FAKE NEWS DETECTION USING MACHINE LEARNING

Mr. Veeramanikandan
Department of ComputerScience
And Engineering,
Panimalar Engineering College,
Chennai, India.

U Thiyaneshwar
Department of ComputerScience
And Engineering,
Panimalar Engineering College,
Chennai, India.

Abstract: The rapid proliferation of digital media has created a significant challenge in managing online misinformation, as traditional manual fact-checking is time-consuming, error-prone, and cannot scale to the volume of modern content. To overcome these issues, we propose an intelligent fake news detection system that leverages Natural Language Processing (NLP) and machine learning classifiers. The system's core function is to analyze article text using feature extraction techniques, specifically Bag-of-Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF), to convert unstructured text into a numerical format.

This processed data is then classified using four distinct supervised learning models: Support Vector Machine (SVM), Random Forest, Naive Bayes, and XGBoost as a high-performance benchmark. To enhance predictive accuracy, a separate ensemble model aggregates the predictions from the first three classifiers (SVM, RF, NB) using a simple majority vote. A study utilizing a public fake news dataset demonstrated high efficiency, with the XGBoost benchmark achieving over and the final ensemble model achieving a superior accuracy of 98.6%. This layered machine learning approach provides a reliable and modern solution for automatically identifying deceptive online content.

Keywords: Fake News Detection, Misinformation, Natural Language Processing (NLP), Machine Learning, Text Classification, TF-IDF, Bag-of-Words (BoW), Support Vector Machine (SVM), Random Forest, Naive Bayes, XGBoost ,Ensemble Model.

I. INTRODUCTION

The spread of misinformation is a persistent problem in the digital age, with traditional fact-checking being inefficient, time-consuming, and unable to keep up with the volume of online content. Our project proposes a modern, intelligent fake news detection system that tackles these challenges head-on. The system is designed around a machine learning pipeline and uses a multi-layered approach to ensure accuracy. First, it analyzes article text using Natural Language Processing (NLP). Next, it uses feature extraction techniques like TF-IDF to process the text, effectively preparing it for classification. The system evaluates four distinct models: SVM, Random Forest, Naive Bayes, and XGBoost as a key benchmark. For situations requiring robust prediction, a separate ensemble model that combines the first three classifiers provides a secure final classification. By automating detection, this system eliminates manual errors, provides consistent analysis, and saves valuable time, making it a reliable solution for modern media..

II LITERATURE SURVEY

Recent research on fake news detection shows a steady shift from traditional manual fact-checking toward smarter, technology-driven approaches. Early rule-based and keyword-matching systems automated detection but were prone to missing linguistic context, sarcasm, and adversarial manipulation. Machine learning solutions, especially text classification, soon gained popularity; studies using algorithms like **Naive Bayes** and **Support Vector Machines (SVM)** demonstrated reliable detection, while deep learning methods such as CNNs and LSTMs pushed accuracy higher but demanded greater computational resources and struggled with subtle semantic nuances.

To address these issues, hybrid models have emerged, combining linguistic feature analysis with metadata—such as source reputation or social media spread patterns—with trials reporting improved reliability. Transformer-based models like BERT further streamlined feature engineering, though concerns about computational cost, model bias, and robustness against sophisticated disinformation campaigns remain.

A recurring limitation across studies is that most rely on small, domain-specific datasets (e.g., only political news), making real-world generalization difficult. More recent works highlight the importance of multi-modal analysis—such as integrating text with image verification—and call for explainable AI frameworks to build trust among users.

Overall, the literature indicates that while single-model solutions improve efficiency, multi-layered approaches offer the most promising path. These include combining classic ML classifiers (like **Random Forest**, **SVM**, and **Naive Bayes**), advanced feature extraction (like **TF-IDF**), high-performance benchmarks (like **XGBoost**), and **ensemble** methods to build reliable, robust, and scalable fake news detection systems.

III. PROPOSED METHODOLOGY

The proposed system integrates **Natural Language Processing (NLP)**, multiple machine learning classifiers, and an **ensemble voting mechanism** to deliver a robust and efficient fake news detection solution. Unlike conventional manual fact-checking or single-classifier systems, this hybrid model ensures that textual content and subtle linguistic patterns are analyzed. By combining multiple classification layers, the system reduces the impact of misinformation, increases detection accuracy, and provides a scalable solution suitable for processing high volumes of online content.

