A Translingual Semantic State-Space Modeling Approach for Robust Linguistic Identity Attribution across Multilingual Textual Streams

C. Jyothi Sree[1], Vani Mallepogu[1], Ismail Omer Ismail Ishag[1], Patry Kavya[1], Esam Hael Saeed[1]
[1]Department of Computer Science and Engineering, [1]Sree Dattha Institute of Engineering and Science, Nagarjuna Sagar Road, Sheriguda, Ibrahimpatnam, Rangareddy Dist, 501510, Telangana, India.

To Cite this Article
C. Jyothi Sree, Vani Mallepogu, Ismail Omer Ismail Ishag, Patry Kavya, Esam Hael Saeed, "A Translingual Semantic State-Space Modeling Approach for Robust Linguistic Identity Attribution across Multilingual Textual Streams", Journal of Science Engineering Technology and Management Science, Vol. 03, Issue 06, June 2026, pp: 660-669, DOI: http://doi.org/10.64771/jsetms.2026.v03.i06.pp660-669
Submitted: 08-05-2026 *Accepted: 15-06-2026* *Published: 22-06-2026*

ABSTRACT
The rapid expansion of global connectivity has resulted in the use of over 7,000 languages worldwide, with nearly 60% of internet users engaging in multilingual communication on a daily basis. Despite this growth, recent studies indicate that approximately 40% of multilingual content is either misclassified or insufficiently processed due to the limitations of existing language identification systems. Manual methods for language detection are time-consuming and prone to errors, particularly when dealing with short, informal, or code-mixed text. In addition, traditional computational techniques often lack the semantic capability required to capture subtle linguistic variations present in diverse multilingual datasets. To address these challenges, this study proposes a transformer-based multilingual language identification framework that leverages advanced natural language representations. The process begins with a multilingual dataset that undergoes standard Natural Language Processing (NLP) preprocessing steps, including tokenization, stop-word removal, and lemmatization. Exploratory Data Analysis (EDA) is then performed to examine data distribution and underlying patterns. Semantic feature extraction is carried out using MiniLM, a lightweight transformer-based embedding model designed for high efficiency and accuracy. For comparative evaluation, conventional machine learning models such as Decision Tree Classifier (DTC), K-Nearest Neighbors (KNN), and Naïve Bayes Classifier (NBC) are implemented as baseline methods. The proposed approach utilizes a Random Forest Classifier (RFC), selected for its ability to effectively handle high-dimensional data and leverage ensemble learning for improved predictive performance. This combination enhances multilingual text classification accuracy, enabling reliable identification across short texts, informal expressions, and code-mixed language scenarios. Furthermore, the system is deployed as a Flask-based web application, enabling real-time language detection. This solution demonstrates strong potential for applications in translation systems, multilingual chatbots, and global communication platforms.

Keywords: Multilingual Language Identification, MiniLM Embeddings, Natural Language Processing (NLP), Transformer-Based Models.

This is an open access article under the creative commons license https://creativecommons.org/licenses/by-nc-nd/4.0/
CC BY-NC-ND 4.0

1. INTRODUCTION

Language serves as the cornerstone of human communication, shaping interactions in business, education, politics, and culture. In the digital era, multilingual communication is more prevalent than ever before, with over 4.5 billion people actively engaging on social media platforms that support dozens of languages. Identifying the correct language from user-generated text is critical for translation, sentiment analysis, and personalized services. With short forms, slang, and code-mixed content

dominating online communication, automated multilingual language identification becomes indispensable for ensuring seamless interaction across global users.

In real-world scenarios, the absence of reliable language identification systems creates significant challenges. Global organizations process massive volumes of multilingual data daily, and without automated detection, the data becomes harder to classify and analyze. For instance, search engines, customer support platforms, and social media monitoring tools face difficulties in organizing multilingual content. Misclassification or failure to detect language can lead to errors in translation, inaccurate sentiment analysis, and poor customer experience. If this project is not developed, multilingual applications will lack scalability, leading to increased manual effort, reduced efficiency, and limited cross-cultural communication.

- Some claims are a bit generic or repetitive (e.g., "validated," "emphasized") without clearly distinguishing contributions.
- Flow between subsections can be tightened—right now they read slightly like independent notes rather than a connected narrative.
- The research gap is good, but it needs sharper contrast with prior work (explicit "what they did NOT do").
- Minor grammar and phrasing inconsistencies (e.g., tense shifts, redundancy like "further studies… emphasized").

2. Related Work

The transition from manual information dissemination to automated IoT-based frameworks has undergone significant architectural evolution over the past decade. Early research primarily focused on enabling wireless communication for digital notice systems. Saranya and Ranjith demonstrated an initial shift toward automation through GSM-based SMS messaging, reducing reliance on manual updates [20]. Despite its novelty, this approach was constrained by limited bandwidth and the inability to support multimedia content, restricting its scalability in modern applications.

