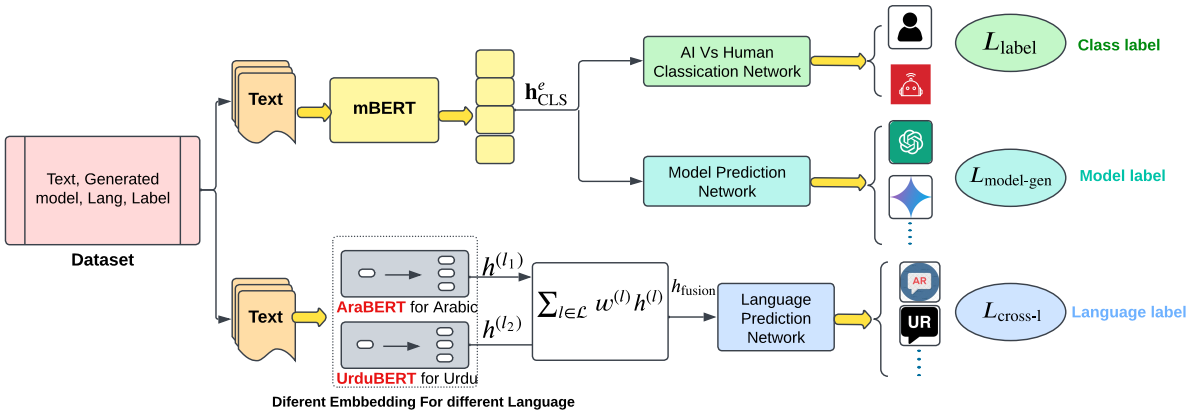

Figure 1: Proposed Detector model architecture: fusing stylometric features with a PLM embedding.

sample across languages. For each pair of languages (l_i, l_j) as shown in equation 4. This loss aligns embeddings across languages, promoting language-invariant features.

$$L_{\text{cross-}l} = \frac{1}{|\mathcal{L}|(|\mathcal{L}|-1)} \sum_{i \neq j} \|\hat{y}^{(l_i)}_{lang} - \hat{y}^{(l_j)}_{lang}\|^2 \quad (4)$$

The notation $|\mathcal{L}|$ denotes the total number of languages in this set.

3.5 Embedding Extraction with mBERT

Each text x from the "text" column, we pass it through mBERT, which produces a sequence of hidden states for each token in the text. The embedding corresponding to the $[CLS]$ token from the final hidden layer is then extracted as the representation for input text as shown in equation 5.

$$\mathbf{h}^e_{\text{CLS}} = \text{RoBERTa}(x)[\text{CLS}] \quad (5)$$

where, $\mathbf{h}^e_{\text{CLS}} \in \mathbb{R}^e$ is the CLS token embedding, and e is the embedding size of the model's output.

3.6 Model Generalization for MGT Detection

After obtaining the embedding $\mathbf{h}^e_{\text{CLS}}$, we pass it through the Model Prediction Network, which predicts the specific model responsible for generating the text. The output of this network is the predicted model label $\hat{y}_m$ as shown in equation 6.

$$\hat{y}_m = f_{\text{model}}(\mathbf{h}^e_{\text{CLS}}) \quad (6)$$

Given that the training and testing set includes 43 and 20 different models respectively, we introduce a model generalization loss to reduce reliance on specific training models.

The Cross-Model Pairwise Loss promote model-invariant features by minimizing the divergence between embeddings from different models, as in equation 7. Noise Augmentation adds Gaussian noise $\epsilon \sim \mathcal{N}(0, \sigma^2)$ during training to simulate unseen models as $\hat{y}^{\text{aug}}_m = \hat{y}_m + \epsilon$.

$$L_{\text{model-gen}} = \frac{1}{|\mathcal{M}|(|\mathcal{M}|-1)} \sum_{m \neq m'} \|\hat{y}_m - \hat{y}_{m'}\|^2 \quad (7)$$

$|\mathcal{M}|$ denotes the total number of generated model.

