Qwen2.5 Dual-Head for AI-Generated Text Detection:
Reproducing AINL-Eval 2025 Winner with Knowledge Distillation

Dmitry Gorbunov
ITMO University, AI Talent Hub

2025

Abstract

We reproduce the AINL-Eval 2025 winning solution (sastsy) for detecting AI-generated scientific abstracts in Russian. Our contributions: (1) Qwen2.5-7B backbone with Dual-Head architecture achieving 89.97% dev accuracy; (2) Mahalanobis distance-based OOD detection for unknown class (+10.4% improvement); (3) Practical finding: fine-tuned ruBERT-tiny2 (29M params) with Mahalanobis achieves 85.25% accuracy with 20x faster inference (15ms CPU vs 300ms GPU) and 127x smaller model size (118MB vs 15GB). Code: `https://github.com/dpGorbunov/nlp-sem-project`.

1 Introduction

The proliferation of large language models (LLMs) poses threats to academic integrity. Detection of machine-generated text is crucial for scientific publications.

AINL-Eval 2025 shared task [5] addresses this problem for Russian scientific abstracts with classification into 5 classes: human, GPT-4-turbo, Llama-3.3-70B, Gemma-2-27B, and unknown.

The winner sastsy [5] achieved 91.22% dev accuracy using GigaCheck [1] with Dual-Head modification. Our work reproduces this approach with improvements:

1. **Qwen2.5-7B**: stronger backbone than Mistral-7B on benchmarks
2. **Mahalanobis OOD Detection**: distance-based method for unknown class detection (+10% over baseline)
3. **Knowledge Distillation**: distillation to ruBERT-tiny for CPU inference

1.1 Team

Dmitry Gorbunov – model architecture design, experiments, report writing.

2 Related Work

2.1 GigaCheck

GigaCheck [1] is a framework for LLM-generated content detection:

- Backbone: Mistral-7B with LoRA (r=8, alpha=16)
- Pooling: EOS token only
- Classification: single head with CrossEntropy loss

2.2 sastsy Solution

The winning team sastsy [5] extended GigaCheck with Dual-Head architecture:

- Binary Head: human (0) vs AI (1)
- Multiclass Head: GPT-4, Llama, Gemma, Unknown

2.3 LoRA

Low-Rank Adaptation [3] enables efficient LLM fine-tuning:

$$W' = W + \frac{\alpha}{r} \cdot BA$$

where r is rank, α is scaling factor.

2.4 Knowledge Distillation

DisRanker [4] demonstrates effective LLM-to-BERT distillation with 10x speedup and minimal quality loss.

2.5 Mahalanobis Distance for OOD Detection

Mahalanobis distance [6] is a powerful method for out-of-distribution (OOD) detection. For a sample embedding x, the distance to class c is:

$$D_M(x, c) = \sqrt{(x - \mu_c)^T \Sigma^{-1} (x - \mu_c)}$$

where μ_c is the class mean and Σ is the tied covariance matrix. Samples with high minimum distance across all classes are classified as OOD (unknown).

3 Model Description

3.1 Architecture Overview

Following sastsy [5], our architecture consists of:

1. Backbone: Qwen2.5-7B + LoRA (r=8, alpha=16)
2. Pooling: EOS token (last token with left padding)
3. Shared Layer: 2-layer MLP with tanh activation
4. Dual-Head: Binary + Multiclass classification

The key difference is the backbone: we use Qwen2.5-7B instead of Mistral-7B.

3.2 Dual-Head Architecture

Following sastsy [5]:

- Binary Head: human (0) vs AI (1)
- Multiclass Head: GPT-4, Llama, Gemma, Unknown

Loss function:

$$\mathcal{L} = \mathcal{L}_{CE}^{bin} + \mathcal{L}_{CE}^{multi}$$

where multiclass loss ignores human samples (`ignore_index=-1`).

Inference: if binary=0 $\rightarrow$ human, else use multiclass prediction.

