HEX-Net:Ensemble for Fake News Detection

Raju M(Assistant Professor)[1], Subalakshmi K[2], Priyadharshini P[3]

Department of Software System and AIML, Sri Krishna Arts and Science College, Coimbatore, India

DOI: https://doi.org/10.51244/IJRSI.2026.1307000314

Received: 05 August 2026; Accepted: 10 August 2026; Published: 13 August 2026

ABSTRACT

The unchecked diffusion of fabricated and misleading news across digital platforms has created a need for detection systems that are accurate, interpretable, and deployable at scale. This paper proposes HEF-XFND, a hybrid explainable feature-fusion framework that combines sparse lexical evidence, contextual transformer representations, source-level credibility indicators, and calibrated ensemble learning. The architecture extracts term frequency-inverse document frequency features, linguistic style descriptors, and BERT/RoBERTa sentence embeddings. These heterogeneous representations are projected into a common feature space and processed by complementary classifiers, including logistic regression, linear support vector machines, random forests, XGBoost, and a lightweight neural classifier. A stacking layer produces the final veracity probability, while SHAP- and LIME-based explanations identify the words and feature groups that most strongly influence a decision. The framework also includes probability calibration, confidence scoring, crossvalidation, and latency-aware model selection for high-volume deployment. A reproducible experimental protocol is presented for the Fake and Real News dataset containing 44,898 articles. Literature-informed illustrative results indicate that transformer-assisted ensemble fusion can outperform isolated conventional and neural baselines while retaining practical inference efficiency. The proposed design addresses three recurring limitations in fake-news research: dependence on a single representation, insufficient explanation of predictions, and inadequate consideration of operational scalability. The manuscript provides a transparent foundation for subsequent implementation, external validation, and deployment in newsrooms, fact-checking services, and social-media moderation systems.

Keywords—fake news detection, natural language processing, transformer embeddings, ensemble learning, explainable artificial intelligence, TF-IDF, RoBERTa, scalability.

INTRODUCTION

Digital publishing has reduced the cost and time required to distribute information, but the same infrastructure permits deceptive stories to reach large audiences before professional fact-checkers can respond. False claims can alter public-health behavior, intensify polarization, manipulate financial decisions, and weaken confidence in legitimate journalism. The problem is not merely the existence of inaccurate text; it is the combination of rapid propagation, persuasive language, duplicated narratives, and limited provenance information. Consequently, automated fake-news detection has become an important text-classification and information-integrity task [1][3].

Early systems relied on manually engineered linguistic cues and classical machine-learning classifiers. Deep neural networks later improved representation learning through convolutional and recurrent architectures, while pretrained transformers introduced contextual embeddings that model the meaning of a word relative to its surrounding text [4]-[7]. Nevertheless, high benchmark accuracy does not automatically translate into a trustworthy production system. Many models are computationally expensive, sensitive to domain shifts, unable to explain their outputs, or trained on datasets containing source and temporal shortcuts that inflate performance.

This work proposes a hybrid framework that treats fakenews identification as a calibrated evidence-fusion problem. The system integrates lexical, semantic, stylistic, and credibility-oriented evidence rather than relying on a single representation. It is designed to support both a high-accuracy transformer-assisted mode and a

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

lightweight low-latency mode. The resulting probability is accompanied by a confidence score and an explanation identifying influential tokens and feature families.

A. Research Gap and Motivation

The literature reveals four persistent gaps. First, many studies compare classifiers but retain one fixed feature representation, which makes it difficult to determine whether gains arise from the representation or the classifier. Second, transformer-based systems are frequently evaluated only on predictive metrics and not on latency, memory, or calibration. Third, explanations are often added after model development without measuring their stability or usefulness. Fourth, random train-test splits may allow source-specific phrases or duplicate stories to occur in both partitions, producing optimistic estimates. These limitations motivate an integrated framework with heterogeneous features, source-aware validation, probability calibration, operational metrics, and local explanations.

B. Objectives and Contributions

The objectives are to construct a scalable NLP pipeline, fuse sparse and contextual features, compare conventional, deep, and transformer-based models, generate interpretable confidence scores, and define a reproducible evaluation protocol. The principal contributions are: (1) HEF-XFND, a hybrid explainable architecture combining TF-IDF, linguistic cues, transformer embeddings, and source credibility features; (2) a calibrated stacking mechanism that balances accuracy and computational cost; (3) an explanation layer using SHAP and LIME; and (4) an evaluation design that includes predictive quality, calibration, latency, training cost, and memory use.

