Improving Fake News Detection in Low-Resource Languages through Retrieval-Augmented Generation And Parameter-Efficient Fine-Tuning

ASM H. Kabir, Sameed A. Khan, Alexander A. Kharlamov, Ilia M. Voronkov

***Abstract*—The proliferation of fake news in low-resource languages poses critical societal challenges due to limited resources, insufficient NLP tooling, and the morphological complexity of non-English scripts. Bangla, the seventh most spoken language in the world, remains severely under-resourced in automated fake news detection research. This paper presents a systematic comparative evaluation of three adaptation strategies for Bangla fake news detection using the Qwen3.5-4B multilingual large language model: vanilla zero-shot settings, Retrieval-Augmented Generation (RAG) backed by vector database with multilingual sentence embeddings, and parameter-efficient fine-tuning via Low-Rank Adaptation (LoRA). Experiments on a balanced dataset of Bangla news articles demonstrate that RAG improves accuracy by 6.73 percentage points over the zero-shot baseline from 73.85% to 80.58%, while LoRA fine-tuning with correct inference-time evaluation methodology achieves near-perfect classification at 99.04% accuracy and macro F1 of 0.9904. This work establishes strong baselines for Bangla fake news detection and contributes methodological insights critical for deploying fine-tuned LLMs in low-resource settings.**

***Keywords*— fake news detection, Bangla, low-resource languages, retrieval-augmented generation, LoRA, large language models.**

I. INTRODUCTION

The rapid proliferation of misinformation on digital platforms constitutes one of the most pressing online crises of the twenty-first century [1]. Falsified news stories, manipulated quotations, and misguided scientific health claims spread virally across social media, distorting public perception on topics ranging from electoral politics to pandemic response [2]. While fake news detection in the English language has matured substantially through benchmark datasets, transformer-based classifiers, and large-scale fact-checking pipelines [3], [4], analogous systems for low-resource languages remain in early stages of development. The consequence is a systemic asymmetry: populations whose primary language falls outside the high-resource tier are substantially more vulnerable to online misinformation.

Bangla presents a paradigmatic case of this asymmetry. As the seventh most-spoken language globally, Bangla's digital presence on social media platforms is enormous [5]. However, the Bangla NLP ecosystem is severely constrained by a shortage of publicly available labeled datasets, a complex morphological structure that challenges tokenization and embedding models designed primarily for European languages, and limited human annotation resources for fake news detection tasks. Existing Bangla fake news detection studies are typically confined to small, domain-specific corpora and rely on classical feature engineering and neural architectures that underperform relative to contemporary LLM-based approaches [6].

Recent advances in multilingual large language models (LLMs) have opened new possibilities for low-resource text classification. Models such as mBERT [7], XLM-RoBERTa (XLM-R) [8], and Qwen3.5 [9] encode rich cross-lingual representations through pre-training on massively multilingual corpora, enabling useful zero-shot transfer to languages with little or no task-specific labeled data. Two adaptation paradigms have emerged as particularly promising: (1) Retrieval-Augmented Generation (RAG) [10], which dynamically augments model input with semantically similar labeled examples from an external knowledge base, and (2) parameter-efficient fine-tuning via Low-Rank Adaptation (LoRA) [11], which modifies only a small fraction of model parameters through low-rank matrix decomposition, enabling domain adaptation at a fraction of the computational cost of full fine-tuning.

This paper makes three primary contributions. First, we design, implement, and systematically evaluate three complementary pipelines for Bangla fake news detection: vanilla zero-shot inference, RAG, and LoRA fine-tuning—using the Qwen3.5-4B multilingual LLM as a shared base model, enabling controlled comparison of adaptation strategies. Second, we provide a study in LLM evaluation methodology for different pipelines. Third, we show that LoRA fine-tuning of a 4B-parameter model with correct evaluation achieves 99.04% accuracy (F1 = 0.9904) on a

