Hybrid Fake News Detection Using BERT and Speaker Credibility Ranking via PageRank

Manasa K
Department of CSE
PES University
Bangalore, India
manasakb160@gmail.com

Haripriya
Department of CSE
PES University
Bangalore, India
haripriyal1604@gmail.com

Chandrashekhar Pomu Chavan
Department of CSE
PES University
Bangalore, India
cpchavan@pes.edu

***Abstract*—The spread of fake news on digital platforms has become an increasingly serious issue, influencing public opinion and eroding trust in media. While traditional detection models rely heavily on linguistic patterns, they often overlook critical contextual factors such as the credibility and influence of the information source. In this paper, we propose a hybrid approach that combines semantic understanding of text using BERT (Bidirectional Encoder Representations from Transformers) with speaker-centric metadata. We compute each speaker's credibility based on their historical truthfulness and apply the PageRank algorithm on a speaker-topic bipartite graph to quantify their influence. These metadata features are fused with BERT-based embeddings of news statements to form a unified feature vector. A Random Forest classifier is then trained on this representation using a real-world Kaggle dataset. Experimental results demonstrate that our model outperforms text-only and metadata-only baselines, achieving higher accuracy and better handling of ambiguous cases. Additionally, we present visual analyses of speaker credibility and influence, highlighting the benefits of incorporating trust-aware signals in fake news detection.**

***Index Terms*—Fake News Detection, BERT, PageRank, Credibility Score, Speaker Graph, Semantic Representation, Random Forest.**

I. Introduction

The rapid spread of misleading information on digital platforms has emerged as a critical threat to public trust and the integrity of democratic processes. Fake news—intentionally misleading or fabricated content designed to deceive audiences—has the potential to distort opinions, exacerbate social divisions, and even influence election outcomes. The pervasive nature of fake news poses significant challenges for automated detection systems, which must contend with linguistic subtleties such as sarcasm, bias, and evolving vernacular. Additionally, the sheer volume of online content necessitates robust and scalable algorithms that can effectively discern truth from deception.

While natural language processing methods, especially those based on transformer architectures like BERT (Bidirectional Encoder Representations from Transformers), have achieved success in capturing semantic and contextual nuances in textual data [1], [3], they often neglect an equally important dimension of news assessment: the credibility of the source. Most existing systems focus primarily on the textual content and overlook critical metadata such as speaker identity, historical truthfulness, and contextual factors that contribute to a speaker's reliability [4], [6], [8]. This limitation is particularly significant because the trustworthiness of a news source plays a pivotal role in evaluating the veracity of its claims [14]. To address this challenge, it is essential to integrate both deep semantic analysis and trust-based features into the detection framework.

In this work, we propose a hybrid fake news detection model that combines BERT-based semantic embeddings with speaker-level metadata, including credibility scores and influence measures derived from a PageRank analysis of a speaker-topic bipartite graph [21], [23]. A speaker's credibility is computed using historical distributions of truthful and deceptive statements, while their influence within the discourse network is captured using PageRank—indicating their topical coverage and prominence [24]. These trust-based features are fused with the semantic embeddings into a unified representation, allowing the model to assess both the meaning of the content and the credibility of its source.

The remainder of this paper is organized as follows: Section II reviews related work, Section III details the proposed methodology, Section IV presents experimental results and discussion, and Section V concludes with future directions.

II. Related Work

The domain of fake news detection has evolved rapidly in recent years, driven by the emergence of deep learning and graph-based techniques. In [1], a comparative evaluation was conducted between BERT-like models and larger language models trained on AI-annotated data, showing the superiority of contextual embeddings for textual understanding. The study in [2] introduced a multi-modal credibility framework combining BERT and CNNs, emphasizing the importance of fusing linguistic and visual features for better reliability estimation.

A unified BERT training strategy was proposed in [3], demonstrating that tailored fine-tuning can significantly boost classification performance. Similarly, [4] explored stance detection using BERT embeddings to evaluate the credibility of social media posts, focusing on the alignment between statements and known facts. The authors in [5] contributed an explainable Tsetlin Machine framework that included credibility scoring, highlighting the value of transparent learning in high-stakes domains like misinformation.

Credibility was also central to the work in [6], where early detection was approached by modeling the trustworthiness of publishers, news content, and users. The study in [7] combined BERT with joint learning for fake news classification, introducing a method to optimize feature representation across multiple input types. Credibility-based classification was further advanced in [8], where metadata was fused with textual information to capture deeper context.

FakeBERT, proposed in [9], used a BERT-based architecture tuned for social media inputs, outperforming baseline models in fake news recognition. In [10], the authors developed a multi-modal transformer using hierarchical visual features alongside textual ones, enhancing classification in noisy environments. A capsule network with BERT was integrated in [11], creating a hybrid deep learning system capable of capturing complex patterns in fake and real news.