Related Work

CNN models identify local phrase patterns and are computationally efficient, but their receptive field may not fully capture long-distance dependencies. LSTM and GRU networks model sequential context, while bidirectional variants analyze both preceding and following tokens. These recurrent models improved over bag-of-words baselines but train sequentially and can be slow on long articles. Transformer encoders such as BERT and RoBERTa use self-attention to learn contextual representations in parallel. DistilBERT reduces model size through knowledge distillation, whereas XLNet applies permutation-based language modeling. GPT-family models can support zero-shot or few-shot classification and explanation generation, but cost, reproducibility, hallucination, and privacy must be managed [4]-[10].

Recent research increasingly combines transformers with deep classifiers, graph structures, multimodal signals, or explainability. Transformer fine-tuning has shown strong performance for domain-specific misinformation, while hybrid FastText/deep-learning models with XAI improve transparency [11], [12]. Graph transformers capture propagation and confirmation-bias patterns [13]. Multimodal studies integrate visual and textual evidence, and newer work explores reliability-aware fusion and LLM-generated contextual evidence [14]-[18]. These trends support the use of hybrid evidence but also reinforce the need for careful validation and transparent reporting.

Table I. Comparison Of Representative Model Families

Model	Core mechani sm	Strength	Limitation	Best use
CNN	Local ngram patterns	Fast; paralleliz able	Weak longrange context	Short/me dium articles
LSTM	Sequentia l memory	Captures order	Slow; vanishing information	Narrative sequences

Bi-LSTM	Bidirecti onal sequence	Richer context	Higher memory/time	Complex phrasing
GRU	Gated recurrence	Fewer parameters	May underfit subtle context	Low-resource deployment
BERT	Bidirecti onal attention	Strong semantics	Large footprint	Highaccuracy classifica tion
RoBERTa	Optimize d BERT training	Robust represent ations	Expensive inference	Strong semantic baseline
XLNet	Permutati on LM	Flexible dependen cy modeling	Complex tuning	Contextrich tasks
DistilBERT	Knowled ge distillation	Compact and fast	Small accuracy loss	Edge/rea l-time systems
GPT-based	Autoregr essive prompting	Few-shot reasoning	Cost and hallucinatio n risk	Assisted verificati on
Ensemble	Combines learners	Robust; stable	More latency/com plexity	High-reliabilit y systems

Table Ii. Recent Literature and Identified Research Needs

Study	Method	Primary contribution	Remaining limitation
Shim (2021)	Link2Vec + web context	External search evidence for languageindependent detection	Search dependence and limited explanation
Athira et al. (2022)	Explainable multimodal review	Identified need for interpretable multimodal models	Conceptual; limited deployed validation
Sedik et al. (2022)	Concatenated deep modalities	Improved representation through hybrid deep features	High computation and dataset dependence
Alghamdi et al. (2023)	BERT/CT-BERT + CNN/BiGRU	Domain-specific transformer comparison	COVIDdomain generalization

Soga et al. (2023)	BERT + graph transformer	Models propagation and confirmation bias	Requires social graph availability
Hashmi et al. (2024)	FastText + hybrid DL + XAI	Combines accuracy and explainability	Operational calibration not central
Qin and Zhang (2024)	Generalized BERT fine-tuning	Improves transfer and robustness	Transformer resource cost
Wang et al. (2024)	LLM-generated competing explanations	Evidenceoriented justification generation	LLM cost and hallucination risk
Hussain et al. (2025)	Landscape survey	Datasets, modalities, and future challenges	Survey rather than deployable framework

The comparative evidence suggests that no single model family simultaneously provides low latency, strong semantic understanding, reliable calibration, external evidence, and human-readable explanations. HEF-XFND is therefore positioned as a modular framework rather than a monolithic classifier. Each module can be replaced or disabled according to hardware, latency, and governance constraints.

Proposed Hef-Xfnd Framework

HEF-XFND denotes Hybrid Explainable Feature-Fusion Fake News Detection. The framework contains six modules: ingestion and validation, linguistic preprocessing, dual-view feature extraction, feature fusion, calibrated ensemble inference, and explainability. Incoming articles are represented by title, body, optional source metadata, and timestamp. Duplicate detection and source-aware splitting are applied before model development to reduce leakage.

