Comparative Analysis of Machine Learning Methods for Automated Fake News Detection

Matthew Muscedere
School of Computer Science
University of Windsor
Windsor, Canada
muscede2@uwindsor.ca

Tansh Koul
School of Computer Science
University of Windsor
Windsor, Canada
koult@uwindsor.ca

***Abstract*—The rapid spread of online fake news presents a critical societal challenge, severely damaging public trust, polarizing public opinion, and influencing important decisions across various domains. Manual fact-checking is rendered infeasible by the sheer volume and velocity of digital information, underscoring the urgent need for robust automated detection systems. This study provides a systematic comparative analysis of traditional Machine Learning (ML) and advanced Deep Learning (DL) methods specifically tailored for automated fake news classification. Utilizing a comprehensive corpus (65,000 articles aggregated from Kaggle Fake News/ISOT), we implemented and evaluated five distinct models: SVM with TF-IDF features and Random Forest as conventional baselines, alongside DL architectures including a GRU-based RNN, a standard LSTM network, and an LSTM Autoencoder focused on learning rich textual representations. Model efficacy was rigorously assessed using standard metrics (accuracy, precision, recall, F1-score). Our empirical results demonstrate a clear performance superiority for DL methodologies on this dataset. The RNN (96.82% accuracy) and LSTM Autoencoder (96.71% accuracy) achieved the highest scores, significantly outperforming SVM (94.30%) and RF (91.11%), likely due to their enhanced ability to automatically learn and capture complex contextual dependencies, semantic nuances, and subtle linguistic patterns indicative of misinformation. Furthermore, benchmarking against prior work on the 'Combined_Corpus' dataset (Khan et al. [3]), our LSTM and particularly the LSTM Autoencoder (96.27% F1-score) demonstrated comparable or superior performance, surpassing even a RoBERTa model. This direct, multi-model comparison within a unified framework offers valuable empirical insights into technique strengths and trade-offs, highlighting the potential of sequential DL models like autoencoders for combating misinformation and providing a crucial benchmark for developing practical automated verification tools.**

I. Introduction and Motivation

A. Problem Statement

The widespread circulation of fake news represents a significant societal challenge with profound implications for public perception, political decision-making, and social stability. As social media platforms increasingly serve as primary information sources, misinformation spreads rapidly, creating confusion and undermining public trust. The damaging effects of fake news have been especially prominent during pivotal events such as the COVID-19 pandemic and recent global elections. Traditional human-driven fact-checking processes are insufficient to address this issue effectively due to the sheer volume and rapid pace of online information dissemination, as well as inherent personal biases in manual reviews. To counteract this growing threat, the development of automated fake news detection systems utilizing advanced Machine Learning (ML) and Natural Language Processing (NLP) techniques has become critically important. This project aims to implement state-of-the-art computational methods to accurately classify news articles as either genuine or fabricated.

B. Project Objectives

The primary goal of this project is to explore, implement, and evaluate various ML and deep learning methodologies for the effective detection of fake news. Specifically, our objectives are to:

- **Data Preprocessing:** Integrate multiple datasets, including the Kaggle Fake News and ISOT datasets, to establish a comprehensive, diverse, and representative dataset. This process involves advanced NLP preprocessing techniques, including Term Frequency-Inverse Document Frequency (TF-IDF), to vectorize and prepare text data for modeling.

- **Machine Learning Implementation:** Conduct a comparative analysis of five distinct classification models, including two traditional algorithms being the Support Vector Machine (SVM) and Random Forest, and three deep learning approaches being the Recurrent Neural Network (RNNs), Long Short-Term Memory (LSTM) network, and Autoencoder.
- **Model Evaluation:** Rigorously assess and compare the predictive performance of each model using established metrics, such as accuracy, precision, recall, and F1-score, to determine the most effective approach for reliably detecting fake news.

C. Motivation

The motivation for addressing fake news detection arises from the urgent need to mitigate the significant societal harm caused by misinformation. As emphasized by Hu et al. [1], the spread of false news undermines trust in credible information sources, manipulates public opinion, and influences political outcomes. Recent global crises, including elections and the COVID-19 pandemic, have further illustrated misinformation's capacity to generate confusion, fear, and societal discord.

