U-RoCX: An xLSTM based Approach to AI-Generated Urdu Text Detection

Rabee Adel Al-Qasem

Abstract

Large Language Models (LLMs) have rapidly proliferated, presenting challenges in distinguishing human-written text from AI-generated content, especially in low-resource languages like Urdu. This paper introduces U-RoCX, a novel hybrid architecture for the AbjadGenEval Shared Task on AI-Generated Urdu Text Detection. U-RoCX combines the multilingual semantic capabilities of a frozen XLM-RoBERTa backbone with local feature extraction from Convolutional Neural Networks (CNNs) and the advanced sequential modeling of the recently proposed Extended LSTM (xLSTM). By utilizing xLSTM's matrix memory and covariance update rules, the model addresses traditional Recurrent Neural Network bottlenecks. Experimental results demonstrate the robustness of U-RoCX, achieving a balanced accuracy and F1-score of 88% on the test set.

1 Introduction

The post-ChatGPT era has transformed practices across multiple fields, generating significant positive impacts and driving continual improvements in large language model (LLM) architectures (Liu et al., 2026; Zhang et al., 2026; Xie et al., 2025; Odeh and Natsheh, 2025). However, it has also introduced significant ethical concerns and inherent biases (Hasan et al., 2025), alongside challenges in areas such as academic writing and the proliferation of fake news (Al-Jarf, 2024; Alnsour et al., 2025; Ead, 2024). These issues have led to the emergence of new research domains focused on detecting AI-generated writing in these contexts.

This paper presents our work in AbjadGenEval - Task 2 on AI-Generated Urdu Text Detection (Abudalfa et al., 2025; Ezzini et al., 2026), which focuses on developing an AI model to detect the text generated by LLMs that focus only on detecting Urdu language,which is a language widely used by more than 100 million people in the world, where many people share their tweets,reviews, and comments in Urdu (Yin et al., 2024; Zheng et al., 2023).

This study introduces U-RoCX, an AI model designed to detect AI-generated content in Urdu. U-RoCX integrates convolutional neural networks (CNNs) with the recently proposed xLSTM (Beck et al., 2024), utilizing the XLM-RoBERTa backbone as the primary embedding model (Conneau et al., 2020). **To the best of our knowledge, based on extensive research, we are the first to experiment with and evaluate the performance of xLSTM for binary classification specifically in the Urdu language.** Experimental results on the shared task dataset demonstrate strong performance, achieving an F1 score of 88% and balanced accuracy of 88%. The code and training pipeline are publicly available on Github.

2 Shared Task Background

The primary task is AI writing detection (Lamsiyah et al., 2025), in which the model receives a text in Urdu and classifies it as human-written or AI-generated. Model performance is evaluated using standard binary classification metrics, including F1 score and accuracy. The dataset used for training and evaluating the model consists of 6,800 samples, split into 4,800 for training and 2,000 for testing. The training data is balanced between human-written content, collected from news platforms, and AI-generated content produced using multiple LLMs.

3 Proposed System Architecture

3.1 Data Preprocessing

Since we are working with text data, which is considered unstructured data, and it's common practice to do some preprocessing techniques on the raw text and convert it into a valuable and standardized part, we do this through data cleaning to remove the insignificant and useless parts of the text that

Proceedings of the 2nd Workshop on NLP for Languages Using Arabic Script (AbjadNLP 2026), pages 443–447

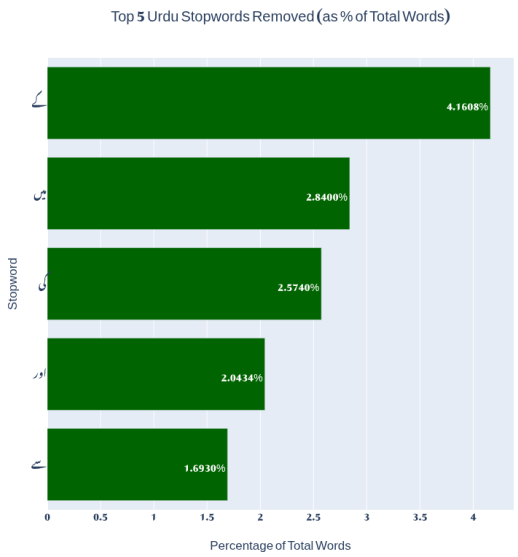

Figure 1: Top 5 most frequent Urdu stopwords found in the training data, shown as a percentage of the total word count.

might make the performance of any AI model that deals with text worse (Ahmed et al., 2024).