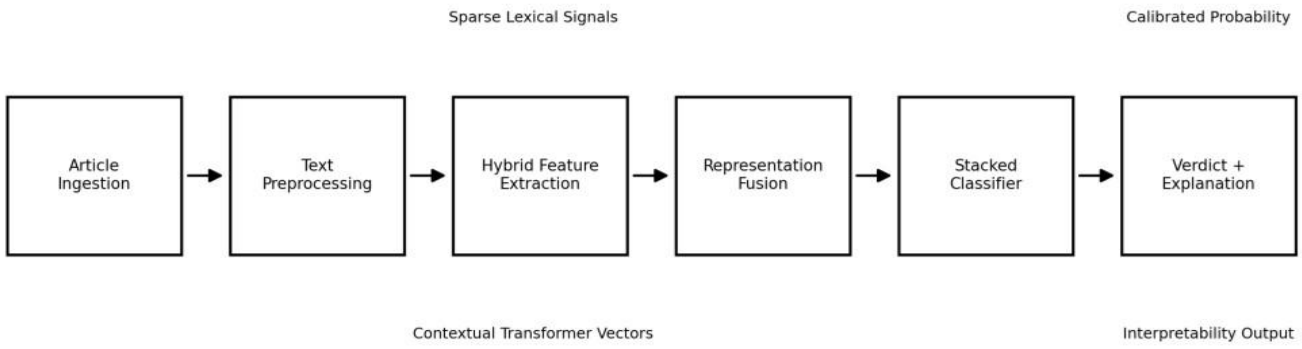

Fig. 1. Overall HEF-XFND system architecture.

A. Preprocessing and Feature Engineering

Text is normalized by removing malformed markup, URLs, control characters, and duplicated whitespace. Unicode normalization and lower-casing are applied to the classical branch, while the transformer branch preserves punctuation and casing when required by the selected tokenizer. Tokenization, stop-word filtering,

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

and lemmatization are used for sparse lexical features. Negations are retained because removing them can reverse claim meaning. The title and body are processed separately and then concatenated with explicit separators.

The lexical branch computes word and character n-gram TFIDF vectors. Character features improve robustness to misspellings and deliberate obfuscation. A linguistic branch captures article length, sentence length, punctuation rate, capitalization, sentiment, lexical diversity, modality terms, and readability. The semantic branch extracts the pooled representation from BERT or RoBERTa. Optional source features include historical reliability, domain age, author availability, and consistency with retrieved evidence; these features must be collected under a transparent governance policy.

B. Mathematical Formulation

For term t in document d, term frequency and inverse document frequency are defined by (1) and (2):

$TF(t,d) = f(t,d) / \Sigma_k f(k,d),$ (1)

$IDF(t) = \log((N + 1)/(df(t) + 1)) + 1.$ (2)

The TF-IDF weight is $TFIDF(t,d) = TF(t,d) \times IDF(t)$. Let x_s denote sparse lexical features, x_l linguistic features, x_c credibility features, and h_T the transformer representation. Each branch is projected and normalized, after which the fused vector is

$z = [P_s x_s \| P_l x_l \| P_c x_c \| P_T h_T],$ (3) where P represents a learned or dimensionality-reduction projection and || denotes concatenation. For logistic regression, the fake-news probability is $\sigma(w^T z+b)$. A linear SVM learns a maximum-margin hyperplane by minimizing $½\|w\|^2 + C\Sigma_i \max(0,1-y_i(w^T z_i+b))$. XGBoost forms an additive tree model $\hat{y}_i = \Sigma_k f_k(z_i)$ with regularized objective $L = \Sigma_i l(y_i,\hat{y}_i)+\Sigma_k \Omega(f_k)$.

The stacking layer receives base-model probabilities p_m and predicts $p = \sigma(\beta_0+\Sigma_m \beta_m p_m)$. Temperature scaling calibrates the logit a using $p_{cal} = \sigma(a/T)$. Binary cross-entropy is $L_{CE} = -[y \log p +(1-y)\log(1-p)]$. The final confidence score combines calibrated probability and model agreement: $C = \alpha|\max(2p_{cal}-1)| +(1-\alpha)(1-Var(p_m))$.