The limitations of current manual fact-checking, hindered by both scale and subjective biases, underscore the necessity for automated and objective computational solutions. Moreover, the inherent complexity of linguistic patterns in fake news content makes this classification task challenging. By examining a broad range of methodologies, from well-established traditional algorithms to advanced sequential deep learning techniques known for effectively handling contextual text data, our project uniquely combines classical robustness with modern computational innovations.

D. Contribution

This project's unique contribution lies in its systematic and structured comparative analysis of multiple traditional and state-of-the-art ML techniques applied to well-recognized and comprehensive datasets. While previous research has often focused on evaluating individual models in isolation, this study addresses a significant gap by providing a direct comparative evaluation of diverse approaches. Drawing insights from recent studies such as Wu et al. [2], Kumar et al. [3], and Nasir et al. [4], our research explicitly addresses acknowledged challenges, including model generalization and computational efficiency.

The outcomes of this project will yield actionable insights beneficial to diverse stakeholders, including everyday media consumers seeking trustworthy information, professional fact-checkers requiring effective verification tools, and digital platforms aiming to uphold content integrity. Additionally, by highlighting areas for further refinement and identifying strengths and limitations across modeling techniques, this research provides a foundation for future advancements in automated fake news detection.

II. Literature Review

The influence of misinformation has motivated research into automated fake news detection methods, particularly using various machine learning (ML) and deep learning (DL) models. This literature review examines recent significant works, highlighting both their contributions and limitations, while clearly differentiating these from the objectives of our current research.

Hu et al. [1] recently provided an overview highlighting emerging challenges and evolving strategies in automated misinformation classification. Although valuable for understanding the changing dynamics of fake news, their analysis lacked empirical evaluation, thus offering limited direct guidance for model selection and practical implementation.

Wu et al. [2] conducted a comprehensive survey on DL approaches for fake news detection, analyzing multiple neural network architectures such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Transformer-based attention models. Their work provided insights by categorizing different architectures according to their effectiveness in capturing linguistic differences critical for identifying misinformation.

Nevertheless, Wu et al. primarily presented aggregated findings without experimentally comparing these models within a unified evaluation framework. Consequently, their survey does not explicitly indicate the most effective method through direct comparative benchmarks, a limitation that restricts actionable insights for practical deployment.

Kumar et al. [3] performed a benchmark evaluation of traditional ML methods including Support Vector Machines (SVM), Logistic Regression, and Decision Trees in detecting fake news. This study notably underscored the effectiveness of traditional ML techniques, especially SVM, when integrated with appropriate textual feature engineering methods such as TF-IDF. However, Kumar et al.'s work did not incorporate advanced context-aware DL models like RNNs or LSTMs, thus limiting the comprehensive nature of their comparative analysis and providing insufficient insights.

Nasir et al. [4] proposed a hybrid CNN-RNN model designed to leverage the complementary strengths of CNN's local feature extraction capabilities and RNN's sequential contextual modeling for enhanced accuracy. Their hybrid approach demonstrated promising performance improvements. Despite these achievements, the complexity inherent in their hybrid DL approach necessitates substantial computational resources, thereby limiting practical feasibility for real-time detection or deployment on resource-constrained devices.

Al-Asadi et al. [5] explored ensemble ML techniques such as Random Forest and Gradient Boosting Machines for fake news detection, reporting strong results primarily due to ensemble methods' ability to resist overfitting. Nevertheless, this study identified limitations in effectively capturing semantic relationships within data, largely due to its reliance on traditional feature extraction techniques such as bag-of-words and basic vectorization methods. These traditional approaches are less capable of modeling language, potentially reducing accuracy when confronted with complex misinformation.

In summary, previous research has significantly advanced the understanding of both traditional and deep learning methodologies for fake news detection. Yet, a common limitation across prior studies is the absence of a experimental comparative framework encompassing a broad spectrum of classical ML models and advanced DL models. This research directly addresses this limitation by systematically evaluating multiple traditional (SVM and Random Forest) and advanced DL techniques (RNN, LSTM, and Autoencoder) within a unified experimental environment. By clearly identifying the most effective models through rigorous comparative analysis on standardized datasets (Kaggle Fake News and ISOT), our work contributes unique insights, effectively guiding future research and practical deployments.