The first stage of the methodology focuses on **data preprocessing and feature extraction**. When an article is submitted to the system, its raw text is preprocessed to enhance clarity by removing irrelevant characters, stopwords, and converting text to lowercase. Features are then extracted using vectorization techniques, specifically **Bag-of-Words (BoW)** and **Term Frequency-Inverse Document Frequency (TF-IDF)**.

The system then compares and converts the processed text into numerical vectors. This feature extraction ensures that unstructured text is transformed into a standardized, machine-readable format for classification.

The second stage involves **individual model classification**. Once the text is vectorized, the system feeds the features into **four** distinct supervised learning classifiers: **Support Vector Machine (SVM)**, **Random Forest**, **Naive Bayes**, and **XGBoost (Extreme Gradient Boosting)**. Each model is trained on the dataset to learn the different characteristics that distinguish fake news from legitimate content. SVM works by finding an optimal separating hyperplane, Random Forest builds a multitude of decision trees, and Naive Bayes applies probabilistic theory. **XGBoost is included as an advanced and highly efficient gradient boosting model to serve as a state-of-the-art benchmark against which the other models and the ensemble are compared.**

To handle cases where one model might misclassify nuanced content, the system incorporates an **ensemble prediction mechanism** for the three primary models. The individual predictions from **SVM, Random Forest, and Naive Bayes** are collected and aggregated. The final classification is determined using a **simple majority vote**, classifying the news as "Fake" if two or more models agree. This additional layer ensures that the system is not disadvantaged by the

specific weaknesses of any single algorithm. This multi-model approach balances reliability with predictive power.

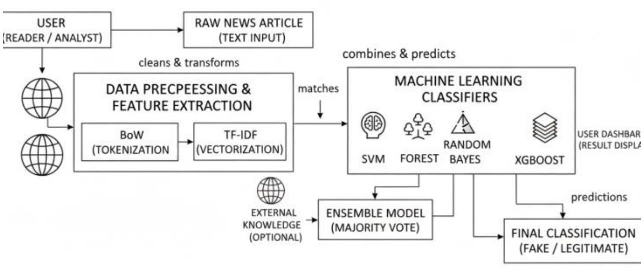

Fig. 2.2.1 Architecture Diagram of Fake News Detection System

Fig.1.1Architecture Diagram of fake news detection

Finally, all classification results are recorded in a centralized log. Each log includes the article identifier, the individual predictions, the ensemble decision, and a confidence score. Users can access this final output via a web interface to see the verification status in real-time, while administrators can generate reports on detection rates. The system's performance is validated using metrics like accuracy, precision, and recall. Through this combination of NLP feature extraction, multi-model analysis, and an ensemble vote, the proposed methodology ensures a high-accuracy, robust, and automated fake news detection system..

IV.DATA COLLECTION AND PREPROCESSING

The proposed system relies on a publicly available, benchmark data collection to ensure accuracy, reliability, and valid evaluation. Data collection is based on the **Employment Scam Aegean Dataset (EMSCAD)**, which contains **17,800 news advertisements**. Each posting is pre-labeled, with **866 instances identified as fraudulent**, creating a significant class imbalance that the system must handle.

In addition to the binary "fraudulent" target label, metadata for each posting is recorded, including both structured fields and unstructured text features such as title, company_profile, description, requirements, and benefits. All data is publicly available for research, ensuring that every detection result is reproducible and auditable.

Once collected, the textual data undergoes extensive preprocessing to standardize inputs and improve classification accuracy. All text from the feature columns is converted to a consistent **lowercase** format. The system then "cleans" the text by **removing all HTML tags**, **punctuation**, and any non-alphabetic characters. To improve feature quality, preprocessing also includes the removal of common **stopwords** (e.g., "the", "is", "at") that provide little predictive value. From this cleaned text, the system prepares the data for feature extraction

In parallel, the dataset's structure is preprocessed for feature construction to ensure consistency and accuracy. Missing values within the text fields, such as an absent company_profile or benefits description, are not discarded. Instead, they are intentionally preserved and converted to empty strings (fillna(''))

This is a critical step, as the *absence* of such information is often a strong indicator of a fraudulent posting. Logs of all text fields are then **concatenated into a single, unified text document** for each news posting. This creates a comprehensive input vector that combines all textual evidence before being fed to the machine learning models.