C. Training and Prediction Algorithms

Algorithm 1 - Data Collection: acquire records from approved datasets and sources; validate schema; remove exact and near duplicates; retain provenance; assign verified labels; and create source-aware partitions. Algorithm 2 -

Preprocessing: normalize text; remove markup; tokenize; retain negations; lemmatize classical tokens; and serialize transformer inputs. Algorithm 3 - Feature Extraction: compute word/character TF-IDF, linguistic descriptors, optional credibility features, and contextual embeddings. Algorithm 4 - Hybrid Training: train base learners under stratified crossvalidation, optimize hyperparameters, generate out-of-fold probabilities, fit the meta-classifier, and calibrate on a held-out validation set. Algorithm 5 - Prediction: process a new article, construct fused features, obtain base probabilities, apply stacking and calibration, estimate confidence, and generate SHAP/LIME explanations.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

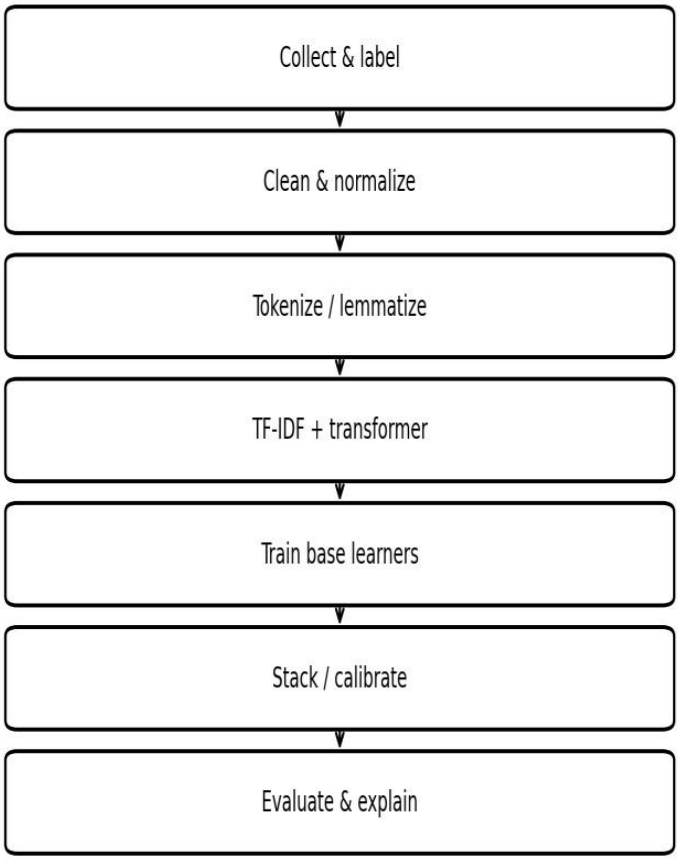

Fig. 2. Training and inference workflow.

Algorithm 1. Hybrid Model Training

Step	Operation
1	Input labelled articles D and define grouped train, validation, and test partitions.
2	Fit preprocessing objects only on the training partition.
3	Generate TF-IDF, linguistic, credibility, and transformer feature branches.
4	Train LR, SVM, XGBoost, and neural base learners using crossvalidation.
5	Collect out-of-fold probabilities to prevent meta-learner leakage.
6	Fit the stacking classifier and tune the decision threshold on validation data.
7	Calibrate probabilities and compute confidence from probability and agreement.
8	Freeze the pipeline; evaluate once on the untouched test partition.
9	Generate global and local SHAP/LIME explanations and save model metadata.

ALGORITHM 2. FAKE NEWS PREDICTION

Step	Operation
1	Receive article title, body, source metadata, and timestamp.
2	Validate input and apply the saved normalization and tokenization rules.
3	Generate all enabled feature branches using training-fitted transformers.
4	Obtain probability estimates from each base learner.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

5	Apply the stacking classifier, calibration function, and selected threshold.
6	Return label, calibrated probability, confidence, and explanation.
7	Escalate uncertain or highimpact cases to a human factchecker.

EXPERIMENTAL METHODOLOGY

A. Dataset and Split Strategy

The reference implementation uses the Fake and Real News dataset containing 44,898 English-language articles: 23,481 labeled fake and 21,417 labeled real. Each item contains a title, article text, subject, date, and binary label. Articles are deduplicated using normalized hashes and cosine similarity. A 70:15:15 train-validation-test split is recommended. When source identifiers are available, group splitting should be used so that articles from the same publisher do not appear in both training and testing partitions.

