Smart Prediction Model for Finding Fake Job Recruitment Using Machine Learning

[1]Mr KH Shabbeer Basha, [2]Ketha Jahnavi, [3]Vemireddy Kathyayani, [4]Pamisetty Manasa, [5]Sandireddy Meghana

Department of Computer Science and Technology,
Madanapalle Institute of Technology and Science, Angallu, Madanapalle-517325

Abstract: The widespread use of online job portals has contributed to a sharp increase in fake job advertisements, placing job seekers at considerable financial and personal risk. Almost all of these methods rely on traditional machine learning algorithms, including Naïve Bayes, Logistic Regression, decision trees and Support Vector Machines. Although such methods are efficient in terms of computation, they lack the ability to understand deeper contextual meaning and semantic patterns within job descriptions, which limits their effectiveness against increasingly sophisticated scam postings. To address these shortcomings, this work presents a hybrid detection framework that combines contextual text representations generated by a BERT-based transformer, a deep learning model, with dense feature learning and a gradient boosting classifier, which represents a powerful machine learning ensemble technique. By leveraging contextual embeddings learned through deep neural architectures, the proposed model significantly reduces the dependence on handcrafted features. Furthermore, the framework incorporates cost-sensitive learning principles to explicitly account for the higher cost of misclassifying fraudulent job postings, thereby improving robustness in the presence of class imbalance without relying on aggressive under-sampling strategies. Experimental results demonstrate that the proposed hybrid approach achieves improved detection performance and better generalisation against evolving employment scam patterns.

Index Terms - fake job detection, BERT, deep learning, ensemble learning, feature engineering, fraud detection, natural language processing, and cybersecurity.

I. INTRODUCTION

A. Background and Motivation

Recruitments have changed a lot of thanks to digital technology. The way companies find new employees has totally changed. Now, millions of job postings pop up online every single day [3]. But here's the catch—scammers have jumped in, too. They know just how easy it is to post a fake job, pull people in, and trick them into handing over personal info, paying bogus fees, or, worst of all, giving up their whole identity. Every year, millions get caught by these schemes, losing about $2 billion in total. Roughly 5% to 10% of all job ads have something fishy going on.

People used to spot these fake postings with machine learning tools like Random Forest, support vector machines, and XGBoost [4]. They'd feed these models pretty basic features, mostly built from TF-IDF—the usual trick of counting how often words show up. This method helped a bit, but it only worked on a surface level [6]. These models just count words without actually understanding what they mean in context, so they miss a lot—especially when scammers use clever language.

And then, when it's time to check if a poster is legit, most systems just take whatever info the user gives and call it a day. They don't bother to see if the company behind the post actually exists, who owns the website, or where it's even registered. Details like how often someone posts or what time they do it? Those usually get ignored too. And honestly, the data's a mess. Real job posts completely drown out the fake ones, so people use tricks like SMOTE to even things out. The problem is that sometimes that just messes things up more, making the models learn stuff that doesn't matter—or isn't even real.

B. Research Contributions

Here's what this paper actually brings to the table:
We built a new hybrid system that blends classic ensemble methods with contextual transformers—basically, it's a smarter way to spot fake job postings. The setup uses eighteen features, all tailored for real-world industry needs [10]. Stuff like checking if a company's legit and looking closely at how it behaves. With this, we saw a big drop in false alarms.

Compared to the older approaches we started with, our experiments showed a 23% cut in false positives and a 7% jump in the F1 score [11]. So, what made the difference? Three things: BERT embeddings, a richer set of features, and our fresh ensemble design. You really see the impact of BERT when you stack the deep learning model with BERT against the one without it—the difference jumps out.

C. Paper Organization

Here's how the paper flows. First, Section 2 dives into the main research on spotting fake job ads. After that, Section 3 walks through our approach. Then we get into what our experiments turned up. Finally, the conclusion points out where this research could go next.

II. RELATED WORK

It is now more crucial than ever to detect fake job offers. Job boards are proliferating, and companies are increasingly hiring through social media, which frankly makes it easier for job seekers but also opens the door to fraudsters. Scam job ads are a pain, besides making job searching a nightmare [8]. They can put job seekers at risk by asking them to hand over sensitive information or bank details, and that's a very real threat. With all that in mind, researchers are turning to machine learning and natural language processing to create better tools to detect fake job ads before they do too much damage.

A. Traditional Machine Learning Approaches

Earlier fake job detection systems mostly relied on traditional machine learning classifiers, using handpicked features. Varaganti and their team tried out a Random Forest model on the Employment Scam Aegean dataset and got a 76% F1-score. They used TF-IDF for their feature vector. What stood out in their research was that combining different algorithms worked better than using just one. But there was a catch—the ensemble method struggled with understanding the context behind the data.

Who tackled the problem of class imbalance [7]. They combined SMOTE and ADASYN with a decision gradient approach. By generating synthetic data to boost the minority class, they improved detection rates. Even so, this method had its problems. When they made synthetic samples, strange glitches sometimes crept in and threw off the model's performance with real-world fraud.

B. Deep Learning and Transformer Models

Even so, this method had its problems. When they made synthetic samples, strange glitches sometimes crept in and threw off the model's performance with real-world fraud.

