IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

NewsZ: An Explainable Hybrid AI System for Fake News Detection Using Classical Models, LSTM, Fact-Checking, and Web Content Extraction

Mr.R.Senthilkumar, V A Rishivaradha, U Priyadharshini, K pugazhandi,

Department of Computer Science and Engineering Sri Venkateswara College of Engineering

Sriperumbudur, Tamil Nadu, India

Doi: https://doi.org/10.56975/ijcrt.v14i7.309055

Abstract

Online platforms now distribute news at a pace that far exceeds the capacity of manual verification. Fake news written in a formal, journalistic register consistently evades single-model classifiers and leaves users without any explanation for an automated verdict. This paper presents NewsZ, a hybrid explainable AI platform for multi-source fake news detection that unifies four complementary techniques: an ensemble of classical machine learning classifiers (Naive Bayes, Logistic Regression, Support Vector Machine, Random Forest), a two-layer Long Short-Term Memory (LSTM) network, claim-level fact-checking against indexed fact-check databases, and a SHAP-based explainability module that produces keyword evidence and natural-language narratives. A confidence-gated decision layer emits three possible verdicts— REAL, FAKE, or UNCERTAIN— suppressing

overconfident outputs when model agreement falls below a calibrated threshold. The system accepts raw text, URLs, RSS feeds, and PDF documents as input, and surfaces results through a React-based user dashboard and an admin analytics panel. Experiments on a balanced benchmark corpus of 44,898 labelled articles demonstrate 92.4% hybrid accuracy and a false-negative rate of 4.1%, outperforming all single-model baselines and improving user trust significantly compared to opaque classifiers in a controlled user study.

Keywords—Fake News Detection; Hybrid AI; LSTM; Explainability; SHAP; Fact-Checking; Web Content Extraction; Uncertainty Quantification; Ensemble Learning; Misinformation

I. Introduction

The proliferation of fake news across social media, messaging platforms, and news aggregators poses measurable societal, political, and economic risks. News spreads faster than traditional fact-checking institutions can respond, and the average reader cannot reliably distinguish professionally written misinformation from legitimate reporting. Automated detection systems offer a scalable complement to human fact-checking, but deployed systems routinely exhibit two critical deficiencies that limit their operational utility.

First, single-model architectures — whether shallow classifiers trained on bag-of-words features or deep transformer models fine-tuned on labelled corpora— lack robustness when adversarial content deliberately mimics credible journalistic

prose. Ensemble strategies offer partial mitigation, but combining classifiers that share the same feature representation still leaves systematic blind spots. Second, black-box predictions erode user trust: a system that returns a FAKE label without evidence is indistinguishable from a biased filter. Journalists, fact-checkers, and platform moderators need to understand why a verdict was reached before they act on it.

This paper introduces NewsZ, a production-grade platform that addresses both limitations simultaneously. NewsZ unites four paradigms: (1) an ensemble of classical machine learning classifiers operating on TF-IDF and linguistic feature vectors; (2) an LSTM sequence model that captures long-range contextual dependencies missed by bag-of-words representations; (3) a claim-verification component that cross-references extracted named entities and factual claims against indexed fact-check databases; and (4) an explainability layer combining SHAP feature attributions and rule-based narrative generation.

Beyond model fusion, NewsZ introduces a confidence-gated decision layer that withholds a binary verdict when weighted model agreement falls below a calibrated threshold, returning UNCERTAIN instead. This mechanism reduces the most damaging failure mode — fake news incorrectly certified as genuine — at the cost of a bounded abstention rate. The complete system exposes a multi-source input pipeline accepting raw text, URLs, RSS feeds, and PDF documents, and is packaged with a React user dashboard and an admin analytics panel.

The principal contributions of this work are: (1) a hybrid prediction engine with selectable strategy modes (ensemble-only, LSTM-only, hybrid); (2) uncertainty-aware three-class classification with calibrated thresholds; (3) a multi-source input pipeline with graceful fallback for inaccessible URLs and malformed documents; (4) human-readable model explanations with SHAP attribution and keyword salience charts; and (5) integrated user-facing and administrative operational dashboards.

The remainder of this paper is organised as follows. Section II surveys related work. Section III states the problem formally. Section IV describes the proposed system architecture. Section V details the methodology. Section VI covers implementation. Section VII reports experimental results. Section VIII discusses findings. Section IX concludes.

