From Misrepresentation to Quantities: Labeling Misinformation Types in South Indian Language Summaries

Rachana Nagaraju[*,†], Hosahalli Lakshmaiah Shashirekha[†]

Department of Computer Science, Mangalore University, Mangalore, Karnataka, India

Abstract

The widespread adoption of Large Language Models (LLMs) has introduced new risks associated with the fluent generation of factually incorrect or misleading content. Addressing this challenge requires fine-grained tools capable of not only detecting misinformation but also distinguishing among types of factual errors. Prompt RecOvery for MisInformation Detection (PROMID)-2025 shared task, held as part of the Forum for Information Retrieval Evaluation (FIRE)-2025, directly targets this issue by encouraging systems that analyze and classify incorrectness in LLM-generated outputs. We - team MUCS participated in Subtask 2, which requires classifying LLM-generated summaries into one of four predefined misinformation categories: *misrepresentation*, *fabrication*, *false attribution*, and *incorrect quantities*. The subtask is challenging due to the semantic similarity between classes and the multilingual setting that includes Kannada, Malayalam, Tamil, and Telugu. We propose three multilingual Deep Learning (DL) pipelines: (i) Bidirectional Long Short Term Memory (BiLSTM) model, (ii) Transformer + BiLSTM hybrid model, and (iii) Bidirectional Gated Recurrent Unit (BiGRU) model, to categorize each data point into one of the four different categories based on the presence of factual incorrectness in the summaries. Each model employs language-aware pre-processing, subword-aware tokenization, and contextual encoders tailored to sequence modeling. The Transformer + BiLSTM model integrates transformer encoders, BiLSTM layers, and multi-head self-attention to capture both global and local dependencies. In contrast, BiLSTM and BiGRU models use simpler recurrent architectures combined with attention mechanisms to reduce computational overhead. To address mild class imbalance, we apply *focal loss* during training along with mixed-precision optimization for efficiency. The proposed models obtained best performances with: BiLSTM model ranking 1st in both Tamil and Telugu, BiGRU model ranking 2nd in Kannada, and the Transformer + BiLSTM model ranking top-3 position in Malayalam. These results demonstrate the utility of hybrid neural modeling and linguistically-informed pre-processing for multilingual misinformation classification in LLM-generated content.

Keywords

Misinformation detection, Large Language Models, Transformer-BiLSTM, Multilingual NLP, South Indian languages

1. Introduction

The emergence of powerful LLMs such as Generative Pre-trained Transformer (GPT)-4, Pathways Language Model (PaLM), and Large Language Model Meta AI (LLaMA) has significantly transformed the landscape of natural language generation. These models exhibit unprecedented fluency, contextual awareness, and language understanding across a wide range of tasks. However, their susceptibility to generate factually incorrect outputs—commonly referred to as *hallucinations*, has become a matter of growing concern [1, 2]. In many use cases, including news summarization, health advisories, or educational content generation, such hallucinated information can mislead users, propagate misinformation, and erode trust in Artificial Intelligence (AI) systems.

Combating misinformation in LLM-generated content is not only a matter of detecting falsehoods but also of understanding their origin. Many instances of misinformation can be traced back to ambiguous or misleading prompts that guide the model to generate specific types of incorrect summaries. Traditional

Forum for Information Retrieval Evaluation, December 17-20, 2025, India

[*]Corresponding author.

[†]These authors contributed equally.

rachananagaraju20@gmail.com (R. Nagaraju); hlsrekha@mangaloreuniversity.ac.in (H. L. Shashirekha)

0000-0002-9421-8566 (H. L. Shashirekha)

misinformation detection systems often neglect this generative aspect, focusing on surface-level textual features or post-analysis of static web content [3]. To address this gap, PROMID [4, 5] - a shared task at FIRE 2025, aims to foster research on detecting and analyzing misinformation through the lens of both its content and its generative context. PROMID-2025 [4] shared task contains three subtasks: Subtask 1: Prompt Recovery from LLM generated misinformative text, Subtask 2: Misinformation Detection in LLM generated text, and Subtask 3: Misinformation Detection in social media texts, and we participated in only Subtask 2. Given a piece of LLM-generated text with misinformation, the objective of Subtask 2 is to categorize each data point into one of the four different categories based on the presence of factual incorrectness in the summaries. The four fine-grained classes of factual incorrectness are: *misrepresentation*, *fabrication*, *false attribution*, and *incorrect quantities*, and the Subtask 2 [6] is offered in four South Indian languages - Kannada, Malayalam, Tamil, and Telugu. This demands a nuanced understanding of not only the summary but also its deviation from grounded source material. The task becomes even more complex in a multilingual setting, particularly across the given low-resource, morphologically rich South Indian languages.