3.7 AI vs. Human Classification Network

The CLS token embedding $\mathbf{h}^e_{\text{CLS}}$ is passed to the AI vs. Human Classification Network. This network is a fully connected layer that outputs the probability of whether the text is human-written or machine-generated. The binary cross-entropy loss is used to compute the classification output as in equation 8.

$$L_{\text{label}} = -\frac{1}{N} \sum_{i=1}^{N} (y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)) \quad (8)$$

Where, $\hat{y}_i$ is the predicted probability for the i-th sample, y_i is the true label, N is the total number of samples.

3.8 Total Loss Function

Our model is optimized with a combination of label classification, cross-lingual, and model generalization losses. The total loss function is given by equation 9.

$$L_{\text{total}} = \alpha L_{\text{label}} + \beta L_{\text{cross-}l} + \gamma L_{\text{model-gen}} \quad (9)$$

where α, β, and γ are hyperparameters controlling the contributions of each component.

3.9 Training and Evaluation

We train the model by minimizing L_{total} with gradient descent using the AdamW optimizer. The CLS token embedding, which is 768 dimensions for mBERT serves as the input to fully connected hidden layers across the task-specific networks, each configured with 512 and 256 neurons and ReLU activations. The Macro F1 score used for evaluation to ensure balanced performance across classes. Further details of experimental setup in presented in section A.1.

During development phase, we consider different models with varied training strategies, detailed in Table 3. This includes direct fine-tuning of pretrained language models (PLMs) such as *XLM-RoBERTa* (Wiciaputra et al., 2021) and *mBERT* (Wu and Dredze, 2020) as the initial model. Furthermore, the *mBERT + CM* model utilizes cross-model adaptation (Section 3.6), while the *mBERT + CL* model applies Cross-Lingual Fusion (Section 3.3). The ***MLDet*** model incorporates Cross-Lingual Adaptation and Model Generalization strategies, as described in Section 3.

4 Results

The comprehensive analysis of the performance of various models on MGT detection based on micro F1, macro F1 score and accuracy are presented in Table 3. Final model, **MTDet** achieves macro F1 (classwise) score of 0.7739, outperforming other models.

Model	Macro F1	Micro F1	Accu.
XLM-RoBERTa	0.4133	0.4631	0.4631
mBERT	0.5203	0.8352	0.8352
mBERT + CM	0.5832	0.8523	0.8521
mBERT + CL	0.6044	0.8264	0.8264
MTDet (Final)	**0.7739**	**0.7938**	**0.7938**

Table 3: Performance scores of different models on Dev Dataset.

The result of evaluating on the test dataset is tabulated in Table 4. Final model, **MLDet**, demonstrates a balanced performance on both macro and micro F1 scores, achieving 0.7067 and 0.7187 respectively. Although it does not reach the highest micro F1 score, its macro F1 performance suggests a more balanced generalization across different languages and domains, reflecting its robustness in multilingual MGT detection.

Test Model	Macro F1	Micro F1
XLM-RoBERTa	0.3876	0.6798
mBERT	0.4307	0.7135
mBERT + CM	0.5678	0.8123
mBERT + CL	0.4897	0.8650
MLDet (Final)	**0.7067**	**0.7187**

Table 4: Performance comparison of various test models on Macro and Micro F1 scores.

5 Analysis

The performance of our **MLDet** model, as presented in Tables 3 and 4, highlights its strengths in achieving a balanced Macro and Micro F1 score. While direct PLM (mBERT) models may perform better in terms of accuracy and Micro F1, their low Class-wise (Macro) F1 scores indicate a bias toward majority classes in the dataset (as discussed in Table 1). These models also struggle to handle diverse languages and text generated by different AI models. In comparison, the mBERT + CM model slightly outperforms the mBERT + CL model in accuracy and micro F1 but falls short in macro F1, highlighting the importance of adaptation to unseen language pairs and model generalization. Our final **MTDet** model (mBERT + CM + CL) successfully balances macro and micro F1 scores, showcasing the effectiveness of integrating advanced cross-lingual adaptation and model generalization strategies.

However, the final model performs better on the development dataset compared to the test dataset. As noted in Section 2, the languages and generation models in the training and development datasets are similar, whereas the test dataset introduces different languages and generation models. Despite this increased challenge, the model still outperforms others in this scenario.

6 Conclusions

In conclusion, the robust performance of **MLDet** on diverse multilingual datasets underscores the importance of incorporating cross-lingual adaptation and model generalization strategies. A robust performance on the test dataset with a macro F1 score of 0.7067. By capturing a wide range of linguistic and contextual information, these strategies allow the model to generalize effectively across languages and domains, positioning **MLDet** as a versatile and efficient solution for MGT detection in multilingual settings.