We processed our data using the UrduHack Python library, an NLP library designed specifically for the Urdu language, utilizing its stopword and normalization functionalities.

3.1.1 Text Cleaning and Normalization

The initial step in the cleaning pipeline involved removing elements such as URLs, hashtags, and emojis from the text using regular expressions. Subsequently, text and whitespace normalization were applied to ensure consistent Unicode representation throughout the dataset.

3.1.2 Stopword Removal

The second step in the cleaning pipeline is the removal of stop words from the text. Stop words, which are the most frequently used words in any language, are typically removed to reduce the feature space and enable the model to focus on more informative terms. A list of Urdu stop words was removed from the dataset, as illustrated in Figure 1.

3.1.3 Stemming

The final phase involves stemming the text to address the morphological complexity of the Urdu language. The approach described in (Ahmed et al., 2024) was adopted, implementing a function that reduces words to their root forms by removing common affixes. The stemmer iterates through predefined, sorted lists of Urdu prefixes and suffixes, prioritizing longer affixes to ensure accurate matching. A safety check ensures that affixes are only removed if the remaining stem retains a minimum length of two characters. This method effectively manages morphological variations while preventing excessive truncation of valid root words.

3.2 U-RoCX Model Architecture

The proposed U-RoCX system is a hybrid neural architecture designed to leverage the semantic capabilities of large language models while maintaining the computational efficiency of recurrent networks. As illustrated in Figure 2, the model pipeline is implemented using PyTorch and consists of four stages: embedding backbone, local feature extraction using CNN, sequential modeling using the new implementation of Xlstm , and binary classification head.

3.2.1 Backbone and Embedding Strategy

Consistent with related studies (Saleem et al., 2025; Ammar et al., 2025), **XLM-RoBERTa** was utilized as the primary embedding layer in the model, as it supports multilingual text, including Urdu. XLM-RoBERTa represents each token as a 768-dimensional vector. Additionally, we decided to freeze the weights of the xlm-Roberta model during training, and we noticed that doing this has helped us to improve our prediction accuracy and decrease the time needed to train the full model

3.2.2 Convolutional Feature Extraction

To capture local n-gram patterns and reduce dimensionality before input to the xLSTM architecture, a simple 1D Convolutional Neural Network is inserted. The 768-dimensional embeddings are first compressed to 128 channels via linear projection. A single convolutional layer scans the sequence with 128 filters of width 3, modeling trigram context. Subsequently, a max-pooling kernel of size 2 halves the sequence length, enabling the downstream xLSTM blocks to operate on a more focused representation.

3.2.3 Extended Long Short-Term Memory (xLSTM)

In this study, we investigate the Extended LSTM (xLSTM) architecture, a recent advancement that modifies the traditional LSTM structure. xLSTM introduces two main innovations: (i) sLSTM, featuring a scalar memory, scalar update, and new memory mixing; and (ii) mLSTM, which utilizes

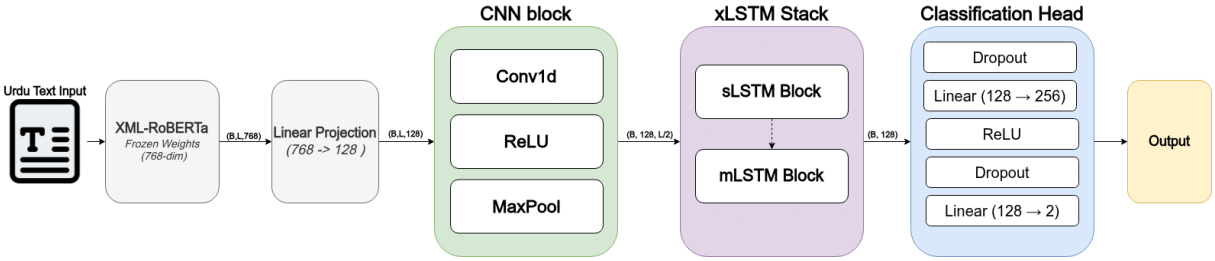

Figure 2: The overall architecture of the proposed U-RoCX model, illustrating the flow from Urdu text input through the frozen XLM-RoBERTa backbone, the CNN feature extraction block, the sequential xLSTM stack, and finally the classification head.

a matrix memory and covariance update rule to enable full parallelizability. These modifications address key limitations of standard LSTMs, specifically the inability to revise storage decisions, limited storage capacities which force information compression into scalar cell states, and a lack of parallelizability due to sequential processing (Beck et al., 2024). By overcoming these bottlenecks, which previously allowed attention mechanisms to outperform RNNs, we aim to evaluate the efficacy of xLSTM for the binary classification of Urdu text.