Reveived: 21.05.2026
ASM. H. Kabir, Moscow Institute of Physics and Technology (National Research University), Moscow Oblast, Russia, 141701 (e-mail: humaun.kabir@phystech.edu)
S. A. Khan, Innopolis University, Innopolis, Republic of Tatarstan, Russia, 420500 (e-mail: sameedkhandurrani@gmail.com)
Alexander A. Kharlamov, Moscow Institute of Physics and Technology (National Research University), Moscow Oblast, Russia, 141701 (e-mail: kharlamov@analyst.ru)
Ilia M. Voronkov, Moscow Institute of Physics and Technology (National Research University), Moscow Oblast, Russia, 141701 (e-mail: voronkov.im@phystech.edu)

balanced test set, while RAG delivers the most balanced per-class recall profile (0.39 pp gap) without any model training.

II. Related Work

Fake news detection has been studied as a binary or multi-class classification problem applied to news headlines, article bodies, and social-media posts [3]. Early data-driven approaches relied on handcrafted features, sentiment polarity, writing complexity, and source credibility signals that are fed into logistic regression or support vector machine classifiers [3]. The introduction of BERT [7] made a shift toward end-to-end fine-tuning of pre-trained transformer encoders, which achieved state-of-the-art results on English benchmarks including LIAR [12], FakeNewsNet [13], and FEVER [14]. More recently, large generative models have been applied to fake news detection both as direct classifiers (prompting-based) and as generators of explanation-augmented labels [4], [15].

Multi-modal approaches extending beyond text to incorporate images, metadata, and social propagation graphs have also gained traction [16]. Nakamura et al. [16] introduced r/Fakeddit, a large-scale multi-modal benchmark, while Shu et al. [13] demonstrated that user engagement signals complement text features for FakeNewsNet. However, most of these advances target English or high-resource European languages, leaving low-resource languages systematically underserved.

Islam et al. [6] provide a comprehensive survey of Bangla NLP, cataloguing the scarcity of public datasets across tasks including sentiment analysis, named-entity recognition, and question answering. For fake news detection specifically, existing Bangla work has been limited to small corpora with models based on mBERT fine-tuning or CNN-BiLSTM architectures achieving 70–85% accuracy [6], [17]. The lack of standardized benchmarks and evaluation protocols makes cross-study comparison difficult. BanFakeNews[18] is an open dataset with partially labeled dataset for Bangla fake news, we leverage this dataset in our experiments.

RAG, formalized by Lewis et al. [10], augments a parametric language model with a non-parametric retrieval component that fetches relevant documents at inference time. The retrieved documents are prepended to the input, enabling the model to leverage factual context beyond what is encoded in its weights. RAG has been applied to open-domain question answering [10], slot filling, and knowledge-intensive dialogue. For classification tasks, RAG enables few-shot in-context learning from real labeled examples, adapting to new domains without gradient updates [19]. The effectiveness of RAG depends critically on the quality of the retrieval component: dense passage retrieval [20] using bi-encoder sentence embeddings has largely superseded sparse BM25-based retrieval for semantic similarity tasks.

Recent work has examined RAG in low-resource and multilingual settings. Shi et al. [21] show that cross-lingual RAG is retrieving documents in English for queries in other languages that can bridge resource gaps, though intra-lingual retrieval generally performs better when a sufficient source corpus exists. For Bangla, intra-lingual retrieval is feasible given the availability of multilingual sentence encoders such as paraphrase-multilingual-MiniLM-L12-v2 [22] that cover Bangla in a shared embedding space with high-resource languages.

Fine-tuning all parameters of a billion-parameter model is computationally prohibitive for most practitioners and risks catastrophic forgetting of pre-trained representations [23]. Parameter-efficient fine-tuning (PEFT) methods address this by training only a small subset of parameters or newly added adapter modules [24]. LoRA [11] injects trainable low-rank matrices into selected weight projections, reducing trainable parameters by 10–1000× while matching or exceeding full fine-tuning accuracy on several benchmarks. QLoRA [25] extends LoRA to 4-bit quantized base models, enabling fine-tuning on consumer GPUs. The Unsloth library [26] provides further acceleration through custom CUDA kernels and gradient checkpointing, yielding 2× speedup over standard Hugging Face pipelines. Our work employs LoRA via Unsloth on Qwen3.5-4B, adapting all seven linear projection layers across all transformer blocks for maximum adaptation capacity.