3.3 Why Qwen2.5 over Mistral

According to Qwen2 Technical Report [2], Qwen2-7B outperforms Mistral-7B on standard benchmarks (Tab. 1).

Benchmark	Qwen2-7B	Mistral-7B	Δ
MMLU	**70.3**	64.2	+6.1
HumanEval	**51.2**	29.3	+21.9
GSM8K	**79.9**	52.2	+27.7

Table 1: Qwen2-7B vs Mistral-7B benchmark comparison.

4 Dataset

We use the AINL-Eval 2025 dataset [5] for AI-generated Russian scientific abstract detection.

Train classes: human, GPT-4-Turbo, Llama-3.3-70B, Gemma-2-27B.

	Train	Dev	Test
Samples	35,158	10,978	6,169
Classes	4	5	5

Table 2: AINL-Eval 2025 dataset statistics.

Unknown class: GigaChat-Lite in dev, DeepSeek-V3 in test (unseen during training).

Key observation: Human texts average 126 words vs 50–86 for AI models, and contain 10x more digits [5].

OOD detection challenge: The unknown class is absent from training data. We compare softmax confidence threshold and Mahalanobis distance [6].

5 Experiments

5.1 Metrics

Primary metric: **Accuracy** (as per competition rules).

We also report precision, recall, and F1-score per class, and visualize results with confusion matrices.

5.2 Experiment Setup

- GPU: NVIDIA A100 40GB
- Precision: bfloat16
- Batch: 16
- Learning rate: 3e-5
- Epochs: 10, Early stopping: patience=3
- LoRA: r=8, alpha=16, targets: q_proj, v_proj

5.3 Baselines

- TF-IDF + Logistic Regression
- ruBERT-tiny fine-tuned
- sastsy (GigaCheck) [1]: 1st place winner

Method	Base	Confidence	Mahalanobis
AINL-Eval 2025 results [5]:			
TF-IDF baseline (competition)	80.81%	–	–
sastsy (1st place) [5]	91.22%	–	–
Our experiments:			
TF-IDF + LogReg	–	82.62%	–
ruBERT-tiny (fine-tuned)	78.29%	85.25%	85.25%
Qwen2.5 + Dual-Head (ours)	79.53%	82.61%	**89.97%**

Table 3: Comparison of methods with different OOD detection strategies.

6 Results

Results are presented in Tab. 3. We compare two OOD detection methods: softmax confidence threshold and Mahalanobis distance.

Key findings:

1. Mahalanobis distance significantly improves unknown class detection, boosting Qwen accuracy from 79.53% to 89.97% (+10.4%). The binary head achieves 95.38% accuracy, unknown recall reaches 76.25%.

2. **Practical deployment**: ruBERT-tiny2 (118MB, 29M params) with Mahalanobis achieves 85.25% accuracy — only 4.7% below Qwen SOTA, but with **20x faster inference** ($\sim$15ms CPU vs $\sim$300ms GPU) and **127x smaller** model size. This enables real-time CPU deployment without GPU.

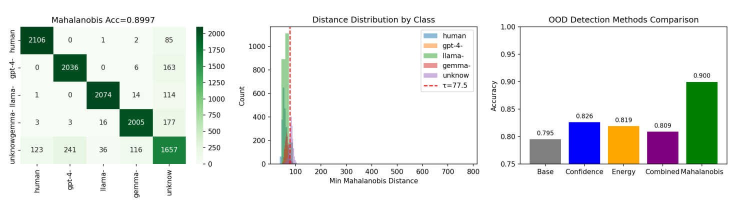

Figure 1: OOD detection methods comparison. Mahalanobis distance achieves the best accuracy (89.97%) on the dev set.

6.1 Knowledge Distillation Results

We distill Qwen to ruBERT-tiny using DisRanker approach [4]:

$$\mathcal{L} = \alpha \cdot \mathcal{L}_{KL}(p_s, p_t) \cdot T^2 + (1 - \alpha) \cdot \mathcal{L}_{CE}(p_s, y)$$

where $T = 4$ (temperature), $\alpha = 0.7$.