We, team MUCS, address the challenges of Subtask 2 by designing a language-agnostic misinformation classification framework capable of capturing both contextual semantics and syntactic nuances that differentiate hallucinated content types. We develop three end-to-end multilingual DL pipelines: (i) BiLSTM model, (ii) Transformer + BiLSTM hybrid model, and (iii) BiGRU model, to categorize each data point into one of the four different categories based on the presence of factual incorrectness in the summaries. Each pipeline incorporates robust text pre-processing, custom subword-aware tokenization, and neural encoders designed to classify misinformation into one of four predefined categories across four South Indian languages - Kannada, Malayalam, Tamil, and Telugu. The architectures vary in complexity: the Transformer+ BiLSTM model includes transformer encoders, multi-head self-attention, and BiLSTM layers, while the BiLSTM and BiGRU models use purely recurrent layers along with attention mechanisms tailored to their depth. These configurations allow the models to learn both global dependencies and local sequential patterns—critical for distinguishing subtle misinformation cues. To address the mild class imbalance present in the dataset, we apply the *focal loss function* with class-aware weighting. Additionally, mixed-precision training is used for computational efficiency in deeper configurations.

Performances of the proposed models varied across languages depending on the architecture. The BiLSTM model attained Rank 1 in both Tamil and Telugu, the BiGRU model achieved Rank 2 in Kannada, and the Transformer + BiLSTM model secured a consistent top-3 position in Malayalam. These results underscore the benefits of combining linguistic preconditioning, cross-lingual representation learning, and architecture-specific modeling choices for classifying misinformation. Overall, this work contributes to the broader goal of developing robust and prompt-resilient misinformation detection systems for LLM-generated content [7].

The subsequent sections of this paper details the related works (Section 2), methodology (Section 3), experiments, results, and implications of our approach (Section 4), followed by conclusion and future works (Section 5).

2. Related Works

Recent advancements in natural language generation have intensified efforts to detect hallucinated or misleading content automatically. Misinformation classification, particularly in LLM-generated text, intersects with factuality evaluation, hallucination detection, prompt sensitivity, and deep contextual modeling. Researchers have proposed a broad spectrum of models, evaluation metrics, and benchmarks to address these challenges, ranging from Question Answer (QA) based fidelity evaluation to fine-grained hallucination categorization.

Ji et al. [1] presents a foundational taxonomy of hallucinations, categorized into intrinsic (information not grounded in the source) and extrinsic (conflicting with source content). The authors systematically evaluated summarization models, notably BART and PEGASUS, on CNN/DailyMail, XSum, and WebNLG

datasets. They observed that ROUGE - a commonly used metric is incapable of penalizing factual mismatches, thereby overestimating performance. The authors also reviewed recent evaluation tools - FactCC, DAE, and QuestEval, finding their F1-based factuality improvements modest compared to human evaluation. While the survey is comprehensive, it remains focused on English data and does not cater to multilingual or low-resource hallucination scenarios. Zhou et al. [8] focus on hallucination phenomena in large-scale instruction-tuned models like GPT-2, GPT-3, and T5. The authors identified various failure cascades, especially under weak prompt specification and retrieval-agnostic generation. They test strategies such as Retrieval-Augmented Generation (RAG) and fact-aware k-nearest neighbor decoding on datasets like WebGPT and TriviaQA, achieving 10–12% reductions in factual errors. While the interventions offer measurable improvements, the solutions are tightly coupled with access to document-level retrieval systems, posing scalability issues in disconnected or unknown knowledge domains.

Manakul and Gales [9] introduced SelfCheckGPT, a zero-reference evaluation method that detects hallucinations by measuring variation between multiple LLM responses under the same input. Evaluated across XSum, Wikibio, and CommonGen, SelfCheckGPT outperforms supervision-based tools like DAE and QAEval, achieving an average F1 score of 0.72. Because SelfCheckGPT treats the LLM as a black box, it is suitable for commercial models (e.g., ChatGPT) where internal architectures are inaccessible. Nevertheless, the model's reliability depending on generative diversity—hallucinations may go undetected if the model consistently repeats incorrect outputs. Rashkin et al. [10] propose TruthfulQA, a benchmark system built to expose models to adversarial factual errors through misleading or underspecified prompts. They show that even few-shot GPT-3 achieves only 58% accuracy when prompted with deceptive or ambiguous inputs. Their analysis reveals the models' alignment with misinformation commonly found on the web due to pretraining. TruthfulQA is useful in diagnosing hallucination types related to real-world disinformation, but its design is task-specific—limited to QA rather than open-ended summarization or generation contexts.

Gupta and Vishwakarma [7] address fine-grained misinformation detection in the context of COVID-19 tweets. They apply BERT and RoBERTa classifiers to a custom-annotated dataset with misinformation categories that include denial, satire, and conspiracy. Achieving macro F1 scores of up to 0.76, their approach significantly outperformed traditional linear and tree-based models like Support Vector Machine (SVM) and Decision Trees. However, class imbalance and semantic overlap between misinformation types reduce per-class precision and recall, particularly for satire vs. sarcasm. Furthermore, the dataset is monolingual and domain-specific, restricting generalizability. Wright and Pavlick [11] present Factool - a factuality evaluation tool built for summarization. It integrates dependency-based heuristics and semantic entailment checks into a unified scoring mechanism. On XSum and CNN/DailyMail (evaluated against human annotations), Factool improves alignment scores by roughly 15% over BART and T5 outputs. Its integration of linguistic parsing provides control over soft factual inconsistencies. However, the tool requires high-quality syntactic parsers and cannot be easily deployed in non-English or noisy language settings, limiting its scalability.