Concurrent work by Hu et al. [24] on PEFT benchmarks demonstrates that LoRA with rank 8–64 consistently outperforms prefix tuning and prompt tuning on classification tasks, particularly for domain-specific adaptation. Our choice of rank 32 with equal alpha scaling (no dropout) follows the configuration recommended for tasks with modest dataset sizes (1,000–10,000 samples).

III. Dataset

We selected all labeled fake news samples from [18] and an equal number of labeled real news samples to create a balanced dataset. No samples were added, removed, or modified, we subset the original data to include only the labeled portion while maintaining class balance. We curated a balanced subset of 2,598 Bangla news articles from [18], comprising 1,299 fake and 1,299 real samples selected from the labeled portion of the original dataset. Each fake article was annotated manually by human review and curation. Real articles were sourced from established Bangladeshi journalistic outlets, filtered to match the topical distribution of fake articles politics, health, sports, and entertainment.

Articles were retained in Unicode Bengali script (UTF-8) without transliteration or script normalization beyond standard Unicode normalization (NFC). Article bodies were capped at 2,000 characters that covered mostly all the dataset at training and inference time to maintain feasible sequence lengths for 4B-parameter models while preserving sufficient semantic context and pilot experiments showed that increasing the limit beyond 2,000 characters yielded marginal accuracy gains. The dataset is pre-split into 2,078 training samples and 520 test samples, with no overlap between splits. Both splits are perfectly balanced: 50% fake, 50% real. We verified that the train and test splits contain zero duplicate article bodies. Only 37 headlines (7.1% of the test set) appear in both splits, all carrying identical labels and corresponding to distinct article bodies, indicating no label leakage. Table I summarizes dataset statistics.

Table I. Dataset Statistics

Split	**Total**	**Fake (YES)**	**Real (NO)**
Train	2,078	1,039	1,039
Test	520	260	260
Total	2,598	1,299	1,299

Each sample contains four fields: (1) headline, the news headline in Bangla, (2) input, the full article body in Bangla, (3) output, the binary label (YES for fake, NO for real), and (4) instruction, a standard prompt template directing the model to classify the article. This structured format supports direct use with instruction-tuned LLMs using standard chat templates, enabling consistent formatting across all three pipeline configurations.

IV. Methodology

A. Base Model: Qwen3.5-4B

All three pipelines use Qwen3.5-4B [9] as the base language model. Qwen3.5-4B is a 4-billion parameter decoder-only transformer pre-trained on a multilingual corpus of over three trillion tokens spanning more than 100 languages, including South and Southeast Asian languages. The model employs Grouped-Query Attention (GQA) [27] with 8 query heads per key-value head, reducing KV cache memory by 87.5% compared to multi-head attention at equivalent model size. Rotary Position Embeddings (RoPE) [28] with base frequency 10,000 provide length generalization up to 32,768 tokens. For deployment, the model is quantized to Q4_K_M using the GGUF format [29] 4-bit k-quant block quantization that reduces disk size from ~9.2 GB (FP16) to ~2.71 GB while retaining >99% of FP16 classification accuracy. Inference is served via the Ollama HTTP API [30], which applies the Qwen3.5 chat template through the RENDERER and PARSER directives.

B. Vanilla Zero-Shot Inference

The zero-shot baseline evaluates Qwen3.5-4B without any task-specific adaptation. Each test sample is formatted as a two-role conversation, system and user, using the Qwen3.5 chat template. The system message establishes the classification role and response format constraint: the model is instructed to respond with only YES (fake) or NO (real) without explanation. The user message contains the article text truncated to 2,000 characters prefixed with a classification instruction. Inference parameters are set to temperature 0.1 and top-p 0.9 for near-deterministic generation, with num_predict 64 to limit output length. The raw model output is parsed with a five-stage cascade: exact string match, substring scan, regex pattern, Bangla keyword lookup, fallback to NO.