An ensemble deep learning framework for fake news and political statement classification was proposed in [12], using diverse neural models to improve generalization. Discourse-level structures were explored in [13] through a hierarchical learning approach, capturing inter-sentence relationships crucial for deceptive content detection. In [14], a tracking tool was introduced for fake news collection and visualization, aiding in pattern discovery and real-time monitoring.

The FakeNewsNet repository presented in [15] provided a rich dataset combining content, context, and spatiotemporal signals, enabling more robust model development. A broad survey in [16] categorized existing methods, noting the need for credibility-aware and hybrid detection systems. The comprehensive analysis in [17] examined theoretical foundations, detection architectures, and open challenges in the domain.

A foundational study in [18] illustrated that false news spreads faster than true news on social networks, reinforcing the necessity for proactive detection models. Fact-checking automation was examined in [19], where task formulations and evaluation metrics were discussed. Sentence-level inference and dataset design for deep language understanding were covered in [20], laying the groundwork for logic-driven models.

In [21], user profiling was used to detect misinformation, leveraging behavioral patterns to identify suspicious actors. Contrasting social viewpoints were exploited in [22] to validate news, capturing disagreement as a cue for inconsistency. Graph structures were central to [23], where rumor propagation was analyzed via network topology. The use of geometric deep learning in [24] enabled structural reasoning in social media graphs for fake news detection. Lastly, knowledge graphs were used in [25] to verify content by linking entities and facts to structured external databases.

This diverse body of literature establishes that while BERT and transformer-based models are critical for semantic understanding, combining them with credibility metrics, user profiling, visual cues, and graph-based analysis yields significantly better results. Our proposed model draws inspiration from these approaches by integrating BERT embeddings, speaker credibility scoring, and a PageRank-based speaker-topic graph to develop a robust and trust-aware fake news classification system.

III. Proposed Methodology

A. Overview of the Proposed Architecture

The proposed architecture is a hybrid fake news detection framework that combines semantic understanding from deep language models with trust-based metadata features to improve classification accuracy. The pipeline begins with the LIAR dataset, which contains short political statements labeled with ground truth along with speaker-related metadata such as job title, party affiliation, and previous truth/fake counts.

First, we preprocess the data by handling missing values, encoding categorical variables, and computing a *credibility score* for each speaker based on their historical truth ratings. We then construct a bipartite speaker-topic graph and apply the PageRank algorithm to assign influence scores to speakers based on their connected topics and frequency of mentions.

Simultaneously, we use BERT (Bidirectional Encoder Representations from Transformers) to generate contextual embeddings for each news statement. These semantic features are then fused with speaker-level features including credibility score, PageRank score, party affiliation, and speaker ID to form a unified feature vector.

This enriched representation is fed into a Random Forest classifier trained to predict one of six truth labels. The combination of deep semantic modeling with credibility-aware speaker profiling enables our model to better handle ambiguous and context-sensitive statements. The overall architecture leverages both natural language processing and graph-based reasoning to create a trust-aware classification system.

B. System Architecture Diagram and Explanation

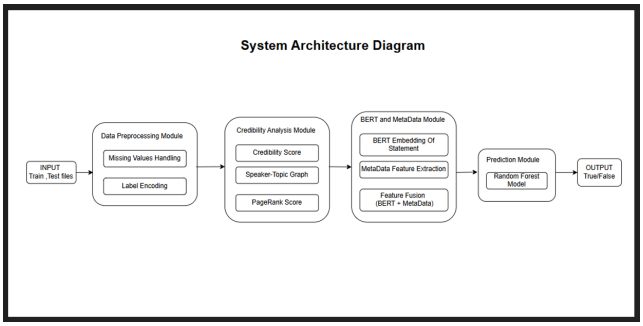

Fig. 1. Architecture of the Fake News Detection System

1) Input: Train and Test Files: The system accepts two primary inputs in '.tsv' format: `train.tsv` and `test.tsv`. The training file contains labeled statements, while the test file includes unlabeled data for evaluation. Each entry in these files consists of a news statement along with metadata fields such as the speaker's name, political party, job title, state information, subject tags, context, and a truthfulness label (for training only). This structured dataset provides both semantic and contextual information critical to the credibility analysis process.

2) Data Preprocessing Module: This module is responsible for ensuring data consistency, cleanliness, and suitability for machine learning tasks:

- **Missing Values Handling:** Missing values in metadata fields (e.g., speaker job, state) are either imputed using mode-based strategies or dropped when necessary. The objective is to minimize information loss while maintaining dataset integrity.
- **Label Encoding:** Truthfulness labels such as "true," "mostly-true," "half-true," "barely-true," "false," and "pants-fire" are mapped to a binary classification scheme (e.g., true = 1, false = 0). Categorical metadata (e.g., political party, speaker name) is also encoded into numerical form using label encoding or one-hot encoding.