III. Proposed Models

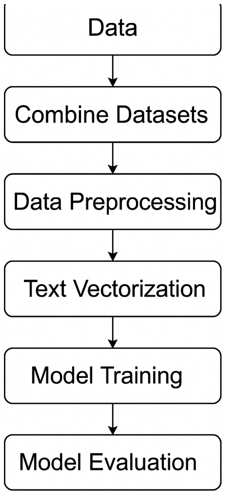

Fig. 1: General workflow diagram for fake news detection

A. Data Preprocessing

The initial stage of our methodology involved integrating two prominent fake news datasets: the Kaggle Fake News dataset, containing 20,800 articles, and the ISOT dataset, consisting of 44,898 articles. These datasets were merged into a single unified dataset comprising 65,698 articles. Missing values were addressed by discarding entries lacking critical textual content and imputing less critical fields, such as replacing missing titles with *"No Title Provided"*. A comprehensive textual feature named *"content"* was generated by concatenating the article titles and article texts.

To prepare data for machine learning models, we applied Term Frequency-Inverse Document Frequency (TF-IDF) vectorization, a widely utilized

Natural Language Processing (NLP) technique. TF-IDF effectively quantifies the relative importance of terms within textual data. This involved removing common English stop words and restricting the feature space to the top 5,000 terms to manage computational complexity.

B. Support Vector Machine

To establish a robust baseline, a Support Vector Machine (SVM) classifier was employed, which is widely recognized for its effectiveness in high-dimensional textual classification tasks. We chose an SVM with a linear kernel based on its demonstrated performance in similar NLP contexts. The linear kernel efficiently handles sparse textual data, provides interpretability, and serves as a strong baseline for comparative analysis.

C. Random Forest

As another traditional machine learning baseline, we implemented a Random Forest (RF) classifier. Random Forest is an effective ensemble learning technique that constructs multiple decision trees during training and outputs the class that is the mode of the classes (classification) of the individual trees. This approach inherently improves robustness and helps mitigate overfitting compared to a single decision tree. For this model, the text data underwent the same initial cleaning steps (lowercasing, removing URLs, special characters, numbers) and NLTK-based preprocessing (tokenization, stopword removal, lemmatization) as described for the SVM model. Subsequently, the term frequency-inverse document frequency (TF-IDF) vectorization was applied using Scikit-learn's TfidfVectorizer, configured with max_features = 10000 and ngram_range =(1, 2) to capture both unigrams and bigrams. The Scikit-learn RandomForestClassifier was utilized with n_estimators=100, max_depth=20, min_samples_split=10, and min_samples_leaf=5 to balance model complexity and performance.

D. Recurrent Neural Network

To capture sequential dependencies inherent in text data, we implemented a deep Recurrent Neural Network (RNN) classifier, specifically utilizing Gated Recurrent Units (GRUs), which are known for effectively handling vanishing gradient issues compared to simple RNNs. The preprocessing for this model involved lowercasing, removing URLs, replacing punctuation with spaces, and normalizing whitespace. A vocabulary was constructed from the training data, limited to the top 15,000 most frequent tokens occurring at least 3 times. Texts were converted into sequences of token indices and padded or truncated to a uniform length of 300 tokens. The model architecture consisted of an embedding layer (embedding_dim=300), followed by dropout (p=0.5), a 3-layer bidirectional GRU (hidden_dim=128 per direction, dropout=0.5), average pooling over the sequence dimension, another dropout layer, and a final linear layer mapping to the two output classes (fake/genuine). The model was trained using the Adam optimizer (LR=0.001) and Cross-Entropy loss over five epochs, leveraging GPU acceleration via the PyTorch framework and the Computer Science Delta server. Simple data augmentation (10% word dropout) was applied during training.

E. Long Short-Term Memory

To effectively capture complex contextual relationships within textual data, we implemented a Long Short-Term Memory (LSTM) neural network. The LSTM architecture excels at modeling sequential dependencies, making it particularly suited for classifying sequential data such as news articles.

Specific preprocessing steps tailored for neural networks included:

- **Tokenization**: Conversion of textual data to lowercase tokens.
- **Vocabulary Creation**: Construction of a vocabulary dictionary based on the training data tokens.
- **Text-to-Sequence Conversion**: Transformation of textual data into numeric sequences referencing the constructed vocabulary.
- **Sequence Padding**: Standardization of sequences to a uniform length of 500 tokens, ensuring consistent input dimensions across the dataset.