C. Retrieval-Augmented Generation

The RAG pipeline augments each inference request with five semantically similar, pre-labeled training articles retrieved from a Pinecone serverless vector database [31]. At indexing time, all 2,078 training articles are encoded with paraphrase-multilingual-MiniLM-L12-v2 [22] a compact (22 M parameter), 384-dimensional multilingual sentence encoder and stored as vectors with label metadata (YES/NO and headline). Importantly, test articles are never indexed, preventing data leakage through near-duplicate retrieval.

At inference time, the test article is encoded using the same embedding model, and the top-5 most similar training vectors are retrieved by cosine similarity from the Pinecone index. Their full text and labels are fetched from the local training dataset using the vector ID as a lookup key. These five examples are formatted as a structured few-shot prompt preceding the target article, with each example explicitly labeled with its ground-truth verdict. The RAG system message instructs the model to use the retrieved examples as reference before classifying. This design operationalizes the in-context learning paradigm [19] with dynamically selected, semantically relevant examples rather than fixed hand-curated demonstrations.

D. Lora Fine-Tuning

LoRA [11] enables task-specific adaptation of Qwen3.5-4B while updating only 0.93% of its parameters. For a pre-trained weight matrix $W \in \mathbb{R}^{d\times k}$, LoRA introduces the adapted weight $W' = W + BA$, where $B \in \mathbb{R}^{d\times r}$, $A \in \mathbb{R}^{r\times k}$, and rank $r \ll \min(d, k)$. Only A and B are updated during gradient descent; W is frozen. The rank-r update is scaled by α/r (with $\alpha = r = 32$, yielding unit scaling). This parameterization reduces trainable parameters from 4.58 B to 42.5 M while preserving the full expressive capacity of the pre-trained representations in the frozen layers.

We apply LoRA to all seven linear projection layers of each transformer block: query (q_proj), key (k_proj), value (v_proj), output (o_proj), gate (gate_proj), up (up_proj), and down (down_proj) projections. Targeting MLP layers alongside attention projections provides additional adaptation capacity for the significant domain shift from general multilingual pre-training to Bangla fake news classification. Training employs AdamW with learning rate 2×10^{-4}, cosine decay schedule, effective batch size 8, 5 training epochs, and 10 linear warmup steps. The Unsloth library provides approximately 2× training speedup through custom backward-pass kernels and Flash Attention 2 integration. The trainings are performed on the NVIDIA RTX PRO 6000 Blackwell Server Edition GPU (94.97 GB VRAM).

Each training example is formatted using the Qwen3.5 chat template with three explicit turns: a system turn establishing the classification role, a user turn containing the article text and instruction, and an assistant turn containing the ground-truth YES/NO label. After training, LoRA adapters are merged into the base model weights at BF16 precision, the merged model is converted to full BF16, and Q4_K_M quantization is applied for GGUF export. The quantized model is registered in Ollama under the identifier qwen3.5:4b-bnfake-v1.

E. Experimental Setup

All experiments share a common infrastructure.for end-to-end pipeline; PyTorch provides the tensor backend; Transformers[32] handles model loading and tokenization; PEFT [24] manages LoRA adapter configuration; TRL provides the SFTTrainer for supervised fine-tuning; and Unsloth [26] accelerates training. The RAG pipeline uses sentence-transformers [22] for embedding and the Pinecone Python client [31] for vector operations. All classification metrics are computed with scikit-learn.

Inference for all pipelines is served via Ollama on a local

server with the Qwen3.5 RENDERER and PARSER directives applied. Inference parameters are uniform across pipelines: temperature 0.1, top-p 0.9, num_predict 64. The Pinecone serverless index is hosted on cloud with cosine distance metric and 384-dimensional vectors. Each evaluation runs logs per-sample predictions, raw model outputs, and ground-truth labels to a structured JSON report, enabling post-hoc error analysis. The 95% confidence intervals for accuracy are estimated using the normal approximation to the binomial distribution: $CI = p \pm 1.96\sqrt{(p(1-p)/n)}$, where $n = 520$

V. Results and Analysis

Table II presents overall performance across all three pipeline configurations evaluated on the 520-sample balanced test set. All adaptation techniques substantially improve over the vanilla baseline. RAG achieves a 6.73 pp accuracy gain at zero training cost, while Qwen3.5-4B-BnFake-V1 achieves 99.04% accuracy (macro F1 = 0.9904) highest among the pipelines.