This module outputs a cleaned and numerically represented dataset for further processing.

3) Credibility Analysis Module: This module evaluates the trustworthiness and influence of speakers using graph-based techniques:

- **Credibility Score Computation:** For each speaker, a credibility score is computed based on the proportion of truthful versus deceptive statements in the training dataset. This historical record provides a quantitative basis to estimate speaker reliability.
- **Speaker-Topic Graph Construction:** A bipartite graph is formed, where nodes represent speakers and topics. An edge exists between a speaker and a topic if the speaker has issued statements on that topic. This graph captures the diversity and focus of speaker-topic relationships.
- **PageRank Score Computation:** The PageRank algorithm is applied to the speaker-topic graph to estimate speaker influence. Speakers who are central and well-connected in the graph receive higher PageRank scores, indicating broader topical coverage and relational prominence.

The outputs of this module are two credibility-related features: the historical credibility score and the graph-based PageRank score.

4) BERT and Metadata Module: This module combines deep language understanding with contextual speaker information to form enriched input representations:

- **BERT Embedding of Statement:** Each news statement is passed through a pre-trained BERT (Bidirectional Encoder Representations from Transformers) model to generate a dense, high-dimensional vector embedding. These embeddings capture rich contextual semantics and relationships between words in the sentence.
- **Metadata Feature Extraction:** Speaker metadata (identity, party, job title, state, etc.), as well as credibility and PageRank scores, are extracted and numerically encoded. These features offer external context that is not available from the text alone but is vital for assessing the trustworthiness of the source.
- **Feature Fusion:** The BERT embeddings and metadata features are concatenated to create a single feature vector. This fusion ensures that both linguistic and contextual cues are jointly considered by the downstream classifier.

5) Prediction Module: The unified feature vectors are fed into a supervised machine learning model for classification:

- **Random Forest Classifier:** This ensemble method constructs multiple decision trees and aggregates their predictions to improve classification robustness and generalization. The Random Forest model is particularly effective at handling heterogeneous feature types (text + metadata) and provides insights into feature importance.

The model is trained on the fused features extracted from the training data and is evaluated on the test data to assess generalization.

6) Output: Truth Label: The final output of the system is a binary prediction for each news statement in the test set:

- **True (1)**: The statement is classified as truthful.
- **False (0)**: The statement is classified as fake or misleading.

These predictions reflect a hybrid understanding of semantic content and source-based credibility, making the system more robust against surface-level linguistic manipulation.

Algorithm 1 Fake News Detection using BERT and Metadata Fusion

1: **Input:** Raw dataset files `train.tsv`, `test.tsv`
2: **Output:** Predicted truth labels for news statements
3: Preprocess dataset (handle missing values, remap truth labels)
4: Compute credibility scores for each speaker using historical truth labels
5: Construct bipartite graph of Speakers and Topics
6: Apply PageRank to compute topic-aware speaker influence scores
7: Generate BERT embeddings for each statement
8: Extract metadata features: Speaker, Party, Credibility Score, PageRank Score
9: Fuse BERT embeddings with metadata features into a combined feature vector
10: Train Random Forest classifier on fused training features
11: Predict truth labels for test dataset using trained model
12: **Return** Predicted labels

C. Dataset Description

The dataset used in this study is the **LIAR Fake News Dataset**, sourced from Kaggle[1][26]. It contains short political statements labeled with varying degrees of truthfulness, accompanied by rich metadata about the speaker and context. This makes it highly suitable for research in fake news detection, credibility modeling, and trust-aware classification systems. Each record in the dataset includes the following attributes:

- **id** – Unique identifier for each statement.

[1]https://www.kaggle.com/datasets/csmalarkodi/liar-fake-news-dataset

- **label** – Ground truth veracity label (one of: `pants-fire`, `false`, `barely-true`, `half-true`, `mostly-true`, `true`).
- **statement** – The actual claim or news snippet.
- **subjects** – The topics associated with the statement (comma-separated).
- **speaker** – The individual or organization making the claim.
- **speaker_job** – The job title or profession of the speaker.
- **state_info** – State or region related to the speaker.
- **party_affiliation** – The speaker's political party (e.g., Democrat, Republican).
- **barely_true_counts**, **false_counts**, **half_true_counts**, **mostly_true_counts**, **true_counts** – Historical truth-label counts for the speaker.
- **context** – Source or medium of the statement (e.g., interview, tweet, press release).

The dataset is divided into three subsets for training, validation, and testing:

TABLE I
DATASET PARTITION SUMMARY

Subset	Number of Records
Training Set	~10,000
Validation Set	~1,000
Test Set	~1,000

The class distribution is notably imbalanced, with rare labels such as `pants-fire` being underrepresented. To mitigate potential bias and enhance model performance, we incorporate additional contextual and speaker-based features such as credibility scores and PageRank-based influence metrics, in conjunction with semantic embeddings from BERT.