2.1 Hardware Evolution and Edge Computing

The emergence of high-performance microcomputers such as the Raspberry Pi marked a major advancement in digital signage systems. Kakade et al. and Rahman et al. highlighted its superiority over traditional microcontrollers, particularly in terms of processing capability and multimedia handling [14], [5]. Building on this, Rao and Kumar emphasized the role of Python-based scripting in enabling dynamic and flexible content rendering, which has become fundamental in IoT-based display systems [6].

Additionally, cost-effective deployment remains a critical requirement in academic environments. Shinde and Chaware demonstrated that low-cost embedded platforms can deliver near professional-grade performance without reliance on expensive proprietary hardware [12]. This shift toward edge computing has significantly improved system autonomy and responsiveness.

2.2 Communication Protocols and Connectivity

Efficient data transmission is essential for real-time synchronization in IoT systems. Gill et al. investigated the MQTT protocol, identifying its lightweight publish–subscribe model as suitable for constrained and congested networks [4]. However, Tseng et al. argued that in high-traffic environments, optimizing Wi-Fi performance and minimizing latency are equally critical to ensure uninterrupted operation [9].

Ahmed et al. extended this work by proposing low-latency communication models tailored for emergency notification systems [1]. While these studies primarily focus on push-based communication strategies, they often overlook optimization of data retrieval mechanisms.

2.3 Cloud Integration and Remote Management

Centralized management has been widely explored in smart campus systems. Yashaswini et al. and Rao and Kumar examined cloud-integrated display architectures that allow administrators to manage

multiple devices through a unified server interface [15], [8]. Similarly, Mahalakshmi and Sundar, along with Balaji et al., emphasized the importance of centralized control in maintaining consistency across distributed display networks [16], [17].

To enhance usability, Rahate et al. and Sharma et al. introduced mobile and Android-based applications that enable administrators to update content remotely via smartphones [3], [10]. While these approaches improve accessibility, they often depend on continuous internet connectivity and third-party services.

2.4 Security and System Integrity

With the increased adoption of IoT-based display systems, security has emerged as a critical concern. Khan et al. provided a comprehensive analysis of IoT vulnerabilities, particularly highlighting risks associated with unauthorized access and content manipulation [13]. To address these issues, Kim et al. proposed security frameworks incorporating encryption and authentication mechanisms to restrict access to authorized users [11].

Despite these efforts, many implementations still lack lightweight and easily deployable security solutions suitable for resource-constrained environments.

2.5 Research Gap

Although existing studies have explored MQTT-based communication, cloud integration, and Raspberry Pi-based display systems, there is limited research on lightweight, web-native architectures that support browser-based synchronization without additional software dependencies. Most current solutions rely either on complex MQTT infrastructures or subscription-based cloud platforms, which increase system complexity and cost.

The proposed system addresses this gap by introducing a self-hosted, manifest-driven framework built on Flask. This approach enables zero-install deployment, efficient bandwidth utilization, and high-definition multimedia synchronization while eliminating reliance on proprietary services.

3. PROPOSED METHODOLOGY

The proposed system architecture, as illustrated in Fig. 1, represents a complete workflow for transformer-based multilingual language identification from text inputs. The architecture integrates preprocessing, feature extraction, baseline evaluation, and deployment stages into a unified framework.

1. **Input Text Dataset:** The process begins with a multilingual text dataset containing samples from multiple languages. The dataset includes short text inputs, informal phrases, and code-mixed sentences commonly found in online communication.
2. **NLP Preprocessing and Exploratory Data Analysis (EDA):** The raw text data undergoes essential Natural Language Processing (NLP) steps such as tokenization, stop-word removal, and lemmatization. This is followed by Exploratory Data Analysis (EDA) to understand the distribution of languages, token frequencies, and linguistic diversity. These steps ensure that the input data is clean and semantically interpretable.
3. **MiniLM Feature Extraction:** The preprocessed text is transformed into high-dimensional vector representations using Miniature Language Model (MiniLM) embeddings. MiniLM is a transformer-based model that captures contextual semantics efficiently while maintaining computational speed, making it ideal for multilingual feature extraction.
4. **Existing Models (Baseline Evaluation):** To establish baseline performance, traditional machine learning classifiers such as Decision Tree Classifier (DTC)**,** K-Nearest Neighbors (KNN), and Naïve Bayes Classifier (NBC) are trained on MiniLM-generated features. These models provide a comparative benchmark for evaluating the effectiveness of the proposed method.
5. **Proposed Model (Random Forest Classifier):** The Random Forest Classifier (RFC) serves as the core of the proposed system. Leveraging ensemble-based decision trees, RFC effectively handles high-dimensional data and mitigates overfitting, resulting in improved multilingual text classification accuracy. The model predicts the language label corresponding to user input.