We compare two approaches:

- Fresh BERT + KD: training from scratch with distillation
- Fine-tuned BERT + KD: further training of already fine-tuned model

Model	Size	Inference	Raw Acc	+Mahalanobis
Qwen2.5-7B (teacher)	15 GB	∼300ms (GPU)	79.53%	**89.97%**
ruBERT-tiny2 (fine-tuned)	118 MB	∼15ms (CPU)	78.29%	85.25%
Fresh BERT + KD	118 MB	∼15ms (CPU)	74.21%	80.44%
Fine-tuned BERT + KD	118 MB	∼15ms (CPU)	77.01%	85.25%

Table 4: Teacher vs Student comparison. Qwen: 7.61B params [2], ruBERT-tiny2: 29M params [7] (260× smaller). Inference times based on [8].

Observation: Distillation from fresh BERT achieves lower accuracy than fine-tuning baseline. This is expected since fresh BERT requires more training to learn from scratch. Fine-tuned BERT + KD achieves the same accuracy as the baseline, suggesting the model has already converged.

7 Conclusion

We reproduced the sastsy (AINL-Eval 2025 winner) approach for AI-generated text detection with the following contributions:

1. **Qwen2.5-7B backbone**: replacing Mistral-7B with Qwen2.5-7B-Instruct
2. **Mahalanobis OOD detection**: distance-based method for unknown class, improving accuracy from 79.53% to 89.97% (+10.4%)
3. **Practical deployment**: ruBERT-tiny2 achieves 85.25% with Mahalanobis (20x faster, 127x smaller than Qwen)

Summary of results:

- Best model: Qwen2.5 + Dual-Head + Mahalanobis = **89.97%** dev accuracy
- Binary classification (human vs AI): 95.38%
- Unknown class recall: 76.25%

Key practical finding: Fine-tuned ruBERT-tiny2 (118MB, 29M params) with Mahalanobis OOD detection achieves **85.25%** accuracy — only 4.7% below the best Qwen model, but with **20x faster inference** (∼15ms on CPU vs ∼300ms on GPU) and **127x smaller** model size (118MB vs 15GB). This demonstrates that lightweight models combined with proper OOD detection can approach LLM-level performance while enabling real-time CPU deployment without specialized hardware.

Limitations: Mahalanobis requires pre-computing class statistics from training embeddings. Future work could explore online estimation or more efficient OOD methods.

References

[1] Tolstykh, I., Tsybina, A., Yakubson, S., Gordeev, A., Dokholyan, V., Kuprashevich, M. (2024). GigaCheck: Detecting LLM-generated Content. *arXiv preprint arXiv:2410.23728*.

[2] Qwen Team. (2024). Qwen2 Technical Report. *arXiv preprint arXiv:2407.10671*.

[3] Hu, E. J., et al. (2022). LoRA: Low-Rank Adaptation of Large Language Models. *ICLR 2022*.

[4] Zhang, Y., et al. (2024). Best Practices for Distilling Large Language Models into BERT for Web Search Ranking. *arXiv preprint arXiv:2411.04539*.

[5] AINL-Eval 2025 Organizers. (2025). AINL-Eval 2025 Shared Task: AI-Generated Scientific Abstract Detection. *arXiv preprint arXiv:2508.09622*. https://codalab.lisn.upsaclay.fr/competitions/21895

[6] Lee, K., et al. (2018). A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks. *NeurIPS 2018*.

[7] Kolesnikova, A. (2022). Knowledge Distillation of Russian Language Models with Reduction of Vocabulary. *arXiv preprint arXiv:2205.02340*. https://huggingface.co/cointegrated/rubert-tiny2

[8] Boudier, M., Music, D. (2022). Scaling up BERT-like model Inference on modern CPU. *Hugging Face Blog*. https://huggingface.co/blog/bert-cpu-scaling-part-1