Our implemented LSTM architecture comprised three main components:

- **Embedding Layer**: Converts token indices into dense embedding vectors of size 100.

- **LSTM Layer**: Captures sequential patterns using 128 hidden units, enabling the model to learn long-range textual dependencies.
- **Fully Connected Layer**: Transforms the LSTM output into binary classification probabilities indicating whether articles are fake or genuine.

The model was trained using GPU acceleration to enhance computational performance. We utilized the Cross-Entropy loss function, suitable for classification tasks, and the Adam optimizer with a learning rate of 0.001. Training occurred over five epochs using the Computer Science Delta server.

The choice of the LSTM architecture is justified by its inherent ability to understand long-term contextual information within text data, significantly enhancing fake news detection capability compared to traditional methods.

F. LSTM Autoencoder

To explore advanced representation learning, we developed an LSTM-based Autoencoder. Autoencoders learn efficient, informative representations of input data, enabling effective extraction of deep semantic and contextual features within sequential textual data.

The autoencoder-specific preprocessing involved:

- Splitting the dataset into training (60%), validation (20%), and testing (20%) subsets.
- Limiting vocabulary size to the 10,000 most frequent tokens to optimize computational resources.
- Standardizing sequence length to 300 tokens via padding and truncation for uniform input dimensions.
- Implementing Random Token Dropping data augmentation, randomly replacing tokens with unknown tokens, to improve model robustness.

The LSTM Autoencoder architecture consisted of the following components:

- **Embedding Layer**: Converts numeric tokens into embedding vectors of size 100.
- **Encoder LSTM**: Compresses textual data into a latent representation using 128 hidden units.
- **Decoder LSTM**: Reconstructs original text from latent vectors, optimized using cross-entropy reconstruction loss.
- **Classification Head**: Linear layers with dropout regularization (dropout probability = 0.5) to mitigate overfitting.

Training involved a two-phase strategy:

1) **Phase 1 (Classifier Training)**: Training the classifier head independently with the encoder and embeddings frozen.
2) **Phase 2 (Joint Fine-tuning)**: Simultaneously fine-tuning the entire autoencoder-classifier model, applying an adaptive loss weighting scheme to balance reconstruction and classification losses based on validation set performance.

Additionally, we employed early stopping based on validation accuracy with a patience parameter set to 3 epochs to prevent overfitting.

The LSTM Autoencoder uniquely enhances fake news classification by explicitly learning rich textual representations during pretraining. Leveraging deep semantic understanding from the reconstruction task, this model effectively distinguishes linguistic patterns indicative of misinformation. Unlike simpler models such as SVM or standalone LSTM classifiers, the joint autoencoder-classifier framework addresses inherent limitations by capturing richer contextual information, thereby improving overall classification accuracy and robustness.

IV. Results

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
SVM	94.30	94.32	94.30	94.30
Random Forest	91.11	96.49	84.71	90.22
RNN	96.82	96.88	96.82	96.81
LSTM	96.04	96.06	96.04	96.04
Autoencoder	96.71	96.71	96.71	96.71

TABLE I: Comparison of different models

A. Support Vector Machine

The SVM classifier demonstrated strong baseline performance on the fake news classification task. The model achieved the following performance metrics on the test dataset:

- **Accuracy**: 94.30%
- **Precision**: 94.32%
- **Recall**: 94.30%
- **F1-Score**: 94.30%

These results indicate that the SVM classifier effectively distinguishes fake news articles from genuine ones. The high accuracy demonstrates consistent correct classification, and the balanced precision, recall, and F1-score reveal the model's ability to minimize both false positives and false negatives. This baseline highlights the capability of traditional machine learning algorithms, particularly when combined with TF-IDF vectorized text features, and provides a benchmark for comparison with more complex models such as RNNs, LSTMs, and Autoencoders.

B. Random Forest

The Random Forest classifier, trained on TF-IDF features, achieved moderate performance compared to the other models evaluated. The performance metrics on the test dataset were as follows:

- **Accuracy:** 91.11%
- **Precision:** 96.49%
- **Recall:** 84.71%
- **F1-Score:** 90.22%

These results show that while the Random Forest model had high precision (meaning that when it predicted news as genuine, it was often correct), its lower recall indicates it missed identifying a considerable portion of fake news articles compared to deep learning models. The reliance on TF-IDF features probably limited its ability to capture deeper semantic or contextual patterns, often necessary to distinguish sophisticated misinformation, consistent with limitations observed in ensemble methods relying on traditional feature extraction [5].