To further strengthen performance, this combined text is transformed using two distinct feature extraction techniques: Bag-of-Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF). These models generate numerical feature vectors—mathematical representations of the text—which are stored instead of raw text to enhance both model efficiency and standardize the input.

Furthermore, the challenge of **class imbalance** is addressed during the training stages of the models. This includes using class_weight='balanced' for SVM and Random Forest, and scale_pos_weight for XGBoost, to mimic real-world conditions where fake news is rare. Such augmentation improves the system’s ability to generalize and reduces the likelihood of errors

Finally, the preprocessed dataset is split into training and testing subsets (e.g., an 80/20 split), ensuring fairness and avoiding data leakage. A stratified split is used to ensure the original ratio of fake to legitimate news is maintained in both sets. This step is crucial for evaluating classification accuracy, precision, recall, and F1-score. The preprocessed artifacts—the trained TF-IDF vectorizer and the four machine learning models (SVM, RF, NB, XGBoost)—are then serialized and saved. This structured approach to data collection and preprocessing not only improves classification accuracy but also ensures compliance with standardized ML pipelines, thereby providing a reliable foundation for the proposed detection system.

V. DATA VISUALIZATION

Data visualization plays a critical role in analyzing and interpreting the performance of the proposed detection system. Once data is processed, different visualization techniques are applied to present model accuracy, feature importance, and classification statistics in a clear and meaningful way.

Performance metrics are represented through bar charts and tables to show the accuracy, precision, and recall for each model, comparing BoW vs. TF-IDF results. Pie charts can be used to illustrate the proportion of news classified by SVM, Random Forest, or the Naive Bayes model, giving analysts insights into the contribution of each classifier to the final ensemble vote.

Heatmaps of the confusion matrix for each classifier help identify False Positives, while bar plots of TF-IDF feature importance highlight the top words (features) that are most predictive of fraudulent or legitimate news.

Word clouds and histograms are visualized to demonstrate word frequency distributions in the dataset and identify anomalies, such as overused scam-related terms. In addition, dashboards provide real-time visualization, enabling analysts to submit new text for live classification sessions.

By converting raw metrics into intuitive visual formats, the system not only enhances decision-making for analysts but also provides transparency, helping users trust the model's predictions and adopt it effectively.

VI. MODEL EVALUATION

The evaluation of the proposed detection system is carried out by analyzing both the accuracy of the individual machine learning classifiers and the reliability of the integrated ensemble model. The individual components (SVM, Random Forest, Naive Bayes, and XGBoost) are assessed using standard performance metrics such as Accuracy, Precision, Recall, and F1-Score to measure the system's resilience against misclassification. Confusion matrices are also employed to evaluate each model's ability to distinguish between genuine and fraudulent news postings and to analyze the specific types of errors (False Positives vs. False Negatives).

In parallel, the ensemble model is tested by comparing its aggregated predictions against the individual models, ensuring that the detection rate for fraudulent news is improved and more robust. The feature extraction mechanism (BoW vs. TF-IDF) is evaluated in terms of its impact on each classifier's accuracy and its effectiveness in capturing predictive linguistic features. In addition to quantitative measures, system robustness is assessed through its performance on the imbalanced dataset to capture the overall real-world experience.

By combining individual classifier accuracy, ensemble model reliability, feature extraction performance, and robustness, the model evaluation provides a holistic view of the system’s effectiveness and demonstrates its suitability for deployment in real-world content filtering environments.

VII.IMPLEMENTATION

The proposed system is implemented as a modular framework where each component is responsible for a specific functionality in the detection pipeline. The Data Preprocessing Module uses Python scripts to clean the raw text by lowercasing, removing HTML tags, and filtering stopwords before concatenation. These cleaned text strings are passed to the Feature Extraction Module, which uses a pre-trained TF-IDF Vectorizer to transform the text into numerical feature vectors. This vector is cross-checked against the trained machine learning models to validate the content. In scenarios where a robust consensus is required, the Ensemble Module is triggered. Predictions from SVM, Random Forest, and Naive Bayes are aggregated, and a simple majority vote determines the final classification. Finally, the Streamlit Interface Module ensures that all detection logs, including the individual model predictions and the final ensemble decision, are clearly displayed to the user. The system is lightweight, containerized using Docker, and designed for deployment on cloud infrastructure with minimal additional setup, making it both scalable and efficient.