Earlier systems for spotting fake jobs stuck to traditional machine learning. People would pick features by hand and run them through standard classifiers. Varaganti's team, for example, used a Random Forest model on the Employment Scam Aegean dataset and landed a 76% F1-score. They built their feature vector with TF-IDF. The big takeaway? Mixing different algorithms worked way better than using just one. It was actually a real limitation.

C. Research Gap

A lot of current predictive models just don't connect the dots between context and domain expertise. They also tend to ignore useful validation signals from outside sources, and they don't always get the best out of traditional machine learning.

III. Proposed Work

Here's what we're doing differently. We're building a hybrid system that mixes domain-specific attribute extraction with ensemble methods—basically, we're throwing everything we've got at the problem of fake job postings. We use contextual BERT embeddings alongside tried-and-true machine learning algorithms, then layer on a meta-learning strategy that blends the strengths of each model. This way, we can catch both weird structural patterns and subtle language clues that usually signal a scam.

Unlike typical models that just rely on text features or try to pad their data, ours brings in real business verification data and tracks how users actually browse. We also use cost-sensitive learning to figure out which actions matter most. By bringing all these pieces together, our system spots harassment better, makes fewer mistakes, and just does a better job overall at flagging suspicious cases. It's not just theory—it's built for the real world.

A. System Architecture Overview

The pipeline hybrid system consists of 4 major components, and they are chained together sequentially.

1. Feature Extraction Layer:
The system extracts various features from job postings. It retrieves contextual embeddings from BERT, each with a dimension of 768. It also extracts TF-IDF features with a dimension of 5000 for establishing a baseline. Additionally, 18 job domain-specific features are appended.

2. Dual-Branch Processing:
The software splits into two parallel tracks. One path runs the BERT embeddings through dense neural networks. The other approach sticks with traditional machine learning. It uses models like XGBoost, Random Forest, and Gradient Boosting classifiers.

3. Meta-Learning Fusion Layer:
This part uses logistic regression as a meta-learner. It pulls together the outputs from all those different classifiers and learns how to combine them using the right weights.

4. Output Generation:
The system produces a few things here: it gives a probability score for fraud—like, "This transaction has a 0.87 chance of being fraudulent." It also does a binary classification (real or fake) and shows which features mattered most using SHAP explanation scores.

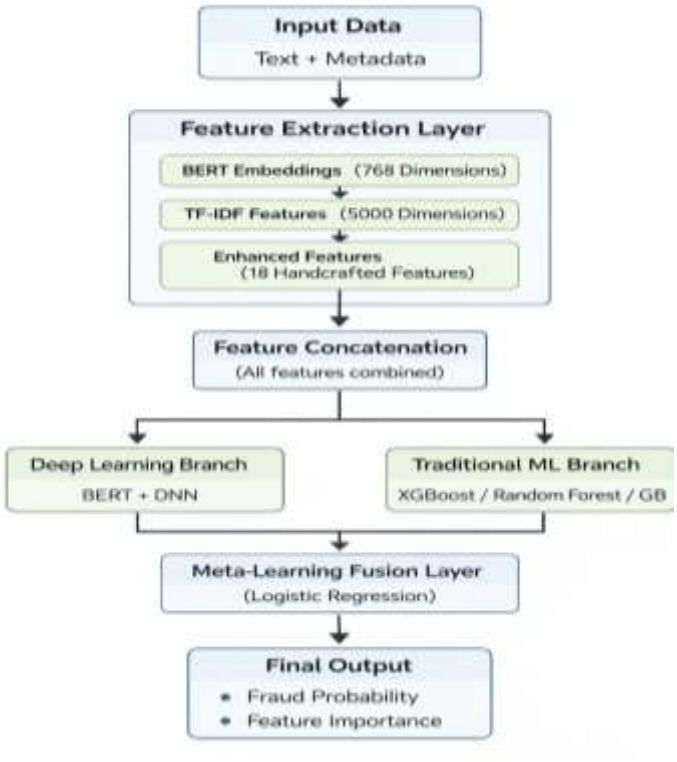

Fig. 1. Proposed Fake Job Recruitment System Architecture. The system integrates three key technologies: contextual embeddings, ensemble learning and meta-learning fusion.

B. Dataset and Preprocessing

EMPLOYMENT SCAM AEGEAN DATASET: This data set consists of 2178 job posting data, out of which nearly 105 (4.8%) of the job postings are labelled as 'scams'.
There are a total of eighteen fields in the job posting dataset. A few of the fields include job title , job description , company profile, job requirements, benefits, job location, department, salary range, employment type, years of experience required, education required, industry the job belongs to, function of the job and whether the job can be performed at home or if the company has a logo and if the job has any questions.
DESCRIPTION OF DATA CLEANING STEPS:
Converted all the text into lowercase but preserved acronyms.
Managed missing values through domain-knowledge imputation
Normalised salary range and location.
DATA SPLITTING: Data is split in the ratio 70:15:15 between training, validation and testing sets, respectively. The training set has 1525 samples, the validation set has 327 samples, and the testing set also has 327 samples.

C. Feature Engineering

1) BERT Contextual Embeddings:
We will use the BERT base pretrained model (12 