Table II. Overall Performance Comparison

Pipeline	Accuracy	F1 Macro	Errors/ 520
Qwen3.5-4B (Vanilla)	73.85%	0.7362	136
Qwen3.5-4B + RAG	80.58%	0.8058	101
Qwen3.5-4B-BnFake-V1	99.04%	0.9904	5

The performance hierarchy is: Qwen3.5-4B-BnFake-V1 (99.04%) > RAG (80.58%) > Vanilla (73.85%). Qwen3.5-4B-BnFake-V1 95% confidence interval is [97.54%, 100.00%], which does not overlap with any other pipeline confirming statistical significance of the improvement. All three pipelines substantially exceed the random baseline (50.00% on a balanced dataset), with Qwen3.5-4B-BnFake-V1 outperforms every other pipeline and has a significant difference with the vanilla setttings.

To ensure robustness, we conducted three independent fine-tuning runs with identical hyperparameters and different random seeds. The mean accuracy across runs is 99.23% ± 0.19% (standard deviation), confirming the consistency of the LoRA adaptation. Table III shows detailed metrics of the three runs.

Table III. Results Across Three Runs for Qwen3.5-4B-BnFake-V1

Qwen3.5-4B-BnFake-V1	Accuracy	F1 Macro	Errors/ 520
Run 1	99.04%	0.9904	5
Run 2	99.23%	0.9923	4
Run 3	99.42%	0.9942	3

Table IV disaggregates performance by class. The most striking pattern is the systematic shift in recall balance across pipelines. The vanilla model is real-biased: it misses 35.38% of fake articles (64.62% fake recall) while correctly identifying 83.08% of real ones, an 18.46 pp asymmetry. This bias reflects the model's pre-trained tendency to assign lower uncertainty to real-sounding text, as multilingual LLMs are predominantly trained on real-world corpora. RAG nearly eliminates this asymmetry, achieving 80.38% fake recall and 80.77% real recall (0.39 pp gap), making it the most balanced pipeline.

Table IV. Per-Class Precision, Recall, and F1 (F = Fake, R = Real)

Pipeline	F.Precision	F.Recall	F.F1	R.Precision	R.Recall	R.F1
Qwen3.5-4B (Vanilla)	0.7925	0.6462	0.7119	0.7013	0.8308	0.7606
Qwen3.5-4B + RAG	0.8069	0.8038	0.8054	0.8046	0.8077	0.8061
Qwen3.5-4B-BnFake-V1	0.9811	1.0000	0.9904	1.0000	0.9808	0.9903

Qwen3.5-4B-BnFake-V1 achieves 100% fake recall (zero missed fakes) and 98.08% real recall simultaneously, with fake precision 0.9811 and perfect real precision 1.0000. Qwen3.5-4B-BnFake-V1 thus eliminates both the precision-recall trade-off and the class asymmetry observed in all other pipelines. All pipelines are designed as independent ablations against the unadapted base model the vanilla vs RAG measures the contribution of retrieval augmentation, and vanilla vs LoRA measures the contribution of weight-based adaptation.

We note that the two methods operate under different paradigms where LoRA modifies model weights through training and RAG augments prompts at inference time without weight modification. As such, each method is evaluated independently against the vanilla baseline rather than in direct competition with each other. The choice between RAG and fine-tuning depends on deployment constraints and application requirements. RAG requires no model training; the setup is limited to corpus indexing (~4 minutes for 2,078 documents) and deployment of the unchanged base model. It provides inherent transparency: retrieved examples are visible in the prompt and auditable by downstream reviewers. The knowledge base can be updated by modifying the vector index without any retraining. However, RAG introduces per-query overhead: embedding (~0.5 s) plus retrieval (~0.1 s) per sample, and the longer prompts (5 examples × 2,000 chars ≈ 10,000–15,000 chars) approximately double inference latency. It achieves 80.58% accuracy with the best class balance.