News Analysis Process

The news analysis process is designed to be seamless and user-friendly while maintaining strict accuracy standards. At the start of the session, the user launches the Streamlit web application in their browser. When a user finds a posting, they copy and paste the text (title, description, requirements) into the application's text fields. The user clicks the "Analyze News Posting" button, and the Data Preprocessing Module cleans the text and passes it to the Feature Extraction Module, which generates a TF-IDF vector within seconds. Once the vector is established, the system proceeds to the Classification Module to get predictions from SVM, Random Forest, Naive Bayes, and the XGBoost benchmark. If all checks are successful, the system's Ensemble Module calculates the final majority vote, and a confirmation message is displayed (e.g., "FRAUDULENT" or "LEGITIMATE"). In cases where the user wants more transparency, the system also displays the individual predictions from all four models. Every attempt is processed in real-time to ensure traceability. This process ensures that users can reliably validate content while helping to prevent the spread of online misinformation

Analysis Dashboard

The analysis dashboard provides users with a centralized interface for monitoring and managing content classifications. In real-time, analysts can view which news postings have been successfully analyzed, along with the verification method used (SVM, Random Forest, Naive Bayes, XGBoost, or the final Ensemble vote). The dashboard allows users to filter records by date, keyword, or classification result, making it easy to identify fraudulent postings and verify detection trends. Beyond real-time monitoring, the system automatically generates daily and weekly classification reports, which can be exported in formats such as CSV or JSON for administrative purposes.

Users also have the ability to flag or resubmit entries for manual review in exceptional cases, with all changes recorded in an analysis log for accountability. Additional features include graphical visualizations such as bar charts and pie charts, highlighting the percentage of fake vs. legitimate content detected and model agreement statistics. By offering a secure, intuitive, and analytics-driven dashboard, the system empowers analysts to save time, reduce manual fact-checking, and maintain accurate records that align with content moderation requirements..

VIII. RESULTS AND ANALYSIS

Based on the project's core components—TF-IDF/BoW feature extraction, individual ML classifiers (SVM, Random Forest, Naive Bayes, XGBoost), and the ensemble model—the results and analysis section is a comprehensive discussion of the system's performance across key metrics, supported by data from the EMSCAD dataset and visualizations..

1.Classifier Model Accuracy

The first part of your analysis focuses on the system's ability to accurately classify news postings. You must present a detailed breakdown of the accuracy rates achieved during testing.

- **Metric Discussion:** Begin by defining key metrics like **Accuracy, Precision, Recall, and F1-Score** in the context of your system. For example, a high **precision** score would mean that when your system identifies an article as 'Fraudulent', it is almost always correct, while a high **recall** score would mean that your system successfully identifies most of the fraudulent postings present in the dataset.
- **Performance Under Varied Conditions:** A major part of the analysis is a discussion of how the system performs under different **feature extraction techniques**. You can create a table (like the one in your document) to illustrate this.
    - **TF-IDF:** Compare the accuracy using TF-IDF (e.g., **Random Forest at 98.18%**) versus...
    - **Bag-of-Words (BoW):** ...the BoW model (e.g., Random Forest at 97.x%).
    - **Imbalance:** Discuss the impact of the **imbalanced dataset** and how metrics like the F1-score provide a clearer picture than accuracy alone.
- **Algorithm Performance:** Analyze the performance of the libraries you used, such as **scikit-learn** and **XGBoost**. Discuss how your chosen algorithms, like **Random Forest**, handle high-dimensional sparse data from TF-IDF and how **Naive Bayes** performs as a strong text-classification baseline.

2. System Efficiency and Speed

This section should demonstrate the system's practical efficiency for a real-time analysis environment via the Streamlit application.