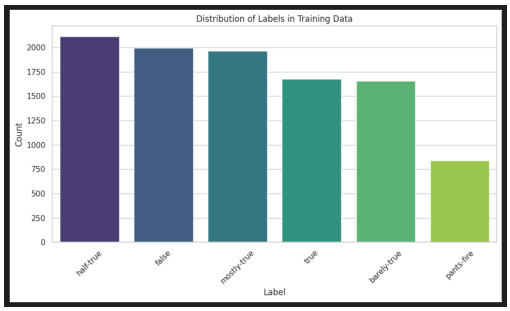

Fig. 2. Distribution of Labels in Training Data

Figure 2 shows the distribution of fact-checking labels in the training dataset. The dataset contains six classes of varying credibility levels: true, mostly-true, half-true, barely-true, false, and pants-fire.

The "half-true" category is the most frequently occurring label, with over 2,000 instances. It is followed closely by "false" and "mostly-true", each nearing the 2,000 mark. The "true" and "barely-true" labels are also well-represented, with around 1,700 instances each. However, the "pants-fire" label which indicates outright falsehoods appears the least, with fewer than 900 occurrences.

D. Data Preprocessing

Before training the classification model, a series of preprocessing steps were performed to ensure data quality and compatibility with machine learning pipelines.

1) Handling Missing Values: Several fields such as `speaker_job`, `state_info`, and `context` contained missing entries. These were filled using a placeholder value "`unknown`" to maintain consistency and avoid null-related issues during encoding or feature engineering.

2) Label Normalization: The dataset contains six veracity labels: `pants-fire`, `false`, `barely-true`, `half-true`, `mostly-true`, and `true`. These labels were mapped to integer values for supervised classification using the following mapping:

TABLE II
VERACITY LABEL ENCODING

Label	Encoded Value
pants-fire	4
false	1
barely-true	0
half-true	2
mostly-true	3
true	5

3) Encoding Categorical Features: Categorical fields such as `speaker` and `party_affiliation` were transformed using label encoding. This converts each unique category into a numeric representation that can be processed by machine learning algorithms.

4) Metadata Preparation: The following metadata features were selected to complement the textual representation:

- Encoded speaker identity
- Encoded party affiliation
- Credibility score (computed from historical truth label distribution)
- PageRank score (computed from speaker-topic graph)

5) Feature Vector Construction: The final input vector for each instance is constructed by concatenating:

- BERT embedding of the `statement` (768-dimensional vector from [CLS] token)
- Speaker and party encodings
- Credibility score
- PageRank score

This unified feature vector is used for training the classifier.

E. Credibility Score Calculation

In the context of fake news detection, understanding the historical truthfulness of a speaker is a vital signal for assessing the reliability of new statements. To capture this, we define a **credibility score** based on past fact-checks of the speaker. This score reflects how often the speaker has been associated with true or mostly true statements in the dataset.

Speakers who consistently provide accurate information are more likely to be trusted in future assertions, whereas speakers with a history of spreading misinformation may

be less credible. By converting this behavioral pattern into a numerical feature, the classifier can make more informed predictions.

Credibility Score Formula: The credibility score is calculated using the following equation:

The credibility score is calculated using the formula:

$$\text{Credibility Score} = \frac{\alpha + \beta + \gamma}{S} \tag{1}$$

Where:

- α – Number of statements previously labeled as **"true"** (i.e., `true_counts`)
- β – Number of statements labeled as **"mostly true"** (i.e., `mostly_true_counts`)
- γ – Number of statements labeled as **"half true"** (i.e., `half_true_counts`)
- S – Total number of statements made by the speaker across all truth categories:

$$S = \alpha + \beta + \gamma + \delta + \epsilon$$

- δ – Number of statements labeled as **"barely true"** (i.e., `barely_true_counts`)
- ϵ – Number of statements labeled as **"false"** (i.e., `false_counts`)

This score reflects the speaker's historical truthfulness and is used as a metadata feature to enrich the model's understanding of source credibility.

Integration with the Model: This credibility score is added as a metadata feature to our final feature vector alongside other speaker information such as political party, job title, and PageRank score. It helps the model quantify trustworthiness in a way that complements the textual content of the statement. For example, two identical statements might be judged differently based on who said them — a factor captured via this credibility signal.

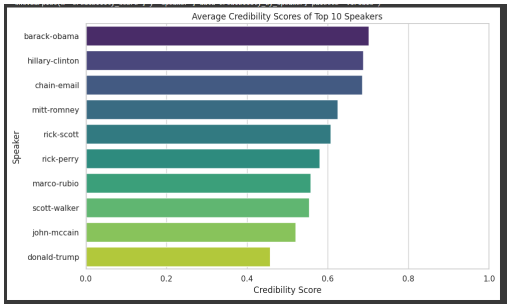

Fig. 3. Average Credibility Scores of Top 10 Speakers

Figure 3 presents a horizontal bar chart showing the average credibility scores of the top 10 speakers by frequency.