II. Related Work

A. Classical Machine Learning Approaches

Early automated fake news detection relied on hand-crafted linguistic features coupled with shallow classifiers. Castillo et al. [1] demonstrated that credibility signals derived from tweet metadata and writing style achieve acceptable precision in rumour identification on social media. Potthast et al. [2] showed that stylometric features — writing quality, hyperpartisan tone, and punctuation density — discriminate hyperpartisan from mainstream news at high recall. Naive Bayes and Logistic Regression remain competitive baselines owing to their interpretability and low computational cost, yet they fail consistently when adversarial text deliberately imitates authoritative register.

B. Deep Learning and Sequence Models

Recurrent architectures, particularly Long Short-Term Memory networks [3], address the sequential dependency limitations of bag-of-words models by encoding contextual relationships across token sequences. Wang [4] proposed the LIAR benchmark and demonstrated that sequence context significantly improves detection of half-truths and nuanced misinformation compared to n-gram baselines. Transformer-based models, notably BERT [5] and its derivatives, subsequently achieved state-of-the-art performance on several fake news benchmarks. However, these models require substantial fine-tuning infrastructure, carry large memory footprints, and offer limited out-of-the-box interpretability — factors that restrict their deployment in resource-constrained or compliance-sensitive editorial settings.

C. Explainability in Misinformation Detection

The interpretability literature distinguishes post-hoc local explanations — LIME [6] and SHAP [7] — from inherently interpretable model families. Nguyen et al. [8] integrated attention visualisations into a graph-based fake news detector to highlight claim-level evidence. Popat et al. [9] grounded predictions in external fact-check articles retrieved from authoritative sources and presented the evidence to users alongside the verdict. NewsZ synthesises both directions: SHAP TreeExplainer values are computed over Random Forest feature vectors to identify lexical signals, while claim verification grounds predictions in retrieved fact-check evidence, offering users a layered, auditable explanation.

D. Gaps in Existing Systems

Despite the advances surveyed above, deployed fake news detection systems exhibit four persistent gaps that limit operational utility. First, generalisation degrades when misinformation is written in formal, professionally worded prose. Second, no

publicly available system offers a robust multi-source input pipeline that uniformly handles raw text, live URLs, RSS feeds, and PDF documents. Third, existing systems seldom provide uncertainty quantification, producing overconfident incorrect verdicts on borderline articles. Fourth, user-facing explanations are rare and rarely actionable. NewsZ directly targets all four gaps within a unified platform.

III. Problem Statement

Let $A = \{a_1, a_2, ..., a_N\}$ be a collection of N news articles sourced from heterogeneous channels— web pages, RSS feeds, PDF documents, and direct text input. For each article a_i, let t_i denote its extracted and cleaned textual content. The detection objective is to learn a function $f: T \rightarrow \{REAL, FAKE, UNCERTAIN\}$ that assigns a calibrated verdict to each article, where UNCERTAIN is emitted when model confidence falls below an operational threshold.

Four challenges characterise the fake news detection problem that existing systems fail to address effectively. **Formal**

Misinformation. Articles crafted to resemble authoritative journalism— measured tone, correct grammar, plausible citations— evade classifiers trained primarily on sensationalist fake news corpora. False-negative rates on this subcategory are consistently higher than overall benchmark figures suggest.

Multi-Source Input Heterogeneity. News reaches audiences through multiple channels. A detection system that accepts only plain text is operationally incomplete; one that fails silently on malformed URLs or bot-protected pages creates gaps in coverage.

Overconfident Classification. When ensemble members disagree significantly, forcing a binary verdict amplifies the risk of systematic errors. A three-class output— including an explicit UNCERTAIN category— enables downstream triage rather than propagating a wrong label.

Absence of Actionable Explanation. Editorial and moderation workflows require auditable justifications. A verdict unaccompanied by evidence cannot be acted on responsibly, and cannot support appeal or review processes.

IV. Proposed System Architecture

A. Architectural Overview

NewsZ is structured as eight functional layers that process a user submission from raw input to a fully explained verdict. The Input Acquisition Layer accepts submissions through a REST API and routes them by type. The Content Extraction Layer resolves URLs, RSS feeds, and PDFs to clean text. The Preprocessing Layer normalises, validates, and language-checks the text. The Feature Engineering Layer constructs TF-IDF, linguistic, and pattern-based feature vectors. The Prediction Engine runs one of three configurable strategy modes. The Decision Logic Layer applies confidence gating and emits the three-class verdict. The Explainability Layer produces SHAP attributions, keyword evidence, and a plain-language narrative. The Persistence Layer stores results for history retrieval and admin analytics.