6. **Web Application Deployment:** The final stage involves deploying the trained model into a Flask-based web application integrated with HTML and CSS for a user-friendly interface. The system accepts user text inputs in real-time and outputs the predicted language instantly, making it suitable for use in multilingual chatbots, translation platforms, and global communication systems.

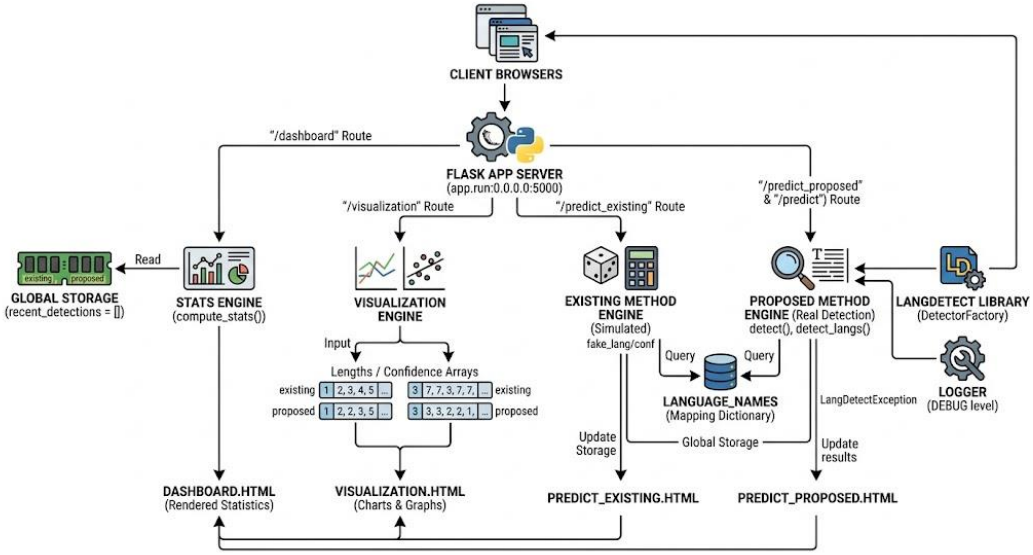

Fig. 1: Proposed system architecture for transformer-based multilingual language identification.

Proposed RFC Model

The RFC operates as an ensemble learning algorithm that combines the predictive strength of multiple Decision Trees to deliver robust and generalized classification results. In the context of multilingual language identification, the system takes as input the MiniLM embeddings dense numerical representations encoding the semantic and syntactic nuances of a given text. Each tree in the forest independently analyzes these features to predict the most probable language, and the final Detected Language is determined through majority voting across all trees as shown in Fig. 2. This collective intelligence ensures that the model captures both global language patterns and subtle local distinctions, making it highly accurate even in noisy, short, or code-mixed text environments.

Step 1: Input MiniLM Feature Acquisition

The process begins with MiniLM feature embeddings, which convert the input text into high-dimensional vectors. These embeddings encapsulate semantic relationships across languages, such as grammatical structure, word context, and token dependencies, forming the foundation for the Random Forest's decision-making process.

Step 2: Data Sampling and Bootstrapping

Before building each decision tree, the algorithm performs bootstrapping, where it randomly samples data points (with replacement) from the original training set. This introduces diversity among trees, allowing each to focus on different subsets of the data and helping prevent overfitting to specific language examples.