3.2.4 Classification Head

The final stage is a fully connected classification head. The output from the last hidden state of the xLSTM stack is passed to a small dense projection to 256 units with ReLU activation. The final linear layer maps these features to the binary classes, producing the final logits for optimization.

The previous architecture resulted in a highly efficient model structure. While the complete U-RoCX architecture comprises 278 millions parameters, the strategic freezing of the XLM-RoBERTa backbone restricts the optimization landscape significantly. Consequently, only 399,114 parameters specifically those associated with the CNN, xLSTM, and classification head are trainable. This design allows the model to leverage deep, pretrained linguistic knowledge while maintaining a lightweight training model.

4 Experimental Setup

4.1 Model Configuration

To optimize the U-RoCX architecture for Urdu text classification, the maximum context length was dynamically determined by the 99th percentile of the training data's length distribution, minimizing padding overhead while avoiding truncation. We configured the CNN layer with 128 filters and a kernel size of 3, followed by a max-pooling operation with a kernel size of 2. For the sequential modeling, we utilized a stack of 2 xLSTM blocks with 4 attention heads. We adopted a hybrid configuration by alternating the block types, positioning an sLSTM block first for scalar memory updates, followed by an mLSTM block to leverage matrix memory parallelization. Finally, a dropout rate of 0.3 was applied to the classification head with ReLU activation.

4.2 Training Strategy

For the training procedures of our model, we decided to go with the Cross-Entropy loss function as our main loss function. We utilized the AdamW optimizer (Loshchilov and Hutter, 2019), and to ensure stable convergence, we adopted the One Cycle Learning Rate (OneCycleLR) policy (Smith, 2018). This strategy dynamically adjusts the learning rate, starting from a lower initial value, increasing to a maximum during a warmup phase, and then annealing it down for the remainder of the training steps.

4.3 Hardware Environment

All training and evaluation experiments were conducted NVIDIA Tesla T4 GPU with 15 GB of VRAM. This hardware setup provided sufficient memory bandwidth to handle the combined computational load of the XLM-RoBERTa embeddings and the xLSTM operations.

5 Results

The performance of the U-RoCX model was evaluated on the official test set. We utilized standard binary classification metrics: Accuracy, Precision, Recall, F1-Score, and Balanced Accuracy.

As shown in Table 1, our proposed architecture achieved a Macro F1-score of 88.04% and a Balanced Accuracy of 88.03%. These results indicate

that the model maintains robustness across both classes (Human-written and AI-generated) without significant bias. The close proximity of the Accuracy (88.02%) and Balanced Accuracy scores further confirms the model's stability.

The Recall of 88.61% is particularly notable, suggesting that the model is highly effective at identifying positive samples (AI-generated text) with relatively few false negatives.

Metric	Score (%)
Accuracy	88.02
Balanced Accuracy	88.03
Precision	87.47
Recall	88.61
F1-Score	**88.04**

Table 1: Detailed performance metrics of the U-RoCX model on the test dataset.

6 Conclusion

In this paper, we presented U-RoCX, a novel approach for detecting AI-generated Urdu text in the context of the AbjadGenEval Shared Task. By integrating the linguistic power of XLM-RoBERTa with the advanced memory capabilities of xLSTM and local feature extraction via CNNs, we developed a system that is both computationally efficient and accurate.

Our experiments demonstrate that xLSTM is a viable and powerful alternative to standard recurrent units for low-resource language tasks, achieving an F1 score of 88.04%. To the best of our knowledge, this is the first application of xLSTM for Urdu binary text classification.

References

Shadi Abudalfa, Saad Ezzini, Ahmed Abdelali, Hamza Alami, Abdessamad Benlahbib, Salmane Chafik, Mo El-Haj, Abdelkader El Mahdaouy, Mustafa Jarrar, Salima Lamsiyah, and Hamzah Luqman. 2025. The AraGenEval shared task on Arabic authorship style transfer and AI generated text detection. In *Proceedings of The Third Arabic Natural Language Processing Conference at EMNLP 2025*, Suzhou, China. Association for Computational Linguistics.

Naeem Ahmed, Rashid Amin, Hamza Aldabbas, Muhammad Saeed, Muhammad Bilal, and Houbing Song. 2024. A novel approach for sentiment analysis of a low resource language using deep learning models. *ACM Transactions on Asian and Low-Resource Language Information Processing*.