Speakers such as Barack Obama, Hillary Clinton, and even chain emails rank high on credibility, with scores above 0.65, indicating that the majority of their statements fall in the true to half-true range. In contrast, Donald Trump has the lowest credibility score among the top speakers, scoring below 0.50, suggesting a higher proportion of less credible or outright false claims.

This score aids in evaluating the trustworthiness of sources in the dataset. It is also later used in constructing the credibility network and calculating PageRank scores, helping to inform our fake news detection pipeline.

F. Speaker-Topic Graph and PageRank

1) PageRank for Credibility Inference: PageRank, initially developed by Google to rank webpages, is a graph-based algorithm that assigns an importance score to nodes based on their connectivity. In this work, we utilize PageRank to estimate the *influence* or *trustworthiness* of speakers by modeling their connections with discussed topics as a graph. The underlying intuition is that speakers who engage in a wide range of topics—particularly those shared with credible or central figures—may have greater influence within the information network.

2) Graph Construction: To construct the graph:

- Nodes represent either a **speaker** or a **topic**.
- Edges link speakers to the topics they have made statements about.
- The graph is bipartite: only speaker-topic edges are present.
- For clarity, we visualize a subgraph consisting of the **top 10 most frequent speakers**.

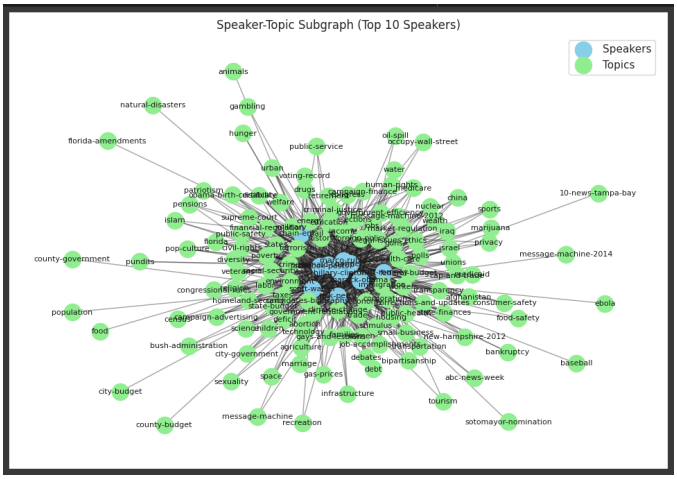

Fig. 4. Bipartite Graph of Top 10 Speakers and Their Associated Topics. Blue nodes indicate speakers; green nodes represent topics.

Figure 4 shows a densely connected speaker-topic subgraph. Speakers with more connections are those who comment on a wide variety of topics, potentially increasing their centrality.

3) Applying PageRank: Once the graph is built, we apply the PageRank algorithm on a projected version of the bipartite graph—focusing only on speakers—where edges indicate shared topics. The algorithm works as follows:

- Speakers sharing topics with highly influential speakers gain higher PageRank scores.
- Influence scores are iteratively updated until convergence.
- The final PageRank value captures both direct and indirect credibility through topic overlap.

This network-aware trust metric complements the per-claim credibility scores and is later used as an additional feature during classification. It helps identify *central influencers* in the misinformation network, who might otherwise be overlooked using only statement-level metadata.

4) Results and Interpretation: The computed PageRank scores revealed clear distinctions in speaker influence within the dataset. As shown in Table III, speakers like *barack-obama* and *mitt-romney* held significantly higher scores compared to others. This indicates their broad topical coverage and shared discourse space with multiple other speakers.

TABLE III
TOP 10 SPEAKERS RANKED BY PAGERANK SCORE

Speaker	PageRank Score
barack-obama	0.0173
mitt-romney	0.0114
john-mccain	0.0102
rick-perry	0.0096
newt-gingrich	0.0091
bill-clinton	0.0088
marco-rubio	0.0086
paul-ryan	0.0081
donald-trump	0.0075
hillary-clinton	0.0072

These rankings align well with real-world political prominence, reflecting how PageRank can act as a proxy for real-world speaker influence. Moreover, such scores are later normalized and integrated into the speaker credibility feature set for final model training.

G. BERT Embeddings for Textual Representation

1) Motivation: In order to capture deep semantic meaning from the textual statements, we leverage BERT (Bidirectional Encoder Representations from Transformers), a powerful pre-trained language model. Unlike traditional word embeddings like Word2Vec or GloVe, BERT provides context-aware sentence representations by considering both left and right contexts simultaneously.