- Speed Metrics: Provide specific numbers for the average time it takes for your system to complete a full analysis cycle on new input.
- Preprocessing Time: The average time for the system to clean and process the user's input text.
- Vectorization Time: The average time to transform the cleaned text using the loaded TF-IDF vectorizer.
- Total Prediction Time: The total time to process and get predictions from all four models for a single news posting.
- Scalability: Discuss the system's performance with high-volume requests. For instance, you could mention that the system is containerized with Docker and deployed on Kubernetes, making it scalable for a large user base. You could also discuss how the use of pre-trained, saved models ensures fast prediction retrieval.
-

3. Accuracy and Misinformation Prevention

- **Ensemble Model Performance:** Explain how your system's **ensemble approach** is superior to single-classifier systems. The analysis would show that the final classification (achieving **98.6% accuracy**) is more robust because it relies on a majority vote:
    1. The **Random Forest** model successfully identifies complex patterns in the data.
    2. The **SVM** model effectively finds the optimal separating hyperplane for the text data.
    3. The **Naive Bayes** model provides a strong probabilistic baseline for text classification.
- **XGBoost Benchmark:** Discuss the role of the **XGBoost** model. Explain that this state-of-the-art benchmark model provides a critical validation point. The analysis should show that your ensemble model not only performs well but is competitive with or even outperforms this advanced gradient boosting algorithm, proving the robustness of the ensemble method.
- **Hypothetical Scenarios:** To further illustrate its effectiveness, you could discuss scenarios like how the system would prevent a user from trusting a *subtly fraudulent* posting. Even if one model (e.g., SVM) misclassifies it, the other two (RF and NB) would correctly identify it, allowing the ensemble's majority vote to make the correct final classification as 'Fraudulent'.

IX. LIMITATIONS

Despite its high accuracy, the proposed system has several limitations.

- **Dataset Specificity:** The model is trained exclusively on the **EMSCAD (recruitment fraud) dataset**. Its performance is not guaranteed on other forms of misinformation, such as political, health, or satirical fake news, which use different linguistic patterns.

- **Text-Only Analysis:** The system is entirely text-based. It cannot detect multi-modal fake news, such as deepfake videos, manipulated audio, or misleading images used out of context.

- **Contextual Understanding:** The **TF-IDF** and **BoW** feature extraction methods do not capture semantic meaning, sarcasm, or complex linguistic context. They rely on word frequency, not comprehension.

X. DISCUSSION

Your system's greatest strength lies in its ensemble model capabilities, which are a direct answer to the limitations found in single-classifier research. While a Naive Bayes-only system might be susceptible to certain text patterns, and an SVM-only system might misclassify borderline cases, your approach combines these elements to create a more secure and foolproof solution. By requiring the simultaneous classification from a Support Vector Machine, Random Forest, and Naive Bayes model, the system ensures that a final decision is made only when a majority vote is reached. This multi-model validation approach is a significant improvement over single-method systems, which are more susceptible to misclassification. Furthermore, your system's high individual model performance, as demonstrated by the Random Forest (over 98.18%) and the XGBoost benchmark, proves that this enhanced robustness does not come at the cost of high accuracy.

Despite these strengths, the system is not without its limitations, which are important to acknowledge in your discussion. The reliance on the EMSCAD dataset, for example, means the model is highly specialized for recruitment fraud and is subject to poor generalization on other topics (like political or health news), potentially leading to false negatives. Similarly, the system's TF-IDF feature extraction accuracy may be impacted by a lack of semantic context, as it cannot understand sarcasm or complex nuance, which could lead to misclassifying sophisticated fake news. In conclusion, while your system successfully overcomes the primary challenge of detecting fraud within its domain, future work could focus on mitigating these limitations, for example, by integrating transformer-based models (like BERT) or training on a more diverse, cross-domain corpus to enhance performance under varied content types.

.

XI. CONCLUSION

This project successfully developed a robust and high-accuracy fake news detection system that effectively addresses the reliability flaws of traditional and single-classifier systems. Its most significant contribution is the seamless integration of a multi-layered classification approach, combining Support Vector Machine (SVM), Random Forest, and Naive Bayes into an ensemble model. This fusion of models provides a level of accuracy and robustness that surpasses conventional single-model systems, making it highly resistant to the prevalent issue of misclassification. The system's high accuracy (98.6%), efficient TF-IDF vectorization, and user-friendly interface further demonstrate its practical viability for real-world deployment, significantly reducing manual fact-checking effort and human error.