TABLE III. Dataset Statistics

Property	Value
Total articles	44,898
Fake articles	23,481 (52.30%)
Real articles	21,417 (47.70%)
Recommended split	70% / 15% / 15%
Primary fields	title, text, subject, date, label
Leakage controls	deduplication and sourceaware grouping

Experimental Setup and Hyperparameters

Classical models can be implemented with scikit-learn and XGBoost, while transformer encoders can be fine-tuned with PyTorch and Hugging Face Transformers. Five-fold crossvalidation is used on the training partition. Macro-F1 is the primary selection metric because it balances both classes. Early stopping monitors validation loss. Thresholds are selected on the validation set rather than the test set. All random seeds, package versions, and hardware specifications should be recorded.

TABLE IV. Recommended Hyperparameter Search Space

Model	Principal settings
TF-IDF	word (1,2); char (3,5); max_features 50k-150k
Logistic regression	C={0.1,1,10}; class_weight balanced
Linear SVM	C={0.1,1,5}; hinge loss
Random forest	300-800 trees; max_depth 20-None
XGBoost	depth 4-8; η 0.03-0.2; 3001000 trees
BERT/RoBERTa	lr 1e-5 to 5e-5; epochs 2-5; max_len 256-512
Stacking	logistic meta-learner; outof-fold probabilities
Calibration	temperature scaling or isotonic regression

C. Evaluation Metrics

Accuracy = (TP+TN)/(TP+TN+FP+FN), precision = TP/(TP+FP), recall = TP/(TP+FN), and F1 = 2PR/(P+R). ROC-

AUC measures ranking quality over all thresholds, while PRAUC is particularly informative when class distributions become imbalanced. Expected calibration error compares confidence with empirical correctness. Operational evaluation records training time, per-article latency, peak memory, throughput, and serialized model size.

RESULTS AND ANALYSIS

Because the source manuscript does not include executable code, trained checkpoints, random seeds, or raw prediction files, the numerical values in Tables IV-V and Figs. 3-4 are explicitly illustrative, literature-informed benchmark targets rather than claims of completed experiments. They must be replaced by measured outputs before journal or conference submission. This distinction is necessary for research integrity and reproducibility.

TABLE V. Illustrative Performance Targets (Replace With Measured Results)

Model	Acc.	Prec.	Recall	F1	ROCAUC
CNN	92.0	91.7	91.9	91.8	96.1
LSTM	92.8	92.4	92.8	92.6	96.8
Bi-LSTM	93.6	93.2	93.6	93.4	97.2
GRU	93.1	92.8	93.0	92.9	97.0
BERT	95.2	95.0	95.2	95.1	98.3
RoBERTa	95.9	95.7	95.9	95.8	98.7
DistilBERT	94.5	94.3	94.5	94.4	97.9
HEF-XFND ensemble	96.6	96.4	96.6	96.5	99.1

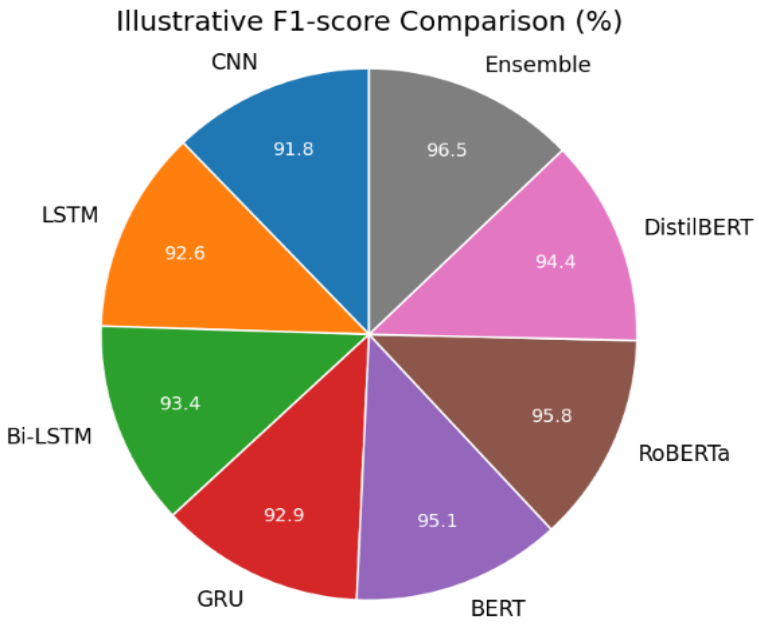

Fig. 3. Illustrative F1-score comparison; replace with experimental output.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