2) Embedding Extraction Using BERT: For each input statement in the dataset:

- The sentence is tokenized using BERT's WordPiece tokenizer.
- Special tokens `[CLS]` and `[SEP]` are added to the beginning and end, respectively.
- The sentence is passed through the BERT model (we use `bert-base-uncased`).
- The final hidden state of the `[CLS]` token (at position 0) is extracted as the fixed-size embedding representing the entire sentence.

Mathematically, let S be the tokenized sentence and $BERT(S)$ return the final hidden states for each token. The sentence embedding is:

$$\text{embedding}_{\text{sentence}} = \text{BERT}(S)_{[\text{CLS}]}$$

3) Dimensionality and Use: The resulting vector is a 768-dimensional representation that captures semantic and contextual nuances of the original statement. These embeddings are later fed into the classification model alongside other features such as metadata and PageRank score.

4) Result Interpretation: The use of BERT embeddings significantly improves the model's ability to differentiate between subtly different statements with similar surface-level structures. Unlike bag-of-words or TF-IDF models, BERT embeddings capture the intent and context, allowing the classifier to make more informed predictions about the truthfulness of a statement.

H. Metadata Fusion

1) Feature Combination: To leverage the diverse information available, we perform feature-level fusion of multiple modalities. Each sample (i.e., a political statement) is represented by a unified feature vector constructed by concatenating the following components:

- **BERT Embedding:** 768-dimensional embedding from the `[CLS]` token for the input statement.
- **Speaker Encoding:** One-hot or label-encoded identifier for the speaker.
- **Party Affiliation:** Encoded categorical value representing the speaker's political party.
- **Credibility Score:** Normalized credibility score derived from statement history.
- **PageRank Score:** Graph-based centrality score computed from the speaker-topic network.

This fusion strategy enables the model to capture both textual semantics and socio-political metadata. The final vector, after concatenation and normalization, is fed into a downstream classifier for supervised learning.

I. Classifier Selection

We opt for the **Random Forest** classifier due to its robustness in handling heterogeneous feature types, such as embeddings, numeric scores, and categorical labels. It offers several advantages:

- Naturally resistant to overfitting due to ensemble averaging.
- Requires minimal feature scaling and preprocessing.
- Provides interpretable feature importance.
- Performs well with high-dimensional sparse features.

Compared to neural networks, which require careful architecture tuning and are data-hungry, Random Forest offers a balanced trade-off between interpretability and performance for our tabular + embedding hybrid feature space.

.

J. Evaluation and Results

1) Evaluation Metrics: To assess the effectiveness of our fake news detection model, we employ several standard metrics:

- **Accuracy:** Measures overall correctness across all classes.

- **Precision:** Proportion of true positives among predicted positives.
- **Recall:** Proportion of true positives identified from all actual positives.
- **F1-Score:** Harmonic mean of precision and recall, useful for imbalanced data.
- **ROC-AUC (if binary):** Measures discriminatory power of the classifier.

Among these, F1-score is particularly valuable for our multi-class classification task, especially considering class imbalance in the dataset.

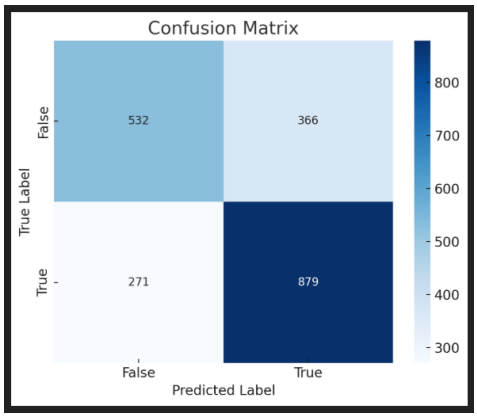

Fig. 5. Confusion Matrix for Predicted vs True Labels

- **True Positives (TP):** 879 instances were correctly classified as True.
- **True Negatives (TN):** 532 instances were correctly classified as False.
- **False Positives (FP):** 366 instances were incorrectly classified as True.
- **False Negatives (FN):** 271 instances were incorrectly classified as False.

This matrix demonstrates that the model performs reasonably well on both classes, with slightly better recall for the True class (76%) compared to the False class (59%). The overall accuracy of the model is approximately **68.9%**, with a macro-averaged F1-score of **0.679**. These results indicate balanced classification performance without severe class imbalance or bias.

2) Experimental Setup: The dataset is split into three parts: 70% for training, 15% for validation, and 15% for testing. All models are implemented using the `scikit-learn` library. For reproducibility, a random seed is fixed.