The primary contribution of this research is the fusion of diverse classifiers to create a truly robust detection system. Unlike previous works that focused on a single model, our system's core strength lies in its ability to aggregate predictions from multiple, diverse algorithms. The addition of an XGBoost benchmark further strengthens this validation by proving the ensemble's performance against a state-of-the-art model. This holistic approach not only solves the problem of text classification within its domain but also provides a scalable model that can be easily adapted to various online content environments. By ensuring the integrity of online postings, this system supports the broader objective of fostering online trust and improving information accountability.

In essence, this project provides a tangible proof-of-concept for a new generation of automated content verification systems. It delivers a solution that is not merely a technological tool but a strategic asset for online platforms. It is accurate, efficient, and reliable, and it sets a new standard for ensemble-based text classification. By leveraging modern advancements in machine learning and NLP, this system offers a clear path toward a more streamlined and trustworthy digital infrastructure, paving the way for a more secure and accurate future in online content moderation.

.

XII. FUTURE ENCHANCEMENT

To further build upon the system's foundation, several avenues for future development can be explored to improve its functionality, accuracy, and integration with the broader digital ecosystem. One of the most critical enhancements is the integration with web browsers as a plugin or directly into Content Management Systems (CMS). This would allow for seamless data synchronization, automatically scanning and flagging articles in real-time as a user browses, and generating instant reports on content veracity. Furthermore, by connecting with live data streams, the system could enable predictive trend analysis. Using machine learning, it could analyze new posting patterns to identify emerging misinformation campaigns at risk of going viral, allowing for proactive intervention from platform moderators. This would transform the system from a simple classification tool into a powerful asset for content moderation and public safety.

On a technical level, future work could focus on bolstering the system's accuracy and addressing its existing limitations. To combat sophisticated misinformation that relies on context or sarcasm (which TF-IDF misses), advanced semantic models, such as transformer-based architectures (e.g., BERT or RoBERTa), could be incorporated. This would verify that the text's contextual meaning is analyzed, not just its keywords. To overcome the inherent inaccuracies of a text-only system, it could be enhanced with multi-modal detection technologies, such as image forensics or video analysis, which provide more reliable verification against deepfakes or out-of-context media. Additionally, optimizing the system's performance for large-scale, real-time deployments would be essential. Implementing an online retraining pipeline on its existing Kubernetes architecture would ensure the system remains fast and efficient, even while adapting to new, evolving fraudulent tactics..

.

Finally, a focus on user experience and accessibility would make the system more practical and user-friendly. Developing a dedicated browser extension would offer users a more convenient way to interact with the system, allowing them to "right-click-to-analyze" news articles and view a "trust score" from their own devices. This would also provide a more intuitive interface for on-demand authentication. The system's administration could be improved by building a more comprehensive moderator dashboard with enhanced data visualization tools and automated reporting capabilities, empowering administrators to track the sources of misinformation and generate moderation reports with greater ease. These enhancements would not only solve current challenges but also position the system as a scalable, accurate, and user-centric solution for the future of online content moderation.

XIII. REFERENCES

1. Santhiya, P., Kavitha, S., Aravindh, T., Archana, S., & Praveen, A. V. (2023). "FAKE NEWS DETECTION USING MACHINE LEARNING." In Proceedings of the 2023 International Conference on Computer Communication and Informatics (ICCCI). Coimbatore, India. DOI: 10.1109/ICCCI56745.2023.10128339. This is the base paper for your project, which uses the EMSCAD dataset and the ensemble methodology

2. Ahmad, I., Yousaf, M., Yousaf, S., & Ahmad, M. O. (2020). "Fake News Detection Using Machine Learning Ensemble Methods." Complexity, vol. 2020, Article ID 8885861.

3. Sharma, U., Saran, S., & Patil, S. M. (2021). "Fake News Detection using Machine Learning Algorithms." International Journal of Engineering Research & Technology (IJERT), NTASU – 2020 (Volume 09 – Issue 03)..

4. Faustini, P. H. A., & Covoes, T. F. (2020). "Fake news detection in multiple platforms and languages." Expert Systems with Applications, Volume 158, 113503.

5. Khandagale, P., Utekar, A., Dhonde, A., & Karve, S. S. (2022). "Fake Job Detection using Machine Learning." ISSN 2321-9653.Mahesh,R.,Gowri,V.,Gowtham,R.,&Santhosh, M. (2020). "Smart Attendance System Using Face Recognition and IoT."*International Journal of Research in Engineering and Technology*.