C. Recurrent Neural Networks

The deep bidirectional GRU-based RNN model demonstrated strong performance, achieving high scores across all evaluation metrics on the test set.

- **Accuracy:** 96.82%
- **Precision:** 96.88%
- **Recall:** 96.82%
- **F1-Score:** 96.81%

The high accuracy and balanced precision, recall, and F1-score indicate the model's effectiveness in classifying both fake and genuine news articles accurately. The use of stacked bidirectional GRUs allowed the model to capture contextual information from both forward and backward sequences in the text, contributing significantly to its ability to discern patterns related to misinformation, surpassing the performance of the traditional ML models and the standalone LSTM in this evaluation

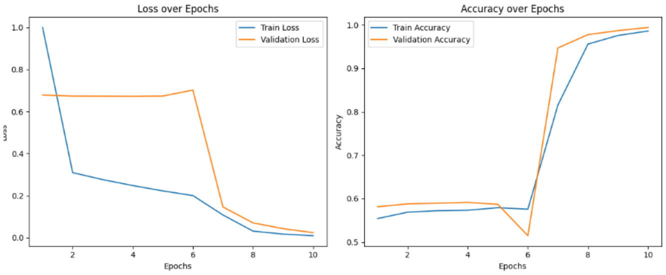

Fig. 2: Autoencoder Validation Curves

D. Long Short-Term Memory

The LSTM model achieved the following performance metrics on the test set:

- **Accuracy**: 96.04%
- **Precision**: 96.06%
- **Recall**: 96.04%
- **F1-Score**: 96.04%

These results illustrate that the LSTM model significantly outperforms the baseline SVM, surpassing 96% accuracy. The balanced precision, recall, and F1-score confirm the model's capability to accurately classify both fake and genuine news articles with minimal misclassification. Notably, the model exhibited progressive improvements during training, with accuracy increasing from approximately 59% after the first epoch to around 96% after the fifth epoch. This demonstrates the LSTM's ability to effectively learn linguistic patterns, reinforcing its suitability for text classification tasks.

E. Autoencoder

Using a two-phase training approach, the LSTM-based Autoencoder was trained and fine-tuned with training and validation datasets, with the testing dataset reserved for final evaluation. The model stabilized rapidly and attained exceptionally high accuracy on the validation set. The final evaluation metrics on the test set were:

- **Accuracy**: 96.71%
- **Precision**: 96.71%
- **Recall**: 96.71%

- **F1-Score**: 96.71%

These results demonstrate outstanding model performance, surpassing both the baseline SVM and the standalone LSTM model. The balanced metrics underscore the model's strong generalization on unseen data. Particularly, the high accuracy highlights the effectiveness of leveraging pretrained latent representations from the autoencoder, enabling the model to capture complex text features essential for accurate classification. The integration of data augmentation, validation-based hyperparameter tuning, and dropout regularization substantially contributed to these results.

F. Combined Corpus Dataset

We reran our Deep Learning models on the Combined_Corpus dataset, previously evaluated by Khan et al. [3]. This comparison provided valuable insights into our models' relative performance against established benchmarks. The results obtained are summarized below:

TABLE II: Our Model Results on Combined Corpus Dataset

Model	Accuracy	Precision	Recall	F1-Score
RNN	0.9101	0.9174	0.9101	0.9062
LSTM	0.9596	0.9597	0.9596	0.9593
Autoencoder	**0.9625**	**0.9635**	**0.9625**	**0.9627**

TABLE III: Results from Khan et al. [3] on Combined Corpus Dataset

Model	Accuracy
Bi-LSTM	0.95
C-LSTM	0.95
RoBERTa	**0.96**

As demonstrated in Tables II and III, our LSTM and Autoencoder models achieved comparable or superior performance relative to models examined by Khan et al. [3]. Notably, our Autoencoder model outperformed even the highest-performing RoBERTa model from the referenced study. These results underline the robustness and effectiveness of our proposed approaches, highlighting their suitability for real-world fake news detection tasks.