References

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. Arabert: Transformer-based model for arabic language understanding. *arXiv preprint arXiv:2003.00104*.

Wissam Antoun, Virginie Mouilleron, Benoît Sagot, and Djamé Seddah. 2023. Towards a robust detection of language model-generated text: Is chatgpt that easy to detect? In *Actes de CORIA-TALN 2023. Actes de la 30e Conférence sur le Traitement Automatique des Langues Naturelles (TALN), volume 1: travaux de recherche originaux–articles longs*, pages 14–27.

Sankalp Bahad, Yash Bhaskar, and Parameswari Krishnamurthy. 2024. Fine-tuning language models for ai vs human generated text detection. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 918–921.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander M Rush. 2019. Gltr: Statistical detection and visualization of generated text. *arXiv preprint arXiv:1906.04043*.

A Hans, A Schwarzschild, V Cherepanova, H Kazemi, A Saha, M Goldblum, J Geiping, and T Goldstein. Spotting llms with binoculars: Zero-shot detection of machine-generated text, 2024. *URL: https://arxiv. org/abs/2401.12070*.

Ehtesham Hashmi, Sule Yildirim Yayilgan, Ibrahim A Hameed, Muhammad Mudassar Yamin, Mohib Ullah, and Mohamed Abomhara. 2024. Enhancing multilingual hate speech detection: From language-specific insights to cross-linguistic integration. *IEEE Access*.

Daphne Ippolito, Daniel Duckworth, Chris Callison-Burch, and Douglas Eck. 2019. Automatic detection of generated text is easiest when humans are fooled. *arXiv preprint arXiv:1911.00650*.

Kalpesh Krishna, Yixiao Song, Marzena Karpinska, John Wieting, and Mohit Iyyer. 2024. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. *Advances in Neural Information Processing Systems*, 36.

Weixin Liang, Mert Yuksekgonul, Yining Mao, Eric Wu, and James Zou. 2023. Gpt detectors are biased against non-native english writers. *Patterns*, 4(7).

S Matthew Liao. 2020. *Ethics of artificial intelligence*. Oxford University Press.

Xiaoming Liu, Zhaohan Zhang, Yichen Wang, Hang Pu, Yu Lan, and Chao Shen. 2022. Coco: Coherence-enhanced machine-generated text detection under data limitation with contrastive learning. *arXiv preprint arXiv:2212.10341*.

Yikang Liu, Ziyin Zhang, Wanyang Zhang, Shisen Yue, Xiaojing Zhao, Xinyuan Cheng, Yiwen Zhang, and Hai Hu. 2023. Argugpt: evaluating, understanding and identifying argumentative essays generated by gpt models. *arXiv preprint arXiv:2304.07666*.

Dominik Macko, Jakub Kopal, Robert Moro, and Ivan Srba. 2024. Multisocial: Multilingual benchmark of machine-generated text detection of social-media texts. *arXiv preprint arXiv:2406.12549*.

Dominik Macko, Robert Moro, Adaku Uchendu, Jason Samuel Lucas, Michiharu Yamashita, Matúš Pikuliak, Ivan Srba, Thai Le, Dongwon Lee, Jakub Simko, et al. 2023. Multitude: Large-scale multilingual machine-generated text detection benchmark. *arXiv preprint arXiv:2310.13606*.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature. In *International Conference on Machine Learning*, pages 24950–24962. PMLR.

Rexhep Shijaku and Ercan Canhasi. 2023. Chatgpt generated text detection. *Publisher: Unpublished*.

Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, Gretchen Krueger, Jong Wook Kim, Sarah Kreps, et al. 2019. Release strategies and the social impacts of language models. *arXiv preprint arXiv:1908.09203*.

Jinyan Su, Terry Yue Zhuo, Di Wang, and Preslav Nakov. 2023. Detectllm: Leveraging log rank information for zero-shot detection of machine-generated text. *arXiv preprint arXiv:2306.05540*.