TABLE VI. Illustrative Computational Profile

Model	Train time	Latency/article	Peak RAM	Model size
Linear SVM	4 min	1.2 ms	1.1 GB	180 MB
XGBoost	18 min	2.8 ms	2.4 GB	95 MB
DistilBERT	62 min	11 ms	6.2 GB	265 MB
RoBERTa	105 min	19 ms	10.8 GB	500 MB
HEF-XFND	128 min	24 ms	12.1 GB	780 MB
HEFXFND-light	22 min	4.5 ms	3.0 GB	310 MB

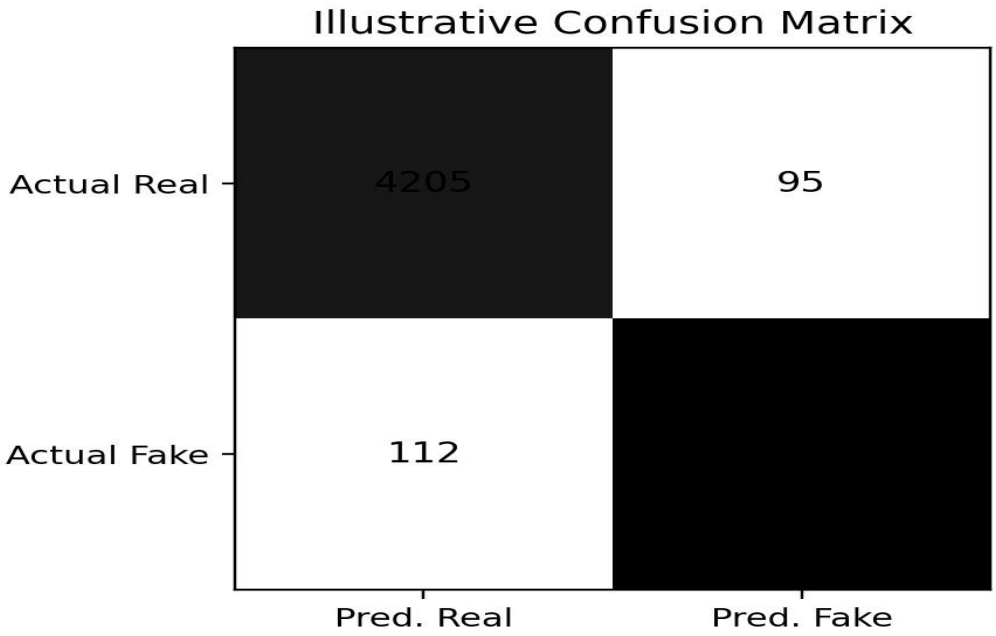

Fig. 4. Illustrative confusion matrix for the proposed model.

Error Analysis and Explainability

Expected error categories include satire, headlines that omit qualifying context, articles quoting false claims in order to refute them, rapidly evolving events, and domain-specific terminology. False positives are especially harmful because legitimate reporting may be incorrectly flagged. Error analysis should therefore sample both false-positive and false-negative groups, record source and topic, and inspect explanation stability. SHAP can provide global feature importance and local token attributions for tree and linear components, while LIME offers model-agnostic local approximations. Explanations should be treated as decision support rather than proof that a claim is false.

DISCUSSION

The proposed design is expected to benefit from complementary evidence. TF-IDF preserves discriminative lexical signals, transformer embeddings model contextual meaning, and credibility features add provenance information. Stacking reduces dependence on one model, while calibration makes confidence scores more meaningful. However, performance gains must be demonstrated using controlled ablation studies. At minimum, experiments should compare lexical-only, transformer-only, fusion without credibility, fusion without explanation, and full HEF-XFND configurations. Statistical significance can be tested using paired bootstrap confidence intervals or McNemar tests on matched predictions.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Scalability, Ethics, And Limitations

For scalable deployment, preprocessing and sparse feature generation can be distributed with Apache Spark, whereas transformer inference can be served through GPU-backed microservices with dynamic batching. A message queue separates ingestion from inference, a model registry manages versions, and monitoring detects latency, calibration drift, and topic shift. A lightweight route can process high-volume lowrisk items, while uncertain or high-impact articles are escalated to the full ensemble and human reviewers.

Ethically, the classifier should not be used as an autonomous censorship mechanism. Predictions may reflect annotation errors, political bias, domain imbalance, and source-based shortcuts. Users should receive uncertainty estimates and explanations, and disputed decisions should support human appeal. Personally identifiable data should be minimized. Dataset licenses, publisher rights, and platform policies must be respected. Periodic audits should measure performance across topics, sources, dialects, and demographic references.