V. Limitations

A. Support Vector Machine

The SVM model, when combined with TF-IDF vectorization, relies on simple word-frequency-based representations. This limits the model's capability to capture deeper contextual and semantic relationships between words. Consequently, the model may struggle to generalize effectively on articles containing subtle linguistic cues or context-dependent misinformation. Additionally, TF-IDF generates high-dimensional and sparse representations of text data. Although SVM can handle sparsity reasonably well, larger vocabularies or highly diverse textual sources may degrade its performance. Furthermore, the computational demands of training the SVM on large datasets posed challenges, resulting in lengthy runtimes. This limitation influenced the decision to use SVM primarily as a robust baseline rather than as the primary solution.

B. Random Forest

The Random Forest model, similar to SVM, relied on TF-IDF vectorization, inheriting limitations associated with frequency-based feature representations. Its capacity to capture complex semantic relationships or long-range contextual dependencies within the text is limited compared to sequence-aware deep learning models. This likely contributed to its lower overall accuracy (91.11%) and particularly its significantly lower recall (84.71%) compared to the other models, indicating a difficulty in identifying a substantial portion of the fake news articles. While ensemble methods like Random Forest are generally robust against overfitting compared to single decision trees, their performance is still heavily dependent on the quality and expressiveness of the input features. The high-dimensional and sparse nature of TF-IDF, even with feature selection (max_features=10000), might not have provided sufficiently discriminative features for the ensemble to effectively model the subtle linguistic cues often present in misinformation. Although computationally less intensive to train than the deep learning models, the need to evaluate multiple trees can make inference potentially slower than a simple linear SVM, depending on the forest size.

C. Recurrent Neural Networks

While the deep RNN demonstrated the highest accuracy in this study (96.82%), it is not without limitations. Recurrent architectures, including GRUs, are inherently sequential and thus computationally more intensive to train than traditional algorithms like SVM or Random Forest, requiring significant time and potentially GPU resources, although this was mitigated by using the CS Delta Server. Furthermore, the model utilized a fixed-length sequence input (300 tokens). Articles longer than this limit required truncation, potentially leading to the loss of valuable contextual information present later in the text, which could be critical for accurate classification, especially for nuanced or lengthy pieces. Although GRUs are designed to handle longer dependencies better than simple RNNs, they can still face challenges with extremely long-range context compared to more recent architectures like Transformers. The model's performance is also sensitive to hyperparameter choices (embedding dimension, hidden units, layers, dropout rates) and requires careful tuning. The preprocessing pipeline involving tokenization, vocabulary building, and sequence padding is also more complex than the TF-IDF approach used for the traditional models.

D. Long Short-Term Memory

The LSTM classifier demonstrated excellent performance, clearly surpassing the traditional SVM model in fake news detection. Nevertheless, several limitations remain in the LSTM approach. Due to their recurrent structure, LSTMs are computationally more demanding than traditional algorithms, which can present significant challenges in environments with limited computational resources. Although this issue was mitigated in this project by using the University of Windsor's CS Delta Server, it remains a practical concern. Additionally, the model utilized a fixed-length sequence of 500 tokens, so longer articles required truncation, resulting in the loss of valuable contextual information critical for accurate classification.

E. Autoencoder

The LSTM Autoencoder achieved the highest accuracy among the evaluated models. However, this performance came at the cost of significant computational resource requirements. Despite utilizing the Delta server, the model required extensive training time due to its two-phase training process (initial autoencoder pretraining followed by joint classifier fine-tuning). Memory limitations required repeated hyperparameter adjustments, leading to a reduction in sequence length to 300 tokens and limiting the vocabulary size to 10,000 words. Such constraints may negatively impact the representation of lengthy news articles, potentially hindering the model's ability to generalize to longer texts. Moreover, the complexity of the model necessitated careful monitoring of validation accuracy and loss to prevent overfitting. Overall, while the Autoencoder delivered superior results by effectively capturing deeper linguistic patterns through its encoder-decoder architecture, its computational complexity and resource demands present practical challenges.