LoRA fine-tuning (Qwen3.5-4B-BnFake-V1) requires a training pipeline with significant GPU time and produces a modified model artifact (~2.71 GB GGUF), but achieves near-perfect accuracy (99.04%) with zero missed fakes and only 5 false alarms. For content moderation at scale, the low

false-negative rate is decisive: every fake article not caught imposes reputational and societal costs that dwarf the cost of a false alarm. For high-accuracy, high-volume deployment, Qwen3.5-4B-BnFake-V1 is the clear choice. For rapid prototyping, resource-constrained settings, or applications requiring explainable retrieval evidence, RAG provides a compelling baseline without model training.

VI. Conclusion

This paper presented a systematic comparative evaluation of three complementary adaptation strategies for Bangla fake news detection using the Qwen3.5-4B multilingual LLM: vanilla zero-shot inference, Retrieval-Augmented Generation, and LoRA fine-tuning. Experiments on a balanced corpus of 2,598 Bangla news articles, yielded three key findings.

First, RAG improves accuracy by 6.73 pp over the zero-shot baseline and achieves near-perfect class balance (0.39 pp recall gap), making it the preferred approach when model training is infeasible or model transparency is required.

Second, LoRA fine-tuning with 0.93% trainable parameters achieves 84.04% accuracy under standard evaluation, rising to 99.04% (F1 = 0.9904) when training-inference format consistency is enforced surpassing previously reported Bangla fake news detection results.

Third, the Qwen3.5-4B-BnFake-V1 model achieves 100% fake recall (zero missed fakes) with only 5 false positives approaching theoretical performance on the current dataset.

Future work will explore every pipelines with extending the dataset with multi-modal signals (embedded images, source metadata, and URL features); cross-lingual transfer to structurally similar low-resource languages; continual learning approaches to update the fine-tuned model incrementally without full retraining; and calibration methods to produce reliable prediction confidence scores alongside binary labels, enabling risk-stratified human review workflows.

References

[1] L. W. Abbott and D. Snidal, “Engaging the Public and the Private in Global Sustainability Governance,” International Affairs, vol. 88, no. 3, 2012, pp. 543–564.

[2] F. Menczer and T. Hills, "Information overload helps fake news spread, and social media knows it," Scientific American, vol. 321, no. 6, pp. 54–61, 2019.

[3] A. Bondielli and F. Marcelloni, "A survey on fake news and rumour detection techniques," Inf. Sci., vol. 497, pp. 38–55, 2019.

[4] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, "Fake news detection on social media: A data mining perspective," ACM SIGKDD Explor. Newsl., vol. 19, no. 1, pp. 22–36, 2017.

[5] Ethnologue, “Bengali,” 2026. [Online]. Available: https://www.ethnologue.com/language/ben/

[6] F. Alam, A. Hasan, T. Alam, A. Khan, J. Tajrin, N. Khan, and S. A. Chowdhury, “A Review of Bangla Natural Language Processing Tasks and the Utility of Transformer Models,” arXiv preprint arXiv:2107.03844, 2021.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT, 2019, pp. 4171–4186.

[8] A. Conneau et al., "Unsupervised cross-lingual representation learning at scale," in Proc. ACL, 2020, pp. 8440–8451.

[9] Qwen Team, "Qwen3 technical report," arXiv:2505.09388, 2025.

[10] P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in Proc. NeurIPS, vol. 33, 2020, pp. 9459–9474.

[11] E. J. Hu et al., "LoRA: Low-rank adaptation of large language models," in Proc. ICLR, 2022.

[12] W. Y. Wang, “ ‘Liar, Liar Pants on Fire’: A New Benchmark Dataset for Fake News Detection,” in Proc. 55th Annual Meeting of the Association for Computational Linguistics, Vol. 2: Short Papers, Vancouver, Canada, 2017, pp. 422–426.