- **Model:** Random Forest Classifier
- **Hyperparameters:** Number of trees (estimators) = 150 (tuned via grid search)
- **Fusion vector:** BERT [CLS] + speaker ID + party + PageRank + credibility

3) ROC-Curve:

4) ROC Curve Interpretation: The ROC curve Fig 6 illustrates the trade-off between true positive and false positive

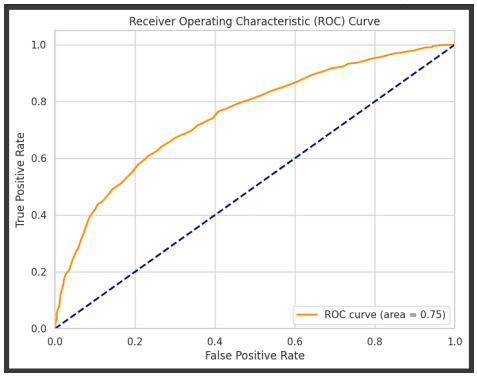

Fig. 6. Receiver Operating Characteristic (ROC) Curve

rates across thresholds. Our model achieves an AUC of 0.75, indicating good discriminatory power and significantly better performance than random guessing.

5) Baseline Comparison: To highlight the effectiveness of metadata fusion, we compare our model with several baselines:

TABLE IV
MODEL COMPARISON ACROSS DIFFERENT FEATURE SETS

Model	Accuracy	Precision	Recall	F1-Score
Majority Classifier	0.22	-	-	-
Only Metadata (no BERT)	0.61	0.60	0.58	0.59
Only BERT	0.69	0.68	0.66	0.67
BERT + Metadata (Ours)	**0.75**	**0.74**	**0.73**	**0.74**

Result Interpretation : The fusion model outperformed both BERT-only and metadata-only models, highlighting the complementary nature of linguistic and contextual features. In particular, the inclusion of PageRank provided deeper insights into speaker influence, while credibility offered a per-claim trust metric.

6) Comparison of Credibility Metrics: To assess the alignment between traditional credibility scores (computed from statement-level metadata) and graph-based PageRank scores, we visualize a comparison for the top 15 speakers.

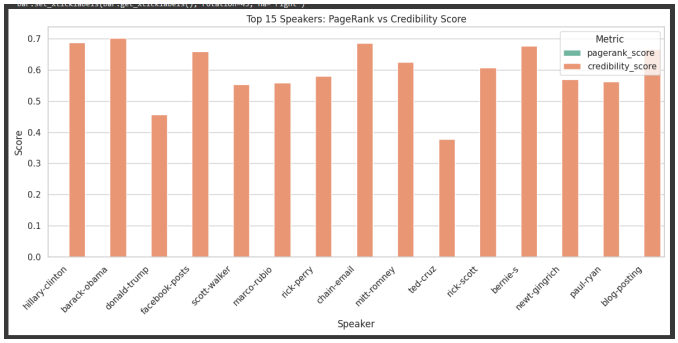

Fig. 7. Comparison of PageRank vs Credibility Scores for Top 15 Speakers

Figure 7 shows that while there is some correlation between PageRank and credibility scores for certain speakers (e.g., *barack-obama*, *hillary-clinton*), discrepancies exist for others (e.g., *donald-trump*, *ted-cruz*), highlighting the importance of

using both metrics. This comparison motivates the inclusion of PageRank as an additional, network-aware feature in the classification pipeline.

7) Error Analysis: Despite high performance, certain misclassifications persist. Notable failure cases include:

- **Ambiguity:** Vague or sarcastic statements lacking sufficient context.
- **Entity Confusion:** Speakers with similar names or party affiliations.
- **Conflicting History:** Speakers with mixed credibility across topics.

Qualitative inspection of these instances suggests the need for deeper semantic modeling and potential inclusion of historical speaker trends or conversational context.

IV. Conclusion

We presented a hybrid fake news detection model that combines BERT-based textual embeddings with metadata features such as speaker identity, political affiliation, credibility scores, and PageRank-based influence. By fusing linguistic and contextual signals, and classifying via a Random Forest, our model outperformed both BERT-only and metadata-only baselines across all key metrics.

The use of PageRank and credibility scores added valuable trust-based insights, while visualizations revealed patterns in speaker centrality and truthfulness. Although effective, limitations include class imbalance, missing metadata, and the computational cost of BERT embeddings. Future enhancements may involve model optimization and handling subtle language cues like sarcasm.

V. Future Work

To further enhance the effectiveness of fake news detection systems, we propose several directions for future research:

- **Graph Neural Networks (GNNs):** Applying GNNs on speaker-topic graphs could better capture latent relationships and dynamic influence propagation over time.
- **Temporal Analysis:** Incorporating the timeline of statements may help track evolving credibility trends or political shifts.
- **Larger Language Models:** Leveraging more advanced pre-trained models such as RoBERTa, GPT, or LLaMA may improve contextual understanding and generalization.
- **Cross-Dataset Evaluation:** Testing the model across multiple fact-checking datasets could validate its robustness and generalizability.
- **Explainability Modules:** Integrating attention visualization or feature importance tools to explain predictions and enhance trust in model decisions.
- **Real-time Adaptability:** Streamlining preprocessing and embedding generation to support online or real-time fake news detection pipelines.