TABLE I. NewsZ System Layers

Layer	Purpose	Key Components
Input Acquisition	Accept text / URL / RSS / PDF	Input Router, Type Detector
Content Extraction	Fetch & clean web/RSS/PDF content	URL Fetcher, RSS Parser, PDF Extractor
Preprocessing	Normalise, validate, language-check	Text Cleaner, Language Detector
Feature Engineering	Build TF-IDF and linguistic features	Vectoriser, Feature Builder
Prediction Engine	Multi-model probabilistic inference	Ensemble, LSTM, Hybrid Selector
Decision Logic	Weighted vote with confidence gating	Decision Module, Threshold Gate
Explainability	SHAP, keyword evidence, narrative	Explainer, Chart Builder
Persistence	Store and retrieve analysis history	DB Manager, History Store

B. Input Resolution and Content Extraction

The Input Resolution Module inspects each submission and routes it to the appropriate extractor. URL submissions trigger an HTTP GET with browser-realistic headers; gzip and deflate encodings are handled transparently. HTML is stripped of navigation, advertisement, and footer boilerplate using DOM heuristics in BeautifulSoup before the main article body is extracted. RSS feed submissions are parsed with feedparser; entry.summary and entry.content fields are concatenated, HTML-stripped, and cleaned. PDF submissions are parsed with pypdf, with per-page text concatenated after whitespace normalisation. All extraction paths converge on a shared validation step that rejects submissions whose cleaned text falls below a minimum token threshold and returns structured error messages in place of a prediction.

C. Prediction Engine

The Prediction Engine loads pre-trained model artefacts and selects an inference strategy at request time based on the operator-configured mode parameter. In ensemble-only mode, four scikit-learn classifiers are queried and their soft probabilities

combined with learned weights. In LSTM-only mode, the sequence model alone determines the output probability. In hybrid mode— the default— the two probability streams are combined with a fixed mixing coefficient. Strategy selection requires no retraining and allows operators to trade off interpretability against predictive power depending on context.

D. Decision Logic and Explainability

The Decision Logic Layer receives the final fake-class probability and applies the calibrated threshold gate. SHAP TreeExplainer values are computed for the Random Forest component; the ten features with the largest absolute SHAP values are extracted and ranked. A claim verification step runs in parallel: spaCy named-entity recognition extracts factual claims, which are matched against the indexed fact-check database via BM25 retrieval. A support score $s \in \{-1, 0, +1\}$ is aggregated across retrieved checks and used to apply a post-hoc adjustment $\Delta p = 0.05 \cdot s$ before thresholding. The Explainability Layer assembles the verdict, confidence score, ranked SHAP features, and claim verification summary into a JSON payload that drives both the Chart.js visualisation in the user dashboard and the plain-language narrative paragraph.

V. Methodology

A. Feature Engineering

Three feature groups are constructed and concatenated to form the classical classifier input vector. (i) *TF-IDF features:* a 50,000-term unigram-bigram TF-IDF matrix with sublinear term-frequency weighting and L2 normalisation. (ii) *Linguistic features:* sentiment polarity and subjectivity computed with TextBlob, punctuation density, capitalisation ratio, type-token ratio, and mean sentence length. (iii) *Pattern signals:* binary flags for the presence of sensationalist lexicon, hedge phrases, and first-person plural appeals to authority. The LSTM model receives a zero-padded integer token-index sequence of fixed length 512.

B. Ensemble Classifier Training

Four scikit-learn classifiers are trained on the concatenated feature vector with the following configurations. MultinomialNB uses Laplace smoothing ($\alpha = 1$). LogisticRegression uses L2 regularisation with C = 1.0 and is trained with the lbfgs solver. LinearSVC is calibrated with CalibratedClassifierCV to produce soft probability estimates. RandomForestClassifier uses 300 estimators and bootstrap sampling. Ensemble weights w = [0.15, 0.30, 0.30, 0.25] were determined by grid search on the held-out validation split, minimising false-negative rate subject to overall accuracy remaining above 90%.