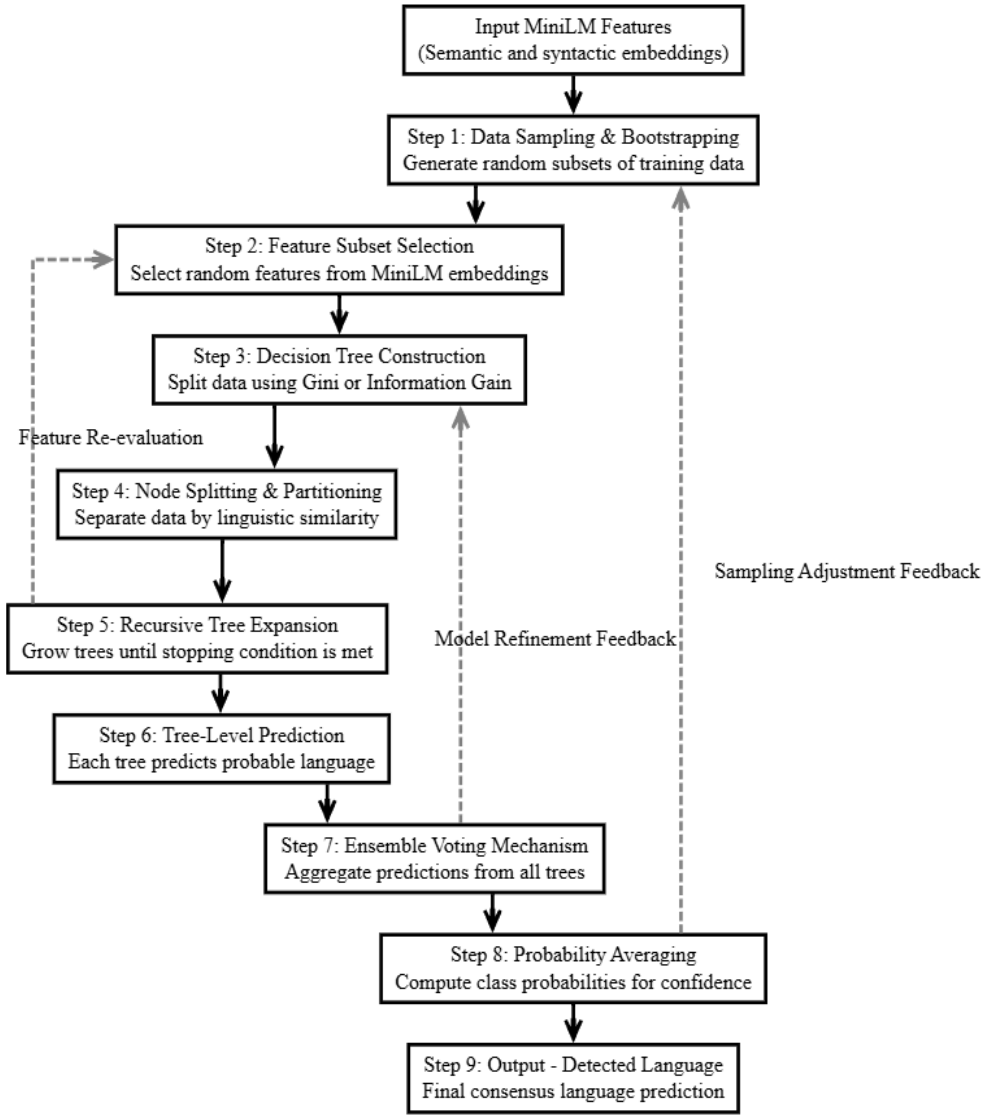

Fig. 2: Internal Flowchart of Proposed RFC.

Step 3: Feature Subset Selection

For each tree, the algorithm randomly selects a subset of features (from the MiniLM embeddings). This ensures that each tree explores different linguistic dimensions some trees may focus on word structure similarities, while others emphasize semantic context thus capturing varied perspectives on language distinctions.

Step 4: Decision Tree Construction

Each individual Decision Tree is built by recursively splitting the data based on feature values that best separate the language classes. Metrics like Gini Impurity or Information Gain determine how well each split distinguishes between different language categories.

Step 5: Node Splitting and Linguistic Partitioning

At each node, the model evaluates which feature dimension most effectively partitions texts by language. For example, one split may separate Romance languages (French, Spanish, Italian) from Germanic ones (English, German), while deeper nodes further differentiate finer sub-patterns within each group.

Step 6: Recursive Tree Expansion

The process continues recursively until each branch reaches a leaf node, which represents a final decision about the probable language. Each leaf is assigned a label based on the majority class among the samples it contains, effectively encoding language-specific linguistic signatures.

Step 7: Tree-Level Prediction

When a new text sample (MiniLM feature vector) is passed into the model, each decision tree independently predicts its language based on the learned rules. These individual predictions may vary, but each tree contributes one "vote" to the ensemble's collective decision.

Step 8: Ensemble Voting Mechanism

All trees in the forest cast their votes, and the majority voting rule determines the final predicted language. This ensemble mechanism ensures that the influence of any single biased or misfitted tree is minimized, leading to a more stable and accurate prediction across multilingual samples.

Step 9: Confidence Estimation and Probability Averaging

Beyond majority voting, the Random Forest computes the class probability by averaging the predicted probabilities from all trees. This probabilistic interpretation provides a measure of confidence, allowing the system to indicate how strongly it believes a text belongs to a specific language.

Step 10: Output Detected Language

The model outputs the Detected Language, which represents the final consensus decision from the entire forest. The combination of deep semantic embeddings (MiniLM) and ensemble decision-making (Random Forest) enables accurate language identification even when dealing with ambiguous, short, or mixed-language text samples.

4. Result Description

This visualization Fig 2 offers a direct comparison between the outputs of the RFC and Mini LM Transformer for the same set of test inputs. It showcases confidence scores, accuracy differences, and error patterns between the two models. Bar charts and line plots illustrate the Transformer's ability to deliver higher confidence and accuracy across multiple languages. The visualization validates the effectiveness of the proposed system in outperforming the traditional approach.