Reima Al-Jarf. 2024. Students' assignments and research papers generated by ai: Arab instructors' views. *Online Submission*, 6(2):92–98.

Marwa M Alnsour, Latifa Qouzah, Sanaa Aljamani, Rasha A Alamoush, and Mahmoud K AL-Omiri. 2025. Ai in education: enhancing learning potential and addressing ethical considerations among academic staff—a cross-sectional study at the university of jordan. *International Journal for Educational Integrity*, 21(1):16.

Muhammad Ammar, Hadiya Murad Hadi, and Usman Majeed Butt. 2025. Ai-generated text detection in low-resource languages: A case study on urdu. *arXiv preprint arXiv:2510.16573*.

Maximilian Beck, Korbinian Pöppel, Markus Spanring, Andreas Auer, Oleksandra Prudnikova, Michael Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. 2024. xlstm: Extended long short-term memory. *Advances in Neural Information Processing Systems*, 37:107547–107603.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 8440–8451.

Hamed Abdelreheem Ead. 2024. Exploring the impact of artificial intelligence on academic writing: Perspectives of ph. d. students in the faculty of science at cairo university.

Saad Ezzini, Irfan Ahmed, Salmane Chafik, Shadi Abudalfa, Mo El-Haj, Ahmed Abdelali, Mustafa Jarrar, Nadir Durrani, Hassan Sajjad, and Farah Adeeba. 2026. Abjadgeneval: Abjad ai generated text detection shared task for languages using arabic script at abjadnlp 2026. In *Proceedings of the 2nd Workshop on NLP for Languages Using Arabic Script (AbjadNLP 2026), co-located with the 19th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2026)*, Rabat, Morocco.

Dana Hasan, Amal Nazzal, and Sulafa Zidani. 2025. Beating algorithmic discrimination: Maneuvering digital surveillance to indigenize the narrative. *International Journal of Communication*, 19:23.

Salima Lamsiyah, Saad Ezzini, Abdelkader El Mahdaouy, Hamza Alami, Abdessamad Benlahbib, Samir El amrany, Salmane Chafik, and Hicham Hammouchi. 2025. M-DAIGT: A Shared Task on Multi-Domain Detection of AI-Generated Text. In *Proceedings of the Shared Task on Multi-Domain Detection of AI-Generated Text*, pages 1–9, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Shih-Yang Liu, Xin Dong, Ximing Lu, Shizhe Diao, Peter Belcak, Mingjie Liu, Min-Hung Chen, Hongxu Yin, Yu-Chiang Frank Wang, Kwang-Ting Cheng,

and 1 others. 2026. Gdpo: Group reward-decoupled normalization policy optimization for multi-reward rl optimization. *arXiv preprint arXiv:2601.05242*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Sabri Odeh and Emad Natsheh. 2025. Toward better solar pv panel fault detection: A multi-ml approach for series and parallel hotspot analysis. *Engineering Research Express*.

Husnain Saleem, Muhammad Javed, and Junaid Khan. 2025. Hate speech identification in formal and informal social media text using roberta-base and xlm-roberta-base models. *BRAIN: Broad Research in Artificial Intelligence & Neuroscience*, 16(4).

Leslie N Smith. 2018. A disciplined approach to neural network hyper-parameters: Part 1–learning rate, batch size, momentum, and weight decay. *arXiv preprint arXiv:1803.09820*.

Zhenda Xie, Yixuan Wei, Huanqi Cao, Chenggang Zhao, Chengqi Deng, Jiashi Li, Damai Dai, Huazuo Gao, Jiang Chang, Liang Zhao, and 1 others. 2025. mhc: Manifold-constrained hyper-connections. *arXiv preprint arXiv:2512.24880*.

Lirong Yin, Lei Wang, Zhuohang Cai, Siyu Lu, Ruiyang Wang, Ahmed AlSanad, Salman A AlQahtani, Xiaobing Chen, Zhengtong Yin, Xiaolu Li, and 1 others. 2024. Dpal-bert: A faster and lighter question answering model. *CMES-Computer Modeling in Engineering & Sciences*, 141(1).

Yifan Zhang, Yifeng Liu, Mengdi Wang, and Quanquan Gu. 2026. Deep delta learning. *arXiv preprint arXiv:2601.00417*.

Wenfeng Zheng, Siyu Lu, Zhuohang Cai, Ruiyang Wang, Lei Wang, and Lirong Yin. 2023. Pal-bert: an improved question answering model. *Computer Modeling in Engineering & Sciences*, 10.