C. LSTM Architecture

The LSTM model consists of an embedding layer (vocabulary size 100,000, embedding dimension 128, trainable from scratch), two stacked LSTM layers each with 128 units and 0.3 recurrent dropout, a global max-pooling layer, a dense layer of 64 units with ReLU activation and 0.3 dropout, and a sigmoid output unit. The model is trained for a maximum of 20 epochs using the Adam optimiser (learning rate 1e-3) with early stopping on validation loss (patience = 3). Binary cross-entropy is the loss function.

D. Hybrid Combination and Confidence Gating

In hybrid mode, ensemble and LSTM fake-class probabilities are linearly combined as:

$$p_{hybrid} = \alpha \cdot p_{ensemble} + (1-\alpha) \cdot p_{LSTM}, \quad \alpha = 0.6$$

The mixing coefficient $\alpha = 0.6$ was selected by grid search on the validation split. Following claim verification adjustment (Δp), the final probability $p_{final} = p_{hybrid} + \Delta p$ is compared against thresholds $\tau_{hi} = 0.65$ and $\tau_{lo} = 0.35$.

Articles with $p_{final} \geq \theta_{hi}$ are labelled FAKE; articles with $p_{final} \leq \theta_{lo}$ are labelled REAL; all others receive the UNCERTAIN verdict. Thresholds were calibrated to keep the UNCERTAIN abstention rate below 12% while minimising the false-negative rate.

E. Claim Verification

spaCy's en_core_web_lg model performs named-entity recognition on the cleaned article text, extracting PERSON, ORG, GPE, DATE, and CARDINAL entities. Factual claims are formed by pairing entities with their surrounding verb phrases using dependency parsing. Each claim is matched against an index of 127,000 fact-check articles from Snopes, PolitiFact, and FactCheck.org using BM25 retrieval (top-5 results). Retrieved articles are classified as contradicting ($s = -1$), neutral ($s = 0$), or supporting ($s = +1$) the claim using a fine-tuned RoBERTa-base natural language inference model. The mean support score across all verified claims is used to compute p.

VI. Implementation

A. Technology Stack

The backend is implemented in Python 3.10. Flask 2.3 serves the REST API with JWT-based authentication and role-based access control separating standard users from administrators. scikit-learn 1.3 provides classical classifiers and TF-IDF vectorisation. TensorFlow 2.13 / Keras trains and serves the LSTM model. spaCy 3.6 handles named-entity recognition and dependency parsing for claim extraction. BeautifulSoup4 and requests handle URL content extraction; feedparser handles RSS; pypdf handles PDF. MongoDB 6.0 stores user records, submission history, and model artefact metadata. The frontend is a single-page React 18 application using Chart.js for result visualisation and Axios for API communication.

B. System Deployment

The system is packaged with Docker Compose orchestrating three services: the Flask API server, a MongoDB instance with journaling enabled for write durability, and a model serving container that loads scikit-learn and TensorFlow artefacts at startup. Nginx acts as a reverse proxy handling SSL termination, request routing, and rate limiting. The deployment has been validated on a quad-core workstation with 16 GB RAM and no GPU. Horizontal scaling at the API layer is achievable through standard container orchestration tools without architectural changes.

VII. Experimental Results

A. Experimental Setup

Experiments used a balanced benchmark corpus of 44,898 labelled articles drawn from three sources: the LIAR dataset [4] (12,791 articles covering six veracity categories collapsed to binary labels), FakeNewsNet [10] (23,907 articles from GossipCop and PolitiFact), and a web-scraped supplement of 8,200 articles from fact-check-verified sources collected during 2023–2024. The corpus is balanced between REAL and FAKE at the binary level. A stratified 70/15/15 train/validation/test partition was applied. Evaluation metrics are Accuracy, Precision, Recall, macro-F1 score, and False-Negative Rate (FNR). FNR receives primary emphasis because undetected fake news constitutes the highest-risk failure mode in editorial workflows. All experiments ran on an Intel Core i7-10700 workstation with 16 GB RAM and no GPU.

B. Classification Performance

Table II reports classification metrics for all evaluated methods. The NewsZ hybrid strategy achieves 92.4% accuracy and a 4.1% FNR, outperforming every single-model baseline. The LSTM alone (90.2%) outperforms the full ensemble (91.3%) on recall, reflecting its advantage on longer articles where discourse-level context matters. Combining the two streams in hybrid mode recovers precision while preserving the recall gain. Claim verification contributes a further 0.4 pp FNR reduction over hybrid inference alone.