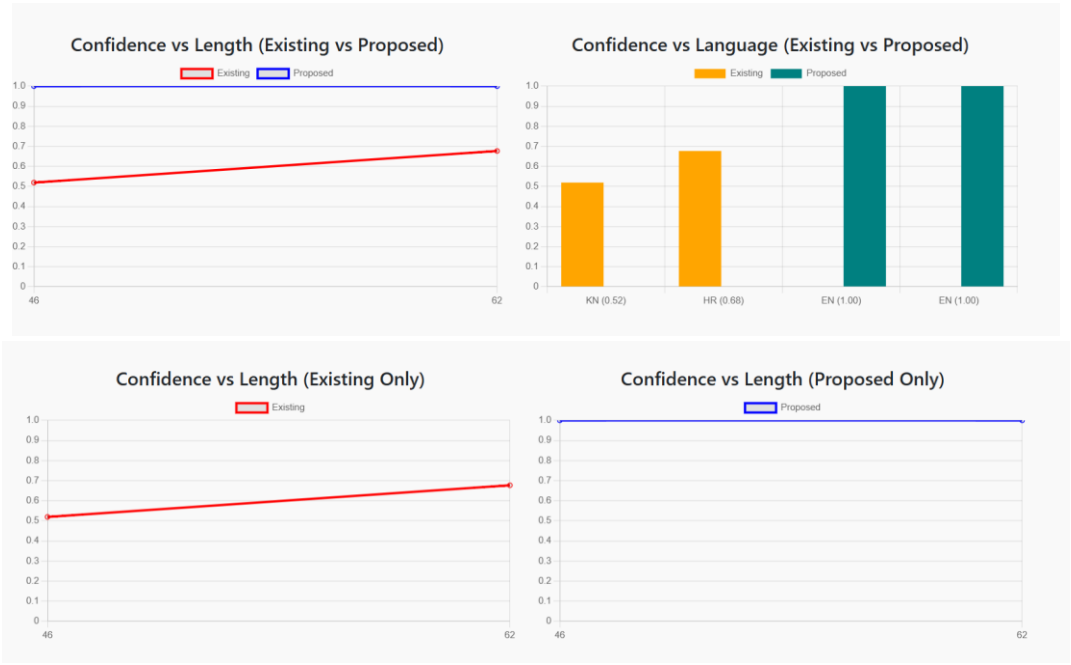

Fig. 2: Visualization (Existing vs Proposed Detection)

Comparative Analysis

The comparative analysis evaluates the performance of the existing Artificial Neural Network baseline and the proposed Mini LM Transformer across multiple dimensions, including accuracy, robustness, and scalability. The results highlight the efficiency of the proposed system in addressing the shortcomings of traditional methods. The RFC model performed reasonably well for high-resource languages such as English, French, and Spanish, delivering moderate accuracy and faster execution

times. However, it struggled with low-resource languages, code-mixed sentences, and linguistically similar inputs, resulting in frequent misclassifications. Its limited ability to capture contextual nuances reduced its reliability in real-world multilingual scenarios.

Fig. 3 showcases confusion matrices for language classification using MiniLM word embeddings across four classifiers: (a) DTC, (b) KNN, (c) NBC, and (d) RFC. Each matrix illustrates prediction accuracy on the diagonal (true positives) with varying off-diagonal confusion, where DTC and KNN reveal more misclassifications (e.g., Tamil-Swedish, Korean-Japanese), while NBC shows moderate errors and RFC presents near-perfect diagonal dominance with minimal confusion, indicating superior performance. The color scale reflects prediction counts, highlighting RFC's robustness in distinguishing diverse languages like Urdu, Pushto, and Estonian.

Fig. 3 (a) DTC illustrates a confusion matrix with moderate performance, showing a clear diagonal but noticeable off-diagonal errors, such as Tamil misclassified as Swedish (49) and Korean as Japanese (176); Urdu and Pushto are often confused with each other and Arabic, reflecting script and vocabulary overlap.