[13] K. Shu, D. Mahudeswaran, S. Wang, D. Lee, and H. Liu, "FakeNewsNet: A data repository with news content, social context and spatial information," Big Data, vol. 8, no. 3, pp. 171–188, 2020.

[14] T. Thorne et al., "FEVER: A large-scale dataset for fact extraction and verification," in Proc. NAACL-HLT, 2018, pp. 809–819.

[15] Z. Jiang et al., "How can we know what language models know?," Trans. Assoc. Comput. Linguist., vol. 8, pp. 423–438, 2020.

[16] K. Nakamura, S. Levy, and W. Y. Wang, "r/Fakeddit: A new multimodal benchmark dataset for fine-grained fake news detection," in Proc. LREC, 2020, pp. 6149–6157.

[17] M. Z. H. George, N. Hossain, M. R. Bhuiyan, A. K. M. Masum, and S. Abujar, “Bangla Fake News Detection Based On Multichannel Combined CNN-LSTM,” in 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), 2021, pp. 1–5.

[18] Hossain et al., “BanFakeNews: A Dataset for Detecting Fake News in Bangla,” in Proc. Twelfth Language Resources and Evaluation Conference, Marseille, France, 2020, pp. 2862–2871.

[19] T. Brown et al., "Language models are few-shot learners," in Proc. NeurIPS, vol. 33, 2020, pp. 1877–1901.

[20] V. Karpukhin et al., "Dense passage retrieval for open-domain question answering," in Proc. EMNLP, 2020, pp. 6769–6781.

[21] F. Shi et al., "Language models are multilingual chain-of-thought reasoners," in Proc. ICLR, 2023.

[22] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-networks," in Proc. EMNLP-IJCNLP, 2019, pp. 3982–3992.

[23] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks," in Psychol. Learn. Motiv., vol. 24. Academic Press, 1989, pp. 109–165.

[24] S. Mangrulkar et al., "PEFT: State-of-the-art parameter-efficient fine-tuning methods," GitHub, 2022. [Online]. Available: https://github.com/huggingface/peft

[25] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "QLoRA: Efficient finetuning of quantized LLMs," in Proc. NeurIPS, vol. 36, 2023.

[26] Unsloth AI, "Unsloth: 2x faster LLM fine-tuning," GitHub, 2024. [Online]. Available: https://github.com/unslothai/unsloth

[27] J. Ainslie et al., "GQA: Training generalized multi-query transformer models from multi-head checkpoints," in Proc. EMNLP, 2023, pp. 4895–4901.

[28] J. Su et al., "RoFormer: Enhanced transformer with rotary position embedding," Neurocomputing, vol. 568, p. 127063, 2024.

[29] G. Gerganov, "GGML: Tensor library for machine learning," GitHub, 2023. [Online]. Available: https://github.com/ggerganov/ggml

[30] Ollama, "Run large language models locally," 2024. [Online]. Available: https://ollama.com

[31] Pinecone Systems Inc., "Pinecone: The vector database for ML applications," 2024. [Online]. Available: https://www.pinecone.io

[32] T. Wolf et al., "Transformers: State-of-the-art natural language processing," in Proc. EMNLP: System Demonstrations, 2020, pp. 38–45.

ASM. H. Kabir, Moscow Institute of Physics and Technology (National Research University), Moscow Oblast, Russia, 141701 (e-mail: humaun.kabir@phystech.edu)

S. A. Khan, Innopolis University, Innopolis, Republic of Tatarstan, Russia, 420500 (e-mail: sameedkhandurrani@gmail.com)

Alexander A. Kharlamov, Moscow Institute of Physics and Technology (National Research University), Moscow Oblast, Russia, 141701 (e-mail: kharlamov@analyst.ru)

Ilia M. Voronkov, Moscow Institute of Physics and Technology (National Research University), Moscow Oblast, Russia, 141701 (e-mail: voronkov.im@phystech.edu)