Zijun Sun, Xiaoya Li, Xiaofei Sun, Yuxian Meng, Xiang Ao, Qing He, Fei Wu, and Jiwei Li. 2021. Chinesebert: Chinese pretraining enhanced by glyph and pinyin information. *arXiv preprint arXiv:2106.16038*.

Teo Susnjak and Timothy R McIntosh. 2024. Chatgpt: The end of online exam integrity? *Education Sciences*, 14(6):656.

Adaku Uchendu, Zeyu Ma, Thai Le, Rui Zhang, and Dongwon Lee. 2021. Turingbench: A benchmark environment for turing test in the age of neural text generation. *arXiv preprint arXiv:2109.13296*.

Yuxia Wang, Artem Shelmanov, Jonibek Mansurov, Akim Tsvigun, Vladislav Mikhailov, Rui Xing, Zhuohan Xie, Jiahui Geng, Giovanni Puccetti, Ekaterina Artemova, Jinyan Su, Minh Ngoc Ta, Mervat Abassy, Kareem Elozeiri, Saad El Dine Ahmed, Maiya Goloburda, Tarek Mahmoud, Raj Vardhan Tomar, Alexander Aziz, Nurkhan Laiyk, Osama Mohammed Afzal, Ryuto Koike, Masahiro Kaneko, Alham Fikri Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2025. Genai content detection task 1: English and multilingual machine-generated text detection: Ai vs. human. In *Proceedings of the 31st International Conference on Computational Linguistics (COLING)*, Abu Dhabi, UAE. Association for Computational Linguistics.

Zecong Wang, Jiaxi Cheng, Chen Cui, and Chenhao Yu. 2023. Implementing bert and fine-tuned roberta to detect ai generated news by chatgpt. *ArXiv*, abs/2306.07401.

Yakobus Keenan Wiciaputra, Julio Christian Young, and Andre Rusli. 2021. Bilingual text classification in english and indonesian via transfer learning using xlm-roberta. *International Journal of Advances in Soft Computing & Its Applications*, 13(3).

Shijie Wu and Mark Dredze. 2020. Are all languages created equal in multilingual bert? *arXiv preprint arXiv:2005.09093*.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. *Advances in neural information processing systems*, 32.

Wanjun Zhong, Duyu Tang, Zenan Xu, Ruize Wang, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. 2020. Neural deepfake detection with factual structure of text. *arXiv preprint arXiv:2010.07475*.

A Example Appendix

A.1 Details of Experimental Setups

The experimental setup for this study includes a comprehensive range of hyperparameters, multilingual datasets, and model embeddings tailored to effectively detect machine-generated text across diverse languages and domains.

Key hyperparameters, mention in Table 4 such as learning rate, batch size, and dropout rate, were carefully tuned to optimize model performance. Additionally, weights for classification, domain, and cross-lingual loss were experimentally adjusted to ensure the model's adaptability to varied linguistic structures. The optimizer used was Adam, with specific parameters for Beta values and epsilon, while learning rate scheduling was customized based on experimental results. The setup is designed to capture fine-grained cross-lingual features, thereby enabling robust language-specific and language-agnostic pattern recognition.

A.2 Details of Dataset and Used Model

The table 2 summarizes the diversity in datasets: Train/Dev (9 lang, 43 models, 36 domains) and Test (16 lang, 20 models, 27 domains), highlighting broader testing scope.

A.2.1 Training Dataset detail

Traning dataset with 7 columns id,source, sub_source, language, model, label and text

Hyperparameter	**Typical Values**
Learning Rate (η)	$1e-5$ to $1e-3$
Batch Size	16, 32, 64
Number of Epochs	100 to 500
Weight for Classification Loss (λ)	Tuned based on experiment
Weight for Domain Loss (γ)	Tuned based on experiment
Weight for Cross-Lingual Loss (δ)	Tuned based on experiment
Dropout Rate	0.1 to 0.5
Gradient Reversal Layer Parameter	Tuned based on experiment
Hidden Layer Dimensions	Tuned based on experiment
Optimizer (Adam Parameters)	Beta1: 0.9, Beta2: 0.999, Epsilon: $1e-8$
Learning Rate Scheduler Parameters	Tuned based on experiment

Table 5: List of Hyper parameters for the Experiment during Training

The dataset includes diverse **languages** Table 6 with English (229,209 human, 381,467 AI) dominating, followed by Chinese (19,315 human, 15,969 AI). Bulgarian (4,205 human, 3,886 AI) and German (231 human, 4,462 AI) emphasize AI. Indonesian, Urdu, and Russian show balanced distributions, while Italian contains only AI samples (4,174).