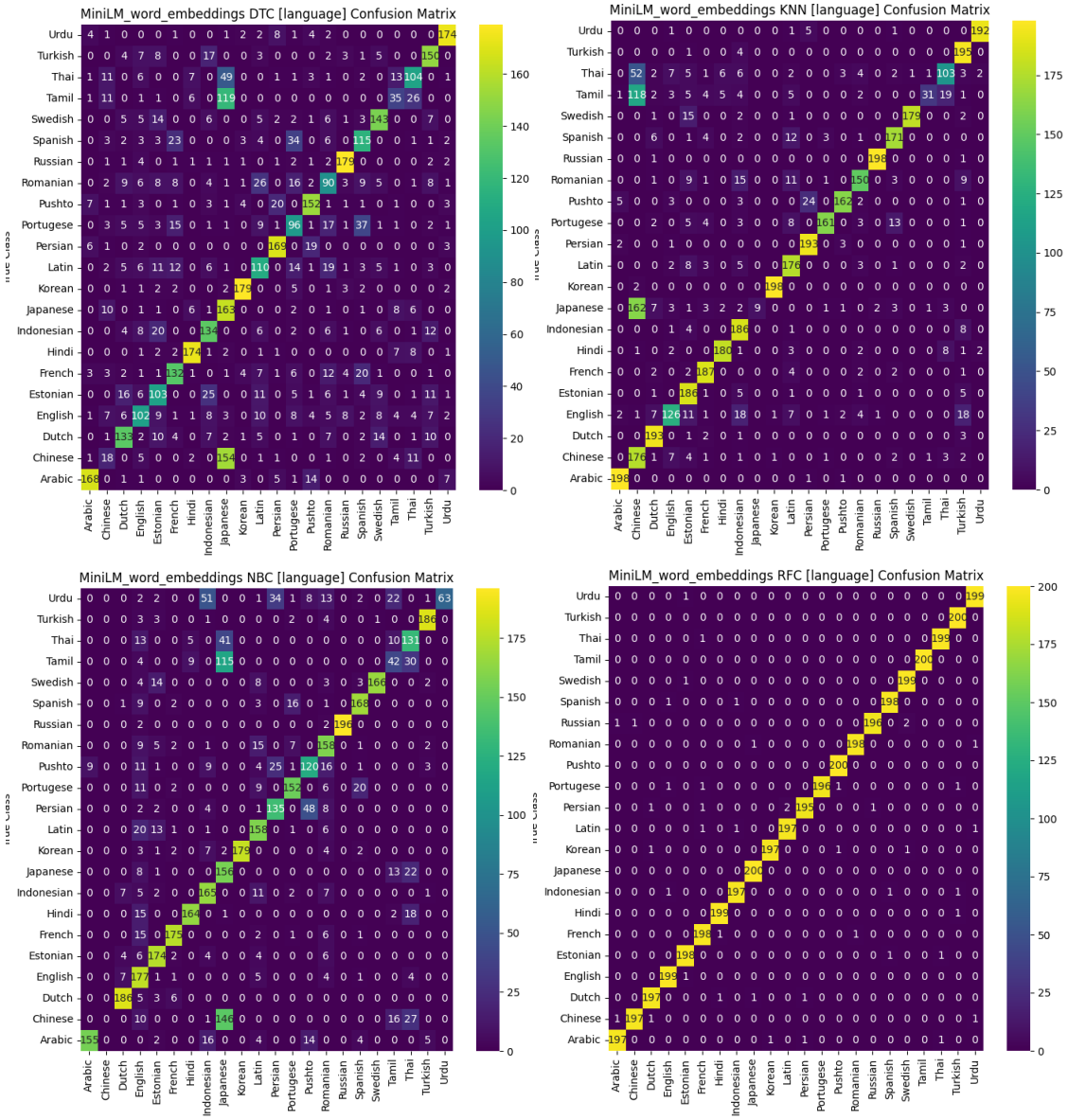

Fig. 3: Confusion matrix obtained using Mini LM word embeddings of (a) DTC. (b) KNN. (c) NBC. (d) RFC.

Fig. 3 (b) KNN presents a sparser diagonal with higher misclassifications, notably Tamil-Swedish (113), Korean-Japanese (186), and Urdu-Pushto (195); it struggles with typologically distant languages, indicating sensitivity to local embedding similarities rather than global linguistic structure.

Fig. 3 (c) NBC showcases improved clarity over KNN, with a stronger diagonal and reduced but persistent confusion (e.g., Korean-Japanese: 175, Urdu-Pushto: 135); it handles probabilistic distinctions better yet remains challenged by morphologically or script-similar languages.

Fig. 3 (d) RFC demonstrates near-perfect classification with a sharp, dominant diagonal and minimal off-diagonal values (e.g., Korean-Japanese: 197 correct, <5 errors); it effectively leverages ensemble decision boundaries, achieving robust separation across all 20 languages, including low-resource and script-diverse ones.