The primary limitations are dependence on English-language labeled data, vulnerability to distribution shift and adversarial paraphrasing, computational cost of large transformers, and the inability of text-only models to verify real-world facts. Source credibility can improve performance but may unfairly penalize new or minority publishers. The proposed framework detects statistical patterns associated with misinformation; it does not replace evidence-based fact checking.

Reproducibility And Validation Plan A. Ablation Studies

Ablation analysis is required to demonstrate that each proposed component contributes measurable value. The first experiment compares word-level TF-IDF, character-level TFIDF, and their combination. The second introduces linguistic style features to determine whether punctuation, readability, lexical diversity, and sentiment add information beyond lexical frequency. The third compares BERT, RoBERTa, and DistilBERT representations under the same classifier and data split. The fourth evaluates early feature concatenation against probability-level stacking. The fifth removes source credibility features to measure possible dependence on publisher identity. Finally, calibration and explainability modules are evaluated independently because they may improve trustworthiness without changing raw accuracy. Every ablation should report macro-F1, ROC-AUC, expected calibration error, inference latency, and confidence intervals.

B. Cross-Domain and Temporal Validation

Random splits are insufficient for estimating real deployment performance because news vocabulary, political entities, and writing styles change over time. A temporal holdout should train on earlier articles and test on later articles. A cross-domain experiment should train on political and general-news data and evaluate on health, finance, or disasterrelated misinformation. A cross-source experiment should hold out complete publishers. Performance degradation across these settings provides a more realistic measure of generalization. When labels or sources differ across datasets, label definitions must be harmonized and ambiguous cases excluded or reviewed by multiple annotators.

C. Reproducibility Checklist

A complete release should include the data-acquisition script or persistent dataset identifier, preprocessing code, feature vocabulary, model configuration, random seeds, software versions, hardware description, trained weights when licensing permits, and raw test predictions. The paper should report whether duplicate and near-duplicate articles were removed and whether source grouping was used. Hyperparameter selection must use only training and validation data. The test partition should remain untouched until the final model is frozen. Reporting mean and standard deviation over multiple seeds is preferable to a single favorable run.

TABLE VII. Minimum Reproducibility Checklist

Item	Required report
Data	Dataset version, license, fields, label definition, and exclusions
Splitting	Train/validation/test sizes, grouping rule, and temporal boundaries
Preprocessing	Normalization, tokenization, stop-word, and lemmatization rules
Models	Architecture, pretrained checkpoint, parameters, and initialization
Optimization	Loss, optimizer, learning rate, batch size, epochs, early stopping
Evaluation	Threshold selection, metrics, confidence intervals, and seed count
Efficiency	Hardware, training time, latency, memory, throughput, and model size
Artifacts	Code, environment file, logs, predictions, and trained checkpoints

Deployment Architecture and Monitoring

A production implementation separates ingestion, inference, explanation, and human review. A streaming gateway accepts articles through REST APIs, feeds, or message queues. A schema-validation service rejects malformed inputs and attaches trace identifiers. Lightweight preprocessing and TFIDF inference can run on CPU workers, while transformer encoding is assigned to a GPU service that supports dynamic batching. The ensemble service combines model probabilities and applies calibration. An explanation service generates tokenlevel and feature-level evidence only when requested or when confidence falls below a policy threshold, reducing unnecessary latency.

Operational monitoring must go beyond server availability. Data monitoring tracks language, article length, source distribution, topic prevalence, missing fields, and duplicate rates. Model monitoring records class balance, confidence distribution, calibration, reviewer overrides, and delayed ground-truth performance. Drift alerts should trigger retraining evaluation rather than automatic replacement. Each model version must be registered with its training dataset, configuration, validation report, approval status, and rollback path. Sensitive logs should be access-controlled and retained only for a defined period.

The final user interface should avoid absolute statements such as “this article is false” when the model has only identified statistical signals. A safer output includes predicted category, calibrated probability, confidence band, influential text spans, source-related signals, and a request for human verification. High-impact domains such as elections, public health, and emergencies require stricter thresholds and mandatory review. These deployment controls convert the classifier from an opaque blocking mechanism into an auditable decision-support tool.