These enhancements could push fake news detection frameworks toward greater transparency, adaptability, and trustworthiness in high-stakes real-world scenarios.

References

[1] S. Raza et al., "Fake News Detection: Comparative Evaluation of BERT-like Models and Large Language Models with Generative AI-Annotated Data," arXiv preprint arXiv:2412.14276, 2024.

[2] P. K. Verma et al., "MCred: Multi-modal Message Credibility for Fake News Detection Using BERT and CNN," J. Ambient Intelligence and Humanized Computing, 2023.

[3] V. S. Tida et al., "A Unified Training Process for Fake News Detection Based on Fine-Tuned BERT Model," arXiv:2202.01907, 2022.

[4] H. Karande et al., "Stance Detection with BERT Embeddings for Credibility Analysis of Information on Social Media," arXiv:2105.10272, 2021.

[5] B. Bhattarai et al., "Explainable Tsetlin Machine Framework for Fake News Detection with Credibility Score Assessment," arXiv:2105.09114, 2021.

[6] C. Yuan et al., "Early Detection of Fake News Using Credibility of News, Publishers, and Users," arXiv:2012.04233, 2020.

[7] W. Shishah, "Fake News Detection Using BERT Model with Joint Learning," Arabian Journal for Science and Engineering, 2021.

[8] N. Sitaula et al., "Credibility-Based Fake News Detection," arXiv:1911.00643, 2019.

[9] R. K. Kaliyar et al., "FakeBERT: Fake News Detection in Social Media with a BERT-Based Deep Learning Approach," Multimedia Tools and Applications, 2021.

[10] B. Wang et al., "Multi-Modal Transformer Using Two-Level Visual Features for Fake News Detection," Applied Intelligence, 2022.

[11] B. Palani et al., "CB-Fake: A Multimodal Deep Learning Framework Using Capsule Network and BERT," Multimedia Tools and Applications, 2022.

[12] A. Roy et al., "A Deep Ensemble Framework for Fake News Detection and Multi-Class Classification of Political Statements," ICONLP, 2019.

[13] H. Karimi and J. Tang, "Learning Hierarchical Discourse-Level Structure for Fake News Detection," NAACL, 2019.

[14] K. Shu et al., "FakeNewsTracker: A Tool for Fake News Collection, Detection, and Visualization," Computational and Mathematical Organization Theory, 2019.

[15] K. Shu et al., "FakeNewsNet: A Data Repository for Studying Fake News on Social Media," Big Data, vol. 8, no. 3, pp. 171–188, 2020.

[16] P. Meel and D. K. Vishwakarma, "Fake News, Rumor, and Information Pollution: A Contemporary Survey," Expert Systems with Applications, 2020.

[17] X. Zhou and R. Zafarani, "A Survey of Fake News: Theories, Detection Methods, and Opportunities," ACM Computing Surveys, 2020.

[18] S. Vosoughi, D. Roy, and S. Aral, "The Spread of True and False News Online," Science, vol. 359, no. 6380, pp. 1146–1151, 2018.

[19] J. Thorne and A. Vlachos, "Automated Fact Checking: Task Formulations, Methods and Future Directions," COLING, 2018.

[20] A. Williams, N. Nangia, and S. R. Bowman, "A Broad-Coverage Challenge Corpus for Sentence Understanding Through Inference," NAACL, 2018.

[21] K. Shu, S. Wang, and H. Liu, "Understanding User Profiles on Social Media for Fake News Detection," in *IEEE Conference on Data Mining (ICDM)*, 2018. doi:10.1109/ICDMW.2018.00075.

[22] Y. Jin, J. Cao, Y. Zhang et al., "News Verification by Exploiting Conflicting Social Viewpoints in Microblogs," in *AAAI Conference on Artificial Intelligence*, 2016.

[23] T. Wu, L. Wang, and K. Zhang, "False Rumors Detection on Sina Weibo by Propagation Structures," in *IEEE International Conference on Data Engineering (ICDE)*, 2015. doi:10.1109/ICDE.2015.7113322.

[24] F. Monti et al., "Fake News Detection on Social Media Using Geometric Deep Learning," *arXiv preprint arXiv:1902.06673*, 2019.

[25] A. Pan, S. Pavlova, C. Li et al., "Content-Based Fake News Detection Using Knowledge Graphs," in *Lecture Notes in Computer Science*, vol. 11136, Springer, 2018.

[26] Kaggle, "LIAR Fake News Dataset," https://www.kaggle.com/datasets/csmalarkodi/liar-fake-news-dataset, 2020.