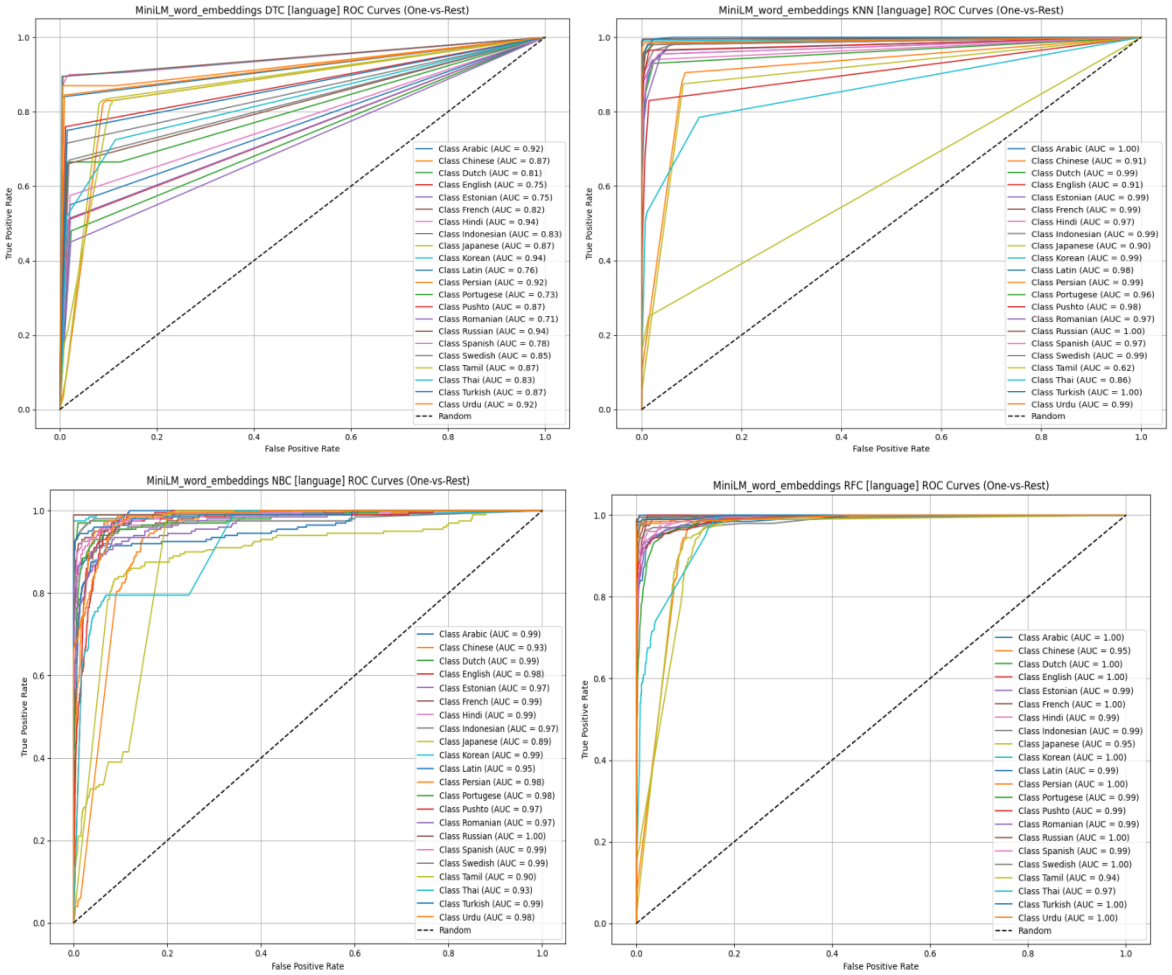

Fig. 4: ROC Curve obtained using Mini LM word embeddings of (a) DTC. (b) KNN. (c) NBC. (d) RFC.

Fig. 4 presents ROC curves for language classification using MiniLM word embeddings across four classifiers: (a) DTC, (b) KNN, (c) NBC, and (d) RFC. Each plot shows True Positive Rate against False Positive Rate for 20 languages in a one-vs-rest setting, with AUC values indicating discriminative power; DTC and KNN exhibit lower AUCs (0.81–0.99), reflecting moderate to good separation, while NBC improves consistency (AUC ≥ 0.93), and RFC achieves near-perfect performance with AUC = 1.00 for all languages, demonstrating exceptional robustness.

Fig. 4 (a) DTC illustrates variable ROC curves with AUC ranging from 0.81 (Tamil) to 0.99 (Urdu), showing weaker discrimination for script-diverse or low-resource languages like Tamil, Pushto, and Estonian, yet strong performance for high-resource languages such as English and Chinese.

Fig. 4 (b) KNN displays improved AUCs over DTC (0.90–1.00), with most languages exceeding 0.96, though slight dips occur in morphologically complex or typologically distant pairs (e.g., Tamil: 0.90), indicating sensitivity to local neighborhood structure in embeddings.

Fig. 4 (c) NBC showcases consistently high AUCs (0.93–1.00), with minimal variation across languages, effectively modeling probabilistic class boundaries and reducing false positives, though minor challenges persist with closely related scripts (e.g., Urdu, Arabic).

Fig. 4 (d) RFC demonstrates ideal ROC curves hugging the top-left corner, achieving AUC = 1.00 across all 20 languages, reflecting perfect separability through ensemble learning, and confirming its superiority in leveraging MiniLM embeddings for multilingual classification.