Alam et al. [12] investigated misinformation detection in multilingual settings using multilingual BERT and XLM-R. Applied to COVID-19 claim datasets in Arabic, Hindi, Tamil, and Bengali, the models demonstrate that fine-tuned multilingual encoders outperform machine-translated pipelines by a margin of 12–15% F1. While the work confirms that transfer learning can be beneficial across low-resource setups, it is constrained to binary classification and does not integrate type-level misinformation tags like hallucinated quantities or misattribution. Krishna et al. [13] analyzed factual consistency by comparing model output (T5, PEGASUS) on summarization datasets (SAMSum, XSum) using both lexical and semantic evaluation frameworks. They illustrated that hallucinations, especially false attribution and fabrication, often go unpunished by standard metrics (BLEU, ROUGE), with up to 40% of incorrect content rated acceptable due to lexical overlap. They introduce a claim-type tagging scheme combined with human judgment alignment but stop short of offering a machine-learnable model for hallucination classification. Min et al. [14] propose FactScore - a sentence-level hallucination evaluator combining dense retrieval and entailment verification. Initially applied to SciFact and PubMedQA, it achieves 85% factual agreement without requiring reference summaries. FactScore is particularly advantageous in

domain-specific settings such as biomedical summarization. Yet, its performance in multilingual or informal narratives remains unexplored, limiting its utility in non-academic LLM-generated content domains.

Razeghi et al. [15] explored the impact of prompt template variations on GPT-3's ability to produce accurate answers and observed a variability of up to 20% in task accuracy (SuperGLUE and ARC) under slight changes to input phrasing. Their study demonstrates that prompts themselves introduce biases and instability, often leading to contradictory answers. While insightful, the study focuses primarily on zero-shot settings and does not extend the prompt variation analysis to hallucination detection. Augenstein et al. [16] present a fine-grained taxonomy of hallucination classes—such as numeric misrepresentation, source fabrication, and attribute drift—and apply the framework to GPT-3 outputs across QA, summarization, and text generation. Human annotations reveal that 60% of analyzed outputs contain overlapping hallucination types. The study identifies multi-layered error patterns but does not provide an automated model to perform this categorization, leaving the system useful largely as an annotation benchmark.

Collectively, these studies offer strong foundational insights into hallucination detection and misinformation classification. Yet, challenges remain with respect to multilingual support, hallucination attribution granularity, prompt variability, and label-level explainability. Although SelfCheckGPT and FactScore contribute to progress in zero-reference evaluation, and taxonomy efforts enhance annotation clarity [16], existing systems still struggle with cross-lingual generative hallucinations that lack structured evidence or precise prompts. Bridging these gaps is critical for the scalable and reliable deployment of LLMs.

3. Methodology

This section describes the three proposed multilingual DL pipelines. Each pipeline processes summaries and source articles in South Indian languages - Kannada, Malayalam, Tamil, and Telugu, and assigns one of the four misinformation class labels to the given input. The models differ in tokenization, sequence encoders, attention mechanisms, and optimization strategies. Each pipeline follows a general structure of: (i) pre-processing the input, (ii) tokenization, (iii) sequence encoding with neural layers, and (iv) classification. The proposed end-to-end multilingual DL pipelines for Subtask 2 is shown in Figure 1 and a description of the models is given in the following sub-sections.

3.1. BiLSTM Model

The BiLSTM model serves as a strong baseline in our system. It combines a deep recurrent encoder with multi-head self-attention to model sequential dependencies and incorporate global context for effective hallucination detection. The details of the model are given below:

- **Text Pre-processing:**
  - URLs and email address are removed using regex[1].
  - Punctuation characters (e.g., `!"#$...~`) are stripped.
  - Whitespace is normalized by collapsing multiple spaces and trimming.
  - No long-token removal is used in this basic version.
- **Tokenization:**
  - We employ a `BasicTokenizer` that splits input text on whitespace characters.
  - Language indicator tokens (`<ta>`, `<kn>`) are retained.
  - Tokens longer than 10 characters are split into fixed-length, non-overlapping 5-character chunks[2].

[1] https://en.wikipedia.org/wiki/Uniform_Resource_Locator

[2] Inspired by Byte-Pair Encoding (BPE) principles: https://huggingface.co/blog/how-to-preprocess

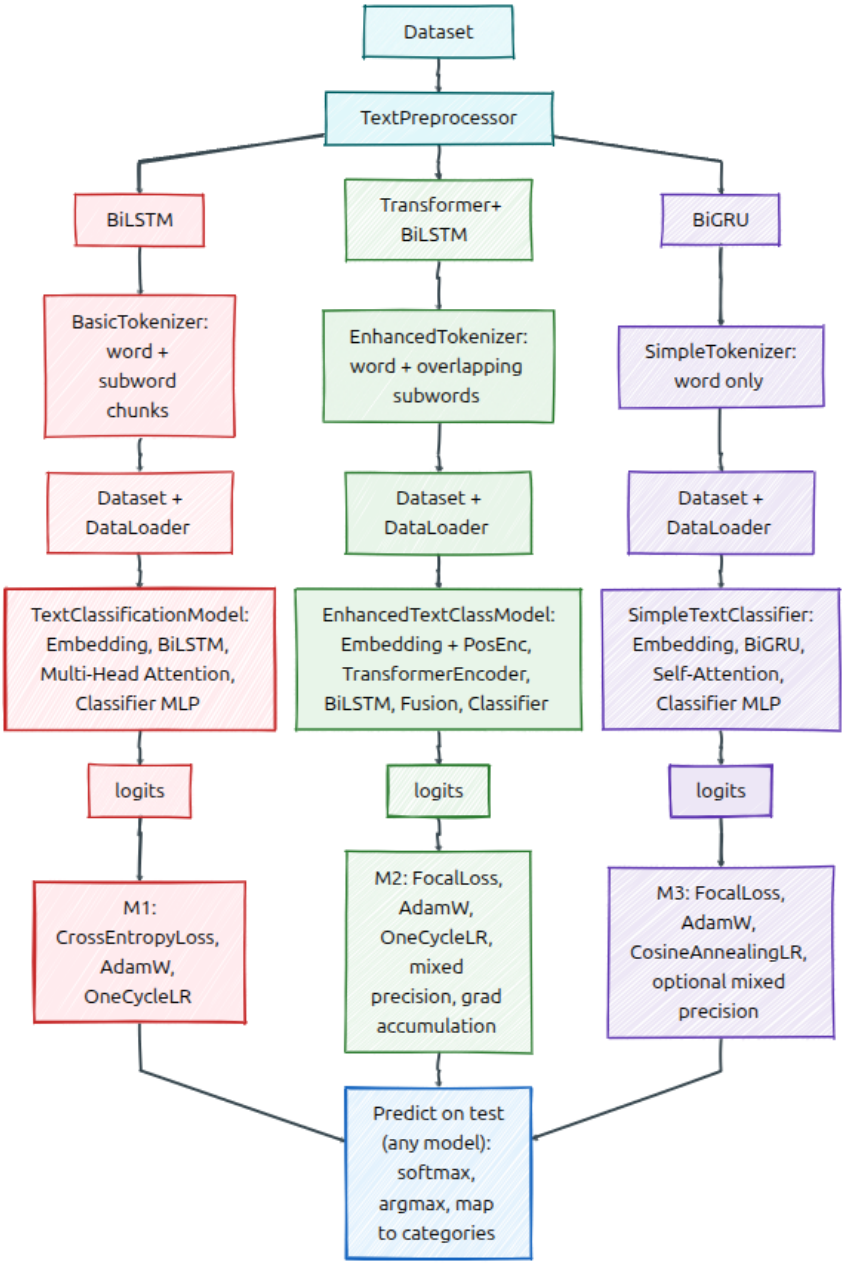


Figure 1: Proposed Multilingual Deep Learning Pipelines for Subtask 2

  - Vocabulary is created using tokens with a frequency ≥ 2 in the training set.

- **Model and Training Loop:**
  - The architecture is based on BiLSTM[3] - a recurrent model that captures both past and future context by processing the sequence in forward and backward directions.
  - Each token is embedded into a 256-dimensional vector.
  - A linear layer with dropout simulates positional information.
  - A 2-layer BiLSTM with 512 hidden units per direction outputs 1024 dimensional contextual embeddings.
  - A Multi-Head Self-Attention layer[4] with 8 heads learns token importance for downstream aggregation.

[3] https://pytorch.org/docs/stable/generated/torch.nn.LSTM.html
[4] https://pytorch.org/docs/stable/generated/torch.nn.MultiheadAttention.html

  - Masked average pooling skips padding and compress the sequence into a fixed-length vector.
  - A Multi-Layer Perceptron (MLP) classifier with dimensions `1024 → 512 → 256 → 4` computes the class prediction. MLPs are fully connected layers that use GELU activations [17], LayerNorm, and dropout for regularization.
- **Loss and Optimization:**
  - We apply Weighted Cross-Entropy Loss where class weights are computed as the inverse frequency from the training labels.
  - Optimization uses the AdamW optimizer [18] — an adaptive gradient method with decoupled weight decay.
  - Learning rate is controlled via the OneCycleLR scheduler [19], which warms up and gradually cools down the learning rate for better convergence.
  - Gradient clipping is applied to prevent exploding gradients.
  - Training is conducted on mini-batches (size 16) for 10 epochs.

This model provides a balance between performance and computational simplicity, offering strong results across two languages while using a purely recurrent encoder architecture. Its relatively lightweight design makes it particularly effective for resource-constrained settings.

3.2. Transformer + BiLSTM Model

This hybrid model combines a Transformer Encoder with a BiLSTM network, enabling a fusion of global attention-based contextual modeling and sequential dependency learning. It is the most architecturally complex of the three approaches. The details of the model are given below:

- **Text Pre-processing:**
  - Same as in BiLSTM model, but includes a filter to discard tokens with length ≥ 50[5].
- **Tokenization:**
  - The `EnhancedTokenizer` splits on whitespace and segments tokens >10 characters using overlapping 5-character windows with stride 3.
  - This overlapping subword approach captures richer morphological structures for agglutinative languages.
  - Vocabulary is constructed with `min_freq = 2`.
- **Model and Training Loop:**
  - Tokens are mapped to 384 dimensional embeddings plus learned positional embeddings.
  - A 3-layer Transformer Encoder applies multi-head self-attention and feed-forward layers to model global context.
  - The two representations (Transformer + BiLSTM) are concatenated and projected down to 768-d using a linear layer.
  - Outputs are passed to a 2-layer BiLSTM (384 hidden units per direction).
  - After dropout and LayerNorm, masked average pooling yields a fixed vector.
  - An MLP classifier (`768 → 384 → 4`) predicts the class.
- **Loss and Optimization:**
  - We use FocalLoss [20] with focusing parameter $\gamma = 2.0$ and class weighting α.
  - It improves learning by penalizing easy examples and focusing on hard-to-classify ones.
  - Optimizer: AdamW with weight decay.
  - Scheduler: OneCycleLR with warm-up phase.

[5] Extreme-length tokens often indicate noise from LLM-generated or malformed inputs

  - Mixed-precision training is enabled via Automatic Mixed Precision (AMP)[6].
  - Gradient accumulation is used to emulate larger batch sizes.

This architecture benefits from the transformer's ability to capture long-range dependencies and the BiLSTM's ability to maintain sequential coherence. It performed robustly on morphologically rich scripts, demonstrating its suitability for complex linguistic structures.

3.3. BiGRU Model

The BiGRU model is a lightweight architecture that discards transformer components in favor of a purely recurrent encoder with attention. It serves as a strong baseline in low-resource scenarios. The details of the model are given below:

- **Text Pre-processing:**
  - Same as in Transformer + BiLSTM model and URL + punctuation removal, space normalization, and dropping tokens $>$50 characters.
- **Tokenization:**
  - `SimpleTokenizer` applies minimal whitespace-based splitting.
  - No subword segmentation is used (baseline).
- **Model and Training Loop:**
  - 256 dimension tokens processed by a 2-layer Bidirectional GRU (BiGRU)[7] with 512 units per direction.
  - Additive attention computes token-level importance using: `Linear(1024 → 512) → Tanh → Linear(512 → 1)` followed by Softmax.
  - A weighted sum over token vectors produces a document representation.
  - MLP classifier: `1024 → 512 → 256 → 4`.
- **Loss and Optimization:**
  - FocalLoss with $\gamma = 2.0$ and class-weighted α.
  - AdamW optimizer.
  - CosineAnnealingLR scheduler[8] reduces learning rate smoothly.
  - Model trained with batch size 32; best checkpoint chosen based on lowest loss.

Due to its simplicity, fast convergence, and minimal memory overhead, the BiGRU model is ideal for deployment in real-time applications or where computational resources are limited.

The configuration of *hyperparameters* used in training these three models are provided in Table 1.

4. Experiments and Results

This section presents the empirical evaluation of our proposed system on Subtask 2 of the PROMID-2025 shared task. We report both quantitative performance scores and qualitative insights. The evaluation is designed to assess the system's generalization across diverse linguistic structures in four Indian languages: Kannada, Malayalam, Tamil, and Telugu. Each model is evaluated using Precision, Recall, Macro F1-score, and overall classification Accuracy. A description of the dataset, followed by a detailed analysis of model performance for each language, is provided below:

[6] https://pytorch.org/docs/stable/amp.html
[7] https://pytorch.org/docs/stable/generated/torch.nn.GRU.html
[8] https://pytorch.org/docs/stable/generated/torch.optim.lr_scheduler.CosineAnnealingLR.html

Table 1
Configuration of Hyperparameters to Train the Proposed Models

Parameter	BiLSTM Model	Transformer + BiLSTM Model	BiGRU Model
Embedding Dim	256	384	256
Sequence Encoder	BiLSTM	Transformer + BiLSTM	BiGRU
Hidden Size	512	384	512
RNN Layers	2	2	2
Attention Type	Multi-head	Multi-head + Fusion	Additive
Feature Fusion	–	Concatenation + Linear	–
Classifier MLP	1024→512→256→4	768→384→4	1024→512→256→4
Loss Function	CrossEntropy	FocalLoss ($\gamma = 2$)	FocalLoss ($\gamma = 2$)
Optimizer	AdamW	AdamW	AdamW
Learning Rate	2×10^{-4}	2×10^{-4}	2×10^{-4}
Weight Decay	0.01	0.01	0.01
Scheduler	OneCycleLR	OneCycleLR	CosineAnnealingLR
Batch Size	16	16	32
Mixed Precision	No	Yes	Yes
Grad Accumulation	No	Yes	No
Epochs	10	10	10

Table 2
Description of the PROMID-2025 Subtask 2 Dataset

Field	Description
`Title`	A short title or headline summarizing the article topic
`Headlines`	Additional or extended versions of the title, if available
`Article`	The full ground-truth article body used as the reference source
`Incorrect_Summary`	A hallucinated or misleading summary, typically generated by an LLM
`Incorrectness_Type`	One of four misinformation categories: *misrepresentation*, *fabrication*, *false attribution*, or *incorrect quantities*
`Correct_Summary`	A human-written, factually accurate summary of the article

Table 3
Language-wise Data Distribution

Language	Train	Test	Class Distribution (Train)
Kannada	4,975	200	294 / 250 / 250 / 195
Malayalam	4,975	200	294 / 250 / 250 / 195
Tamil	4,975	200	294 / 250 / 250 / 195
Telugu	4,975	200	294 / 250 / 250 / 195
Total	19,900	800	–

Classes: Misrepresentation, Fabrication, False Attribution, Incorrect Quantities

4.1. Data Description

The multilingual misinformation classification dataset [21, 22] released as part of the PROMID-2025 [23] shared task contains hallucinated summaries generated by LLMs paired with gold-standard articles in four South Indian languages: Kannada, Malayalam, Tamil, and Telugu. Each summary is labeled with one of four incorrectness types: *misrepresentation*, *fabrication*, *false attribution*, and *incorrect quantities*. The overall category-level distribution across all training data is as follows:

- Misrepresentation: 29.7%
- Fabrication: 25.3%
- False Attribution: 25.3%
- Incorrect Quantities: 19.7%

Table 2 gives the description of the datasets and Table 3 shows the class-wise distribution.

4.2. Results and Analysis

The models are evaluated using four standard metrics: macro-averaged Precision, Recall, F1-score, and overall Accuracy, and ranked based on macro-averaged F1-score. We tested the proposed multilingual models: BiLSTM model, Transformer + BiLSTM hybrid model, and BiGRU-based model, on each language. Table 4 presents the performance metrics for each model across the four languages and Figure 2 offers a visual comparison of team-wise macro F1-scores of all the participating teams in Subtask 2 across the four languages.

Table 4
Performance of Proposed Multilingual Models across all Languages

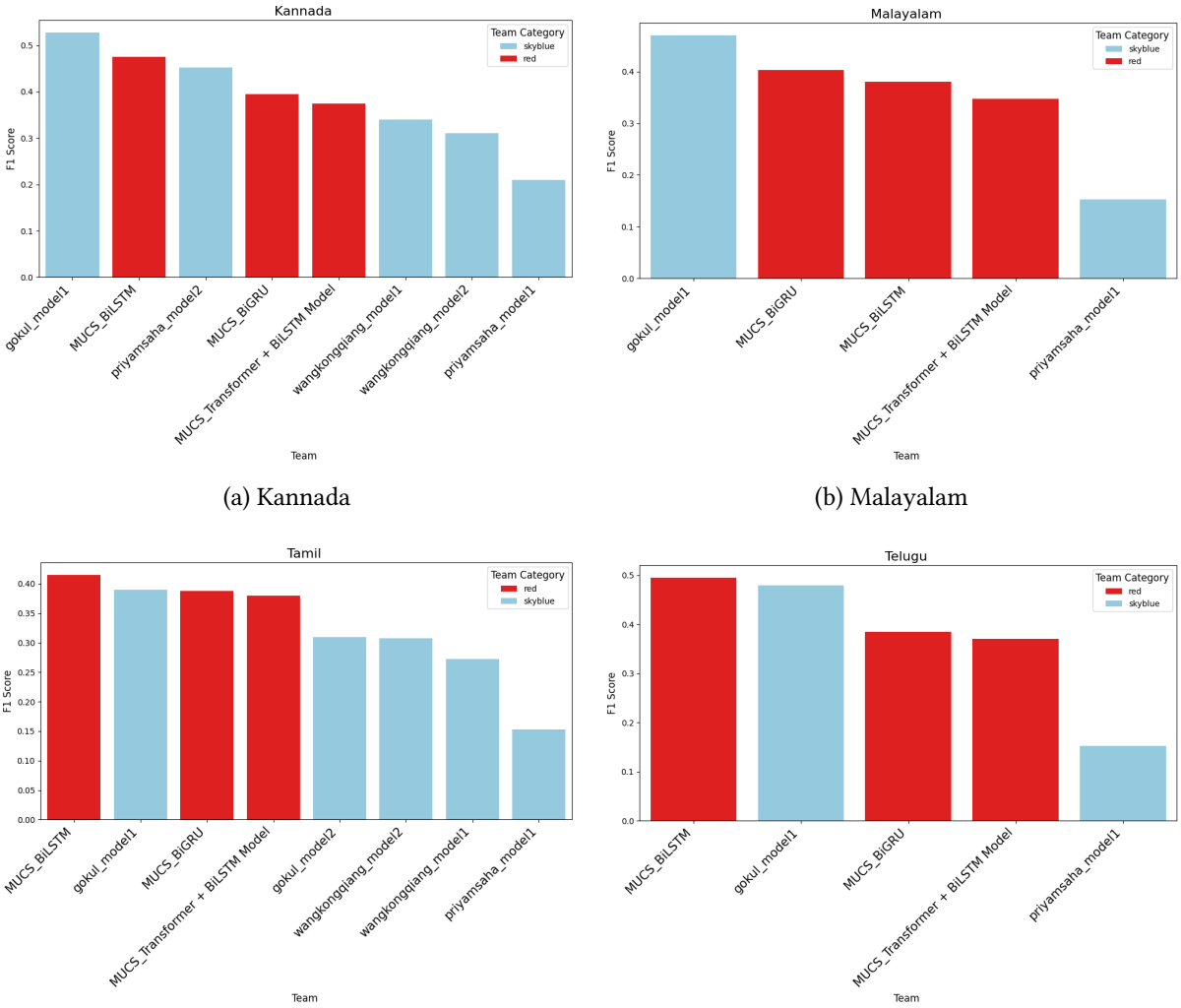
Language	Model	Precision	Recall	F1 Score	Accuracy
Kannada	BiLSTM	**0.5850**	**0.4150**	**0.4750**	**0.7150**
	Transformer + BiLSTM	0.4700	0.3650	0.3750	0.6975
	BiGRU	0.4800	0.3825	0.3950	**0.7200**
Malayalam	BiLSTM	0.4775	0.3675	0.3800	0.6525
	Transformer + BiLSTM	0.4950	0.3825	0.3475	0.6450
	BiGRU	**0.5250**	**0.4200**	**0.4025**	**0.7125**
Tamil	BiLSTM	**0.5300**	0.3650	**0.4150**	0.6950
	Transformer + BiLSTM	0.4300	0.3600	0.3800	0.6950
	BiGRU	0.4625	**0.3725**	0.3875	**0.7050**
Telugu	BiLSTM	**0.5950**	**0.4375**	**0.4950**	**0.7300**
	Transformer + BiLSTM	0.4375	0.3525	0.3700	0.6900
	BiGRU	0.4950	0.3475	0.3850	0.6975

Our analysis reveals that the BiLSTM model shows illustrated performance for Tamil and Telugu, achieving the highest F1-scores and accuracies. This suggests that simpler architectures with adequate attention mechanisms, when paired with class-weighted loss, can effectively capture essential patterns in morphologically rich languages. For Kannada, the BiLSTM model attains the best F1-score, whereas the BiGRU model yields the highest accuracy. This variation suggests that while BiGRU may be better at general predictions, BiLSTM model maintains precision-recall balance better. For Malayalam, the BiGRU model outperforms both the BiLSTM and Transformer + BiLSTM variants across all metrics demonstrating that lightweight architectures can still perform competitively for certain languages.

Although the dataset is relatively balanced overall, some class imbalance particularly under the *incorrect quantities* category introduces challenges. Combined with semantic overlap between categories such as *false attribution* and *fabrication*, this affects low-recall cases. Performance differences also underline the influence of architecture depth, loss strategy, and attention mechanisms. The use of focal loss with class-aware weighting and sequence-level attention improves outcome reliability, especially in low-resource, multilingual settings.

5. Conclusion and Future Work

Our multilingual DL system for Subtask 2 of the PROMID-2025 shared task demonstrates strong performance in classifying LLM-generated summaries into fine-grained misinformation categories across four South Indian languages—Kannada, Malayalam, Tamil, and Telugu. Leveraging hybrid transformer-recurrent architectures, attention-based encoding, and class-aware focal loss, the Transformer + BiLSTM model effectively captures both sequential and contextual signals required to detect subtle factual inconsistencies. While the attention mechanism and focal loss are employed across all three models, the hybrid architecture most directly benefits from combined transformer and recurrent components. BiLSTM model achieved Rank 1 in Telugu and Tamil, BiGRU model achieved Rank 2 in Kannada, and Transformer + BiLSTM model obtained Rank 3 in Malayalam, based on macro F1-scores submitted to the official leader board. These results reflect not only the adaptability of our architecture across typologically distinct languages but also the effectiveness of pre-processing strategies, subword-aware



(a) Kannada

(b) Malayalam

(c) Tamil

(d) Telugu

Figure 2: Comparison of Macro F1-scores of all the Participating Teams in Subtask 2

tokenization, and stable training under a mildly imbalanced class distribution. To further improve the system, we plan to explore methods for enhancing model interpretability, particularly in generating explanations for the predicted misinformation types. We also intend to evaluate lightweight model variants for more efficient deployment in practical applications. These directions are aimed at making LLM-based misinformation detection more scalable, interpretable, and robust in multilingual scenarios.

Declaration on Generative AI

In the course of preparing this paper, we made limited use of a generative AI assistant to support the writing process. The tool was used primarily for language refinement, section structuring, and LaTeX formatting consistency. All technical content, including experimental design, model implementation, and results, was conceived, executed, and validated entirely by the authors. The AI assistant did not contribute novel research ideas, nor did it influence the reported findings. Its role was strictly supportive comparable to using grammar checkers or typesetting tools and all content included in this manuscript has been carefully reviewed and approved by the authors.

References

[1] Z. Ji, N. Lee, J. Fries, T. Yu, Survey of Hallucination in Natural Language Generation, ACM Computing Surveys 55 (2023) 1–38.

[2] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, T. Liu, A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions, ACM Transactions on Information Systems 43 (2025) 1–55. URL: https://dl.acm.org/doi/10.1145/3703155. doi:10.1145/3703155.

[3] X. Zhou, R. Zafarani, Fake News Detection: A Survey, ACM Computing Surveys (CSUR) 53 (2020) 1–40.

[4] G. K. Shahi, A. Hegde, S. Satapara, P. Mehta, S. Modha, D. Ganguly, D. Nandini, H. L. Shasirekha, A. K. Jaiswal, G. Pasi, T. Mandl, Overview of the First Shared Task on Prompt Recovery for Misinformation Detection (PROMID 2025), in: K. Ghosh, T. Mandl, S. Pal, S. Majumdar, A. Chakraborty (Eds.), Working Notes of FIRE 2025 - Forum for Information Retrieval Evaluation, Varanasi, India. December 17–20, 2025, CEUR Workshop Proceedings, CEUR-WS.org, 2025.

[5] A. Hegde, G. K. Shahi, S. Satapara, P. Mehta, S. Modha, D. Ganguly, D. Nandini, H. L. Shasirekha, A. K. Jaiswal, G. Pasi, T. Mandl, Prompt recovery for misinformation detection at fire 2025, in: Proceedings of the 17th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE '25, Association for Computing Machinery, 2025.

[6] S. Satapara, P. Mehta, S. Modha, D. Ganguly, Key Takeaways from the Second Shared Task on Indian Language Summarization (ILSUM 2023), in: K. Ghosh, T. Mandl, P. Majumder, M. Mitra (Eds.), Working Notes of FIRE 2023 - Forum for Information Retrieval Evaluation (FIRE-WN 2023), Goa, India, December 15–18, 2023, volume 3681 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 724–733. URL: https://ceur-ws.org/Vol-3681/T8-1.pdf.

[7] B. Gupta, D. K. Vishwakarma, Detecting Fine-grained Misinformation Categories in Covid-19 Tweets, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 35, 2021, pp. 13557–13564.

[8] Y. Zhou, R. Xu, Y. Guo, X. Qiu, A Survey of Hallucination in Large Language Models, arXiv preprint arXiv:2303.02123 (2023).

[9] W. Manakul, M. Gales, SelfCheckgpt: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models, in: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL), 2023, pp. 8434–8451.

[10] H. Rashkin, E. M. Smith, M. Li, N. Stiennon, S. R. Bowman, TruthfulQA: Measuring How Models Mimic Human Falsehoods, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL), 2021, pp. 3214–3229.

[11] K. Wright, E. Pavlick, Factool: Factuality Evaluation for Abstractive Summarization, in: Findings of the Association for Computational Linguistics: ACL 2022, 2022, pp. 2217–2231.

[12] F. Alam, T. Alhindi, F. Dalvi, U. Umer, H. Mubarak, A. Abdelali, S. Shaar, A. Nikolov, K. Darwish, P. Nakov, Fighting the Covid-19 Infodemic: Modeling the Performance of Multilingual Misinformation Detectors, in: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2021, pp. 6896–6911.

[13] K. Krishna, L. Ma, G. Durrett, Faithful or Not? Revisiting Factual Consistency Evaluation in Abstractive Summarization, in: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2023.

[14] S. Min, P. Lewis, F. Petroni, W.-t. Yih, H. Hajishirzi, FactScore: Fine-Grained Factuality Scoring for Content Hallucination, in: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL), 2023.

[15] Y. Razeghi, R. Logan IV, M. Gardner, S. Singh, Impact of Prompt Formatting and Perturbation on Gpt-3's Zero-Shot Performance, in: Findings of the Association for Computational Linguistics: EMNLP 2022, 2022, pp. 4649–4662.

[16] I. Augenstein, Y. Cao, S. Wiseman, Taxonomy of Hallucinations in Natural Language Processing, Transactions of the Association for Computational Linguistics 11 (2023) 1–23.

[17] D. Hendrycks, K. Gimpel, Gaussian Error Linear Units (GELUs), arXiv preprint arXiv:1606.08415 (2016). URL: https://arxiv.org/abs/1606.08415.

[18] I. Loshchilov, F. Hutter, Decoupled Weight Decay Regularization, 2019. URL: https://arxiv.org/abs/1711.05101. arXiv:1711.05101.

[19] L. N. Smith, A Disciplined Approach to Neural Network Hyper-parameters: Part 1 – Learning Rate, Batch Size, Momentum, and Weight Decay, 2018. URL: https://arxiv.org/abs/1803.09820. arXiv:1803.09820.

[20] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal Loss for Dense Object Detection, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2999–3007. doi:10.1109/ICCV.2017.324.

[21] G. K. Shahi, Y. Mejova, Too Little, Too Late: Moderation of Misinformation around the Russo-Ukrainian Conflict, in: Proceedings of the 17th ACM Web Science Conference (WebSci '25), Association for Computing Machinery, 2025. doi:10.1145/3717867.3717876.

[22] G. K. Shahi, T. A. Majchrzak, AMUSED: An Annotation Framework of Multimodal Social Media Data, in: F. Sanfilippo, O.-C. Granmo, S. Y. Yayilgan, I. S. Bajwa (Eds.), Intelligent Technologies and Applications, Springer International Publishing, Cham, 2022, pp. 287–299.

[23] S. Satapara, P. Mehta, D. Ganguly, S. Modha, Fighting Fire with Fire: Adversarial Prompting to Generate a Misinformation Detection Dataset, CoRR abs/2401.04481 (2024). URL: https://doi.org/10.48550/arXiv.2401.04481. doi:10.48550/ARXIV.2401.04481. arXiv:2401.04481.