TABLE II. Classification Performance Comparison

Method	Accuracy	Precision	Recall	F1	FNR
Naive Bayes	83.1%	0.82	0.81	0.81	9.8%
Logistic Regression	87.6%	0.87	0.86	0.86	7.4%
SVM	88.9%	0.89	0.88	0.88	6.8%
Random Forest	89.4%	0.89	0.89	0.89	6.5%
LSTM	90.2%	0.90	0.90	0.90	5.9%
Ensemble Only	91.3%	0.91	0.91	0.91	5.3%
NewsZ Hybrid	92.4%	0.93	0.92	0.92	4.1%

C. Multi-Source Input Testing

Functional testing evaluated six input categories across 268 total submissions. Results are reported in Table III. The primary failure mode for URL submissions was bot-detection blocking by the target server (3.5% of URL submissions), which is handled gracefully by returning a structured extraction-failed response rather than propagating an uncaught exception. PDF extraction failures (6.0%) arose exclusively from scanned documents with no embedded text layer. UNCERTAIN abstention accuracy of 91.3% was measured by comparing the model's abstention decisions against independent expert labels on a set of 46 deliberately borderline articles.

TABLE III. Multi-Source Functional Test Results

Test Category	Submissions	Successful	Success Rate
Plain-text input	60	60	100.0%
URL extraction	57	55	96.5%
RSS feed parsing	50	49	98.0%
PDF extraction	25	23	94.0%
UNCERTAIN abstention	46	42	91.3%
History retrieval	30	30	100.0%

D. Response Latency Analysis

Table IV reports end-to-end latency statistics measured over 500 requests after a 20-request warm-up. Plain-text submissions complete in a mean of 0.42 seconds, dominated by model inference. URL submissions average 2.1 seconds, dominated by network fetch and HTML cleaning. All response times fall within the interactive-use requirements of editorial fact-checking workflows, where submissions typically involve a brief review period before the analyst reads the result.

TABLE IV. End-to-End Response Latency Statistics (seconds)

Input Mode	Mean (s)	Std Dev (s)	P50 (s)	P95 (s)
Plain Text	0.42	0.08	0.40	0.61
URL Fetch	2.10	0.64	2.05	3.45
RSS Feed	1.65	0.41	1.60	2.80
PDF (□ 20 pp.)	1.80	0.52	1.75	2.95

E. Explainability User Study

A controlled user study (n = 24 journalists, fact-checkers, and platform moderators) evaluated the practical utility of the NewsZ explanation interface compared to a black-box baseline presenting only the verdict and confidence score. Participants reviewed 10 articles each under both conditions in counterbalanced order. NewsZ explanations were rated 'useful' or 'very useful' by 87.5% of participants. Willingness-to-act ratings on a 5-point Likert scale were significantly higher for the NewsZ condition than the black-box baseline (4.2 vs. 2.9, paired t-test $p < 0.01$). Qualitative feedback identified keyword salience charts as the most actionable explanation component, followed by the claim verification summary.

VIII. Discussion

A. Performance Analysis

The experimental results confirm that hybrid model fusion yields consistent improvements over every single-model baseline across all metrics. The 1.1 percentage-point accuracy gain of the hybrid strategy over the ensemble alone translates to a 1.2 pp reduction in false-negative rate— approximately 54 fewer undetected fake articles per 4,500 test items. The LSTM component contributes disproportionate benefit on articles exceeding 500 tokens, where long-range discourse coherence signals complement the bag-of-words features used by the classical ensemble. Claim verification adjustments contributed a further 0.4 pp FNR improvement over hybrid inference alone, validating the integration of external fact-check evidence as a meaningful signal rather than noise.

B. Limitations

Three limitations merit acknowledgement. First, the scope of the current system is restricted to English-language content; multilingual generalisation requires separate training on parallel corpora and language-specific NLP pipelines, which is left for future work. Second, web pages protected by Cloudflare, Akamai, and similar bot-detection services cannot be scraped; users must submit text manually for such sources. Third, the fact-check database covers mainstream anglophone political and public health topics and has sparse coverage for regional, financial, or highly technical misinformation domains, which may limit claim verification effectiveness outside these categories.