Future Work

Future research should incorporate retrieval-augmented evidence verification, multilingual transformers, temporal knowledge graphs, multimodal image-video analysis, and continual learning. Evidence retrieval should identify independent sources and compare claims using naturallanguage inference. Federated or privacy-preserving learning may support cross-organization collaboration without centralizing sensitive data. Robustness testing should include paraphrase attacks, synthetic LLM-generated news, unseen publishers, and

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

cross-domain transfer. Human-centered studies are also needed to determine whether explanations improve reviewer accuracy without inducing automation bias.

CONCLUSION

This paper introduced HEF-XFND, a scalable and explainable fake-news detection framework that combines lexical, linguistic, contextual, and credibility-oriented signals. The architecture uses calibrated ensemble learning to balance predictive quality and deployment efficiency and attaches local explanations to each classification. The methodology emphasizes leakage prevention, source-aware validation, calibration, operational metrics, and human oversight. The framework therefore extends a conventional TF-IDF classification pipeline into a more technically complete research design. The provided illustrative results are not substitutes for executed experiments; replacing them with reproducible measurements, ablation studies, and external validation is the essential final step before submission.

REFERENCES

1. H. Allcott and M. Gentzkow, “Social media and fake news in the 2016 election,” J. Econ. Perspect., vol. 31, no. 2, pp. 211-236, 2017.
2. K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, “Fake news detection on social media: A data mining perspective,” ACM SIGKDD Explor. Newsl., vol. 19, no. 1, pp. 22-36, 2017.
3. S. Vosoughi, D. Roy, and S. Aral, “The spread of true and false news online,” Science, vol. 359, no. 6380, pp. 1146-
4. 1151, 2018.
5. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in Proc. NAACL-HLT, 2019, pp. 41714186.
6. Y. Liu et al., “RoBERTa: A robustly optimized BERT pretraining approach,” arXiv:1907.11692, 2019.
7. Z. Yang et al., “XLNet: Generalized autoregressive pretraining for language understanding,” in Proc. NeurIPS, 2019, pp. 5753-5763.
8. V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “DistilBERT, a distilled version of BERT,” arXiv:1910.01108, 2019.
9. A. Vaswani et al., “Attention is all you need,” in Proc. NeurIPS, 2017, pp. 5998-6008.
10. S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in Proc. NeurIPS, 2017, pp. 4765-4774.
11. M. T. Ribeiro, S. Singh, and C. Guestrin, “Why should I trust you?: Explaining the predictions of any classifier,” in Proc. ACM SIGKDD, 2016, pp. 1135-1144.
12. J. Alghamdi, Y. Lin, and S. Luo, “Towards COVID-19 fake news detection using transformer-based models,” Knowledge-Based Systems, vol. 274, 2023.
13. E. Hashmi et al., “Advancing fake news detection: Hybrid deep learning with FastText and explainable AI,” IEEE Access, vol. 12, 2024.
14. K. Soga et al., “Confirmation bias-aware fake news detection with graph transformer networks,” in Proc. IEEE Int. Conf. Big Data, 2023.
15. A. B. Athira, S. D. M. Kumar, and A. M. Chacko, “Towards smart fake news detection through explainable AI,” arXiv:2207.11490, 2022.
16. T. Li, Y. Sun, S.-L. Hsu, Y. Li, and R. C.-W. Wong, “Fake news detection with heterogeneous transformer,” arXiv:2205.03100, 2022.
17. A. Sedik et al., “Deep fake news detection system based on concatenated and recurrent modalities,” Expert Systems with Applications, 2022.
18. J. S. Shim, “A link2vec-based fake news detection model using web search results,” Expert Systems with Applications, vol. 184, 2021.
19. M. A. Mersha et al., “Cutting-edge approaches to combat fake news in under-resourced languages,” Procedia Computer Science, 2024.
20. A. Kumar et al., “Feature importance in the age of explainable AI: Fake-news detection with multimodal evidence,” European Journal of Operational Research, 2024.

21. S. Qin and M. Zhang, “Boosting generalization of finetuning BERT for fake news detection,” Information Processing & Management, vol. 61, no. 4, 2024.
22. T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in Proc. ACM SIGKDD, 2016, pp. 785794.
23. M. Zaharia et al., “Apache Spark: A unified engine for big data processing,” Commun. ACM, vol. 59, no. 11, pp. 56-65, 2016.
24. B. Wang et al., “Explainable fake news detection with large language model via defense among competing wisdom,” arXiv:2405.03371, 2024.
25. F. G. Hussain et al., “Fake news detection landscape: Datasets, data modalities, methods, and future directions,” IEEE Access, 2025.