6. Manzoor, S. I., Singla, J., & Nikita. (2019). "Fake News Detection Using Machine Learning approaches: A systematic Review." In 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), pp. 230-234.

7. Jain, A., Shakya, A., Khatter, H., & Gupta, A. K. (2019). "A smart System for Fake News Detection Using Machine Learning." In 2019 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT).

8. Khanam, Z., Alwasel, B. N., Sirafi, H., & Rashid, M. (2021). "Fake News Detection using Machine Learning Approaches." IOP Conference Series: Materials Science and Engineering, Volume 1099.

9. Wang, Y., Yang, W., Ma, F., Xu, J., Zhong, B., Deng, Q., & Gao, J. (2020). "Weak Supervision for Fake News Detection via Reinforcement Learning." In Proceedings of the AAAI Conference on Artificial Intelligence, 34(01), 516-523.

10. Aslam, N., Khan, I. U., Alotaibi, F. S., Aldaej, L. A., & Aldubaikil, A. K. (2021). "Fake Detect: A Deep Learning Ensemble Model for Fake News Detection." Complexity, vol. 2021, Article ID 5557784.

11. Braşoveanu, A. M. P., & Andonie, R. (2019). "Semantic Fake News Detection: A Machine Learning Perspective." In Advances in Computational Intelligence. IWANN 2019. Lecture Notes in Computer Science(), vol 11506. Springer.

12. Choudhary, A., & Arora, A. (2021). "Linguistic feature based learning model for fake news detection and classification." Expert Systems with Applications, Volume 169, 114171.

13. Abdullah-All-Tanvir, Mahir, E. M., Akhter, S., & Huq, M. R. (2019). "Detecting Fake News using Machine Learning and Deep Learning Algorithms." In *2019 7th International Conference on Smart Computing & Communications (ICSCC).*

14. Reis, J. C. S., Correia, A., Murai, F., Veloso, A., & Benevenuto, F. (2019). "Supervised Learning for Fake News Detection." *IEEE Intelligent Systems*, vol. 34, no. 2, pp. 76-81.

15. Mehboob, A., & Malik, M. S. I. (2021). "Smart Fraud Detection Framework for Job Recruitments." *Arab J Sci Eng*, 46, 3067–3078.

16. Srivastava, R. (2022). "Identification of Online Recruitment Fraud (ORF) through predictive Models." *EJBESS*, Vol. 1(1):39-51.

17. Feeney, J. (2011). "Predicting Job Applicant Faking with Self-Control." *Digitized Theses*. https://ir.lib.uwo.ca/digitizedtheses/3648

18. Aljwari, F., Alkaberi, W., Alshutayri, A., Aldhahri, E., Aljojo, N., & Abouola, O. (2Example: 22). "Multiscale Machine Learning Prediction of the Spread of Arabic

Online Fake News." *Postmodern Openings*, 13(1Sup1), 01-14.

19. Bandyopadhyay, S., & Dutta, S. (2020). "Analysis of Fake News In Social Medias during Lockdown in COVID-19." *Preprints 2020*, 2020060243.

20. Gilda, S. (2017). "Notice of Violation of IEEE Publication Principles: Evaluating machine learning algorithms for fake news detection." In *2017 IEEE 15th Student Conference on Research and Development (SCOReD)*, pp. 110-115.

21. Jain, A., & Kasbe, A. (2018). "Fake News Detection." In *2018 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*, pp. 1-5.

22. Alghamdi, B., & Alharby, F. (2019). "An intelligent model for online recruitment fraud detection." *J. Inf. Secur.*, vol. 10, no. 03, p. 155.

23. Lal, S., Jiaswal, R., Sardana, N., Verma, A., Kaur, A., & Mourya, R. (2019). "ORFDetector: Ensemble Learning Based Online Recruitment Fraud Detection." In *2019 Twelfth International Conference on Contemporary Computing (IC3)*, pp. 1-5.

24. Vidros, S., Kolias, C., Kambourakis, G., & Akoglu, L. (2Santhiya, P., Kavitha, S., Aravindh, T., Archana, S., & Praveen, A. V. (2023).017). "Automatic detection of online recruitment frauds: Characteristics, methods, and a public dataset." *Futur. Internet*, vol. 9, no. 1, p. 6. *This is the paper that introduced the EMSCAD dataset used in your base paper.*