C. Practical Deployment Considerations

Production deployment requires attention to several operational matters. JWT-based authentication with role-based access control separates standard users— who can submit articles and access their own history—from administrators, who can view aggregate analytics, inspect per-model metrics, and examine verdict distribution calibration over time. Query and response logging for audit purposes is implemented at the Flask middleware layer and does not affect inference latency. Model artefacts are versioned in MongoDB metadata, enabling rollback if a new model update degrades production performance. The Docker Compose deployment has been validated with submission volumes up to 1,200 requests per hour on the reference workstation hardware.

IX. Conclusion

This paper presented NewsZ, an explainable hybrid AI system for fake news detection that combines classical ensemble learning, LSTM sequence modelling, claim-level fact-checking, and multi-source content extraction within a confidence-gated decision framework. The system accepts raw text, URLs, RSS feeds, and PDF documents as input, applies a selectable prediction strategy (ensemble, LSTM, or hybrid), emits a three-class verdict with calibrated uncertainty handling, and delivers SHAP-based keyword attribution and plain-language explanations through a React user dashboard and admin analytics panel.

Experiments on a 44,898-article balanced benchmark corpus demonstrated 92.4% hybrid accuracy and a false-negative rate of 4.1%— the lowest among all evaluated baselines— confirming that model fusion with claim-level grounding reduces the most damaging failure mode in fake news detection. The UNCERTAIN verdict mechanism bounded overconfident incorrect decisions within an acceptable abstention rate. A controlled user study confirmed significantly higher willingness-to-act under the NewsZ explanation interface compared to a black-box baseline.

Future directions include: (1) integration of transformer-based models (RoBERTa, DeBERTa) as an additional prediction head in the hybrid engine; (2) multilingual support via cross-lingual embeddings and language-specific

preprocessing; (3) source credibility scoring using historical publisher reliability signals as a complementary feature channel; (4) active learning from administrator-verified correction feedback to reduce model drift; and (5) cloud-native deployment with automated performance monitoring and scheduled model retraining pipelines.

References

[1] C. Castillo, M. Mendoza, and B. Poblete, "Information Credibility on Twitter," in *Proc. WWW*, Hyderabad, India, 2011, pp. 675–684.

[2] M. Potthast, J. Kiesel, K. Reinartz, J. Bevendorff, and B. Stein, "A Stylometric Inquiry into Hyperpartisan and Fake News," in *Proc. ACL*, Melbourne, Australia, 2018, pp. 231–240.

[3] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[4] W. Y. Wang, "'Liar, Liar Pants on Fire': A New Benchmark Dataset for Fake News Detection," in *Proc. ACL*, Vancouver, Canada, 2017, pp. 422–426.

[5] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proc. NAACL-HLT*, Minneapolis, MN, 2019, pp. 4171–4186.

[6] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why Should I Trust You?': Explaining the Predictions of Any Classifier," in *Proc. KDD*, San Francisco, CA, 2016, pp. 1135–1144.

[7] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.

[8] V. H. Nguyen, K. Sugiyama, P. Nakov, and M.-Y. Kan, "FANG: Leveraging Social Context for Fake News Detection Using Graph Representation," in *Proc. CIKM*, 2020, pp. 1165–1174.

[9] K. Popat, S. Mukherjee, J. Strötgen, and G. Weikum, "CredEye: A Credibility Analysis Tool for Analyzing and Explaining Misinformation," in *Proc. WWW Companion*, Lyon, France, 2018, pp. 155–158.

[10] K. Shu, D. Mahudeswaran, S. Wang, D. Lee, and H. Liu, "FakeNewsNet: A Data Repository with News Content, Social Context, and Spatiotemporal Information for Studying Fake News," *Big Data*, vol. 8, no. 3, pp. 171–188, 2020.

[11] A. Zubiaga, A. Aker, K. Bontcheva, M. Liakata, and R. Procter, "Detection and Resolution of Rumours in Social Media: A Survey," *ACM Computing Surveys*, vol. 51, no. 2, pp. 1–36, 2018.

[12] N. Vo and K. Lee, "The Rise of Guardians: Fact-Checking URL Recommendation to Combat Fake News," in *Proc. SIGIR*, 2018, pp. 275–284.

[13] F. Alam, F. Dalvi, S. Shaar, N. Durrani, H. Sajjad, A. Nikolov, G. Da San Martino, A. Abdelali, H. Mubarak, A. Freihat, and P. Nakov, "Fighting the COVID-19 Infodemic in Social Media: A Holistic Perspective and a Call to Arms," in *Proc. AAAI*, 2021.