The experimental setup includes diverse **generation models**: OpenAI's GPT series (`GPT-3.5-Turbo`, `Davinci`, `GPT4`), BLOOM models (`Bloomz`, `bloom_7b`), Meta's `Llama3`, `OPT`, and `Llama2-fine-tuned`, along with `Flan_T5`, `T0`, and specialized models (`Gemma`, `Jais-30b`). These cover multilingual and task-specific applications, emphasizing robust, fine-tuned, and scalable AI capabilities.

A.3 Testing Dataset details

The testing dataset expands the linguistic range, incorporating additional languages such as Kazakh, Norwegian, and Hindi as mention Table 8, thus testing the model's capacity to generalize to unseen linguistic contexts. A broad array of generative models, including recent releases like GPT-4, Llama, and Baichuan, are represented, allowing a thorough evaluation of the model's effectiveness

Language (Code)	Human	AI
Arabic (ar)	344	1770
Bulgarian (bg)	4205	3886
German (de)	231	4462
English (en)	229209	381467
Indonesian (id)	1895	2081
Italian (it)	0	4174
Russian (ru)	684	630
Urdu (ur)	2085	1676
Chinese (zh)	19315	15969

Table 6: Counts of Human and AI instances across languages in Training Dataset.

Language	Embedding
English	*RoBERTa*
Chinese	*Chinese-BERT*
Bulgarian	*XLM-RoBERTa*
German	*GottBERT*
Italian	*AlBERTo*
Indonesian	*IndoBERT*
Urdu	*UrduBERT*
Arabic	*AraBERT*
Russian	*RuBERT*

Table 7: Languages and their corresponding embeddings during Training.

Language	Human	AI
Arabic	4350	6320
Chinese	29947	33062
Dutch	600	600
German	1865	0
Hebrew	1182	0
Hindi	599	600
Indonesian	600	600
Italian	2496	2800
Japanese	300	300
Kazakh	1171	1300
Norwegian	1544	0
Russian	13039	13094
Spanish	600	600
Urdu	13190	17315
Vietnamese	1126	1200
Russian	1025	0

Table 8: Counts of Human and AI instances across various languages in Testing Dataset.

across diverse AI text generation systems. This experimental design facilitates a detailed assessment of the model's cross-lingual performance and robustness against various language models, ensuring comprehensive validation of the approach.

Model	Count
Human	73634
GPT-4o	28538
GPT-4o-mini	6845
gpt4o	6591
Vikhrmodels	6503
gpt-4o-2024-05-13	5998
Baichuan2-13B-Chat	5521
ChatGLM3-6B	5359
Llama 3.1 405B instruct	4000
gpt-4o	2400
gpt-4	1545
GPT-4-turbo	1400
glm-4-9b-chat	778
claude-3-5-sonnet	773
GPT4	299
Qwen	297
GPT3.5	297
ChatGLM	295
Baichuan	283
qwen2.5 72b	69

Table 9: Counts of instances for different models in Testing Dataset.

Table 9:The model distribution includes 73,634 human samples and a variety of AI models: `GPT-4o` (28,538), `GPT-4o-mini` (6,845), `Vikhrmodels` (6,503), and `gpt-4o-2024-05-13` (5,998). Other models include `Baichuan2-13B-Chat` (5,521), `ChatGLM3-6B` (5,359), `Llama 3.1 405B instruct` (4,000), and smaller counts for models like `GPT-4-turbo` (1,400), `glm-4-9b-chat` (778), and `GPT4` (299). The dataset highlights diverse AI capabilities across various architectures and scales.

A.3.1 Detail of Language-Specific Embedding Used

The dataset is organized by language, embedding models, and instance counts for human and AI content 6. Training spans nine languages with language-specific models (e.g., RoBERTa for English, Chinese-BERT for Chinese, AraBERT for Arabic) as mention Table 7, enabling nuanced feature extraction. It ensures a balanced multilingual setup, with English and Chinese dominating, and sufficient representation for Bulgarian, Indonesian, and Urdu.