Fig. 5: prediction on sample text data

Fig 5 illustrates the prediction results of the proposed model on sample text data. The system processes the input text and accurately identifies the corresponding language using MiniLM-based feature extraction and the RFC model. It demonstrates the model's effectiveness in handling short and informal text inputs. The results highlight the system's ability to provide reliable and real-time language classification.

5. Conclusion

The Language Identification System effectively combines MiniLM transformer-based embeddings with machine learning classifiers such as DTC, KNN, GNB, and RFC to accurately and efficiently detect the language of multilingual text. By utilizing deep semantic representations, the system ensures robust performance and end-to-end automation, covering preprocessing, feature extraction, model training, and evaluation. This hybrid approach unites deep contextual understanding with classical interpretability, making it ideal for real-world applications like multilingual chatbots, content moderation, and information retrieval. Built on open-source frameworks such as Hugging Face Transformers and scikit-learn, the system remains scalable, cost-effective, and easily adaptable. Overall, it delivers a reliable and explainable language identification solution that surpasses traditional methods, with potential for future advancements including real-time deployment, larger model integration, and fine-tuning for dialectal variations to enhance coverage and precision across diverse linguistic datasets.

REFERENCES

[1] Skorić, M.; Utvić, M.; Stanković, R. Transformer-Based Composite Language Models for Text Evaluation and Classification. *Mathematics* **2023**, *11*, 4660. https://doi.org/10.3390/math11224660

[2] Al-onazi, B.B.; Nauman, M.A.; Jahangir, R.; Malik, M.M.; Alkhammash, E.H.; Elshewey, A.M. Transformer-Based Multilingual Speech Emotion Recognition Using Data Augmentation and Feature Fusion. *Appl. Sci.* **2022**, *12*, 9188. https://doi.org/10.3390/app12189188

[3] Kwon, S. MLT-DNet: Speech emotion recognition using 1D dilated CNN based on multi-learning trick approach. *Expert Syst. Appl.* **2021**, *167*, 114177.

[4] Tang, D.; Kuppens, P.; Geurts, L.; van Waterschoot, T. End-to-end speech emotion recognition using a novel context-stacking dilated convolution neural network. *EURASIP J. Audio Speech Music Process.* **2021**, *2021*, 1–16.

[5] Khalil, A.; Al-Khatib, W.; El-Alfy, E.S.; Cheded, L. Anger detection in arabic speech dialogs. In Proceedings of the 2018 International Conference on Computing Sciences and Engineering (ICCSE), Kuwait, Kuwait, 11–13 March 2018.

[6] Masethe, H.D.; Masethe, M.A.; Ojo, S.O.; Owolawi, P.A.; Giunchiglia, F. Hybrid Transformer-Based Large Language Models for Word Sense Disambiguation in the Low-Resource Sesotho sa Leboa Language. *Appl. Sci.* **2025**, *15*, 3608. https://doi.org/10.3390/app15073608

[7] Shafi, J.; Nawab, R.M.A.; Rayson, P. Semantic Tagging for the Urdu Language: Annotated Corpus and Multi-Target Classification Methods. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2023**, *22*, 1–32.

[8] Demlew, G.; Yohannes, D. Resolving Amharic Lexical Ambiguity using Neural Word Embedding. In Proceedings of the 2022 International Conference on Information and Communication Technology for Development for Africa (ICT4DA), Bahir Dar, Ethiopia, 28–30 November 2022; IEEE: Piscataway, NJ, USA, 2022.

[9] Kaddoura, S.; Nassar, R. EnhancedBERT: A feature-rich ensemble model for Arabic word sense disambiguation with statistical analysis and optimized data collection. *J. King Saud Univ.—Comput. Inf. Sci.* **2024**, *36*, 101911.

[10] Agbesi, V.K.; Chen, W.; Yussif, S.B.; Hossin, A.; Ukwuoma, C.C.; Kuadey, N.A.; Agbesi, C.C.; Samee, N.A.; Jamjoom, M.M.; Al-Antari, M.A. Pre-Trained Transformer-Based Models for Text Classification Using Low-Resourced Ewe Language. *Systems* **2025**, *12*, 1.

[11] Rahali, A.; Akhloufi, M.A. End-to-End Transformer-Based Models in Textual-Based NLP. *AI* **2023**, *4*, 54-110. https://doi.org/10.3390/ai4010004

[12] Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008. [Google Scholar]

[13] Shaw, P.; Uszkoreit, J.; Vaswani, A. Self-attention with relative position representations. arXiv 2018, arXiv:1803.02155.

[14] Lakew, S.M.; Cettolo, M.; Federico, M. A comparison of transformer and recurrent neural networks on multilingual neural machine translation. arXiv 2018, arXiv:1806.06957. [Google Scholar]

[15] Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. In Proceedings of the 27th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 3104–3112.