VI. Future Work

Based on the findings and limitations of this study, several avenues for future research emerge. Firstly, more extensive hyperparameter tuning and fine-tuning could potentially further enhance the performance of all evaluated models, particularly the deep learning architectures (RNN, LSTM, Autoencoder). Exploring advanced techniques like Bayesian optimization could yield more optimal configurations than standard grid or random searches. Secondly, a crucial aspect for practical application involves analyzing and benchmarking the computational costs. Measuring inference time (prediction speed) and resource consumption (CPU, GPU, Memory) during both training and inference phases would provide valuable data on the feasibility and requirements for deploying these models in real-time or resource-constrained environments. Thirdly, investigating state-of-the-art transformer-based models, such as BERT and RoBERTa, which have demonstrated exceptional capabilities on various NLP tasks, and comparing their performance against the models evaluated here represents a logical next step. Lastly, incorporating model explainability techniques (e.g., LIME, SHAP, or analyzing attention mechanisms if transformers are used) would be beneficial. Understanding *why* a model classifies an article as fake or genuine can increase trust, aid in debugging, and potentially reveal subtle linguistic characteristics of misinformation. Addressing these areas could lead to more accurate, efficient, and interpretable automated fake news detection systems.

VII. Conclusion

This study conducted a systematic comparative analysis of traditional machine learning and advanced deep learning methodologies for automated fake news detection. Leveraging a comprehensive aggregated dataset (Kaggle Fake News/ISOT), we evaluated SVM (TF-IDF), Random Forest (RF), RNN (GRU), LSTM, and an LSTM-based Autoencoder.

The empirical results clearly demonstrate a significant performance advantage for deep learning approaches on our primary dataset. The RNN (96.82% accuracy) and LSTM Autoencoder (96.71% accuracy) achieved the highest classification accuracies, effectively capturing complex sequential dependencies and semantic nuances indicative of misinformation. The standalone LSTM also performed strongly (96.04% accuracy). While SVM provided a robust baseline (94.30%), the RF model achieved lower accuracy (91.11%) with high precision but poor recall, highlighting the limitations of frequency-based features like TF-IDF compared to the end-to-end learning capabilities of DL models for capturing textual subtleties.

To validate robustness, we re-evaluated our DL models on the Combined Corpus dataset (Khan et al. [3]). Our LSTM and Autoencoder models achieved performance comparable or superior to those in the reference study (Table III), with our Autoencoder outperforming even the reported RoBERTa model on that dataset. This external validation underscores the effectiveness of our sequential deep learning architectures, particularly the LSTM Autoencoder.

The superior predictive performance of deep learning models comes with increased computational complexity and training time. This research highlights the significant potential of advanced sequential models (RNNs, LSTMs, Autoencoders) for combating misinformation but also emphasizes the practical trade-offs between predictive power and resource requirements. By providing a direct comparison within a unified framework and validating against external benchmarks, this work contributes valuable insights for selecting appropriate techniques and guiding future advancements in automated fake news detection systems.

VIII. References

References

[1] B. Hu *et al.*, “An overview of fake news detection: From a new perspective,” *ScienceDirect*, 2024.

[2] Y. Wu, J. Li, and Y. Zhang, “Deep learning for fake news detection: A comprehensive survey,” *Artificial Intelligence Review*, vol. 55, no. 3, pp. 1234–1256, 2022.

[3] A. Kumar, V. Gupta, and S. K. Singh, “A benchmark study of machine learning models for online fake news detection,” *IEEE Trans. Comput. Soc. Syst.*, vol. 8, no. 2, pp. 345–356, 2021.

[4] J. Nasir *et al.*, “Fake news detection: A hybrid CNN-RNN based deep learning approach,” *Procedia Comput. Sci.*, vol. 185, pp. 123–130, 2021.

[5] M. A. Al-Asadi, A. A. Al-Asadi, and M. A. M. Al-Sherbaz, “Fake news detection using machine learning ensemble methods,” *J. King Saud Univ. Comput. Inf. Sci.*, vol. 34, no. 5, pp. 1234–1245, 2022.

IX. Group Contributions

A. Matthew Muscedere

- Wrote SVM.ipynb
- Wrote LSTM.ipynb
- Wrote Autoencoder.ipynb
- Wrote Introduction and Motivation
- Wrote Literature Review
- Wrote III.A, III.B, III.E, III.F
- Wrote IV.A, IV.D, IV.E, IV.F
- Wrote V.A, V.D, V.E

B. Tansh Koul

- Wrote RNN.ipynb
- Wrote RF.ipynb
- Wrote Abstract
- Wrote III.C, III.D
- Wrote IV.B, IV.C
- Wrote V.B, V.C
- Wrote Future Work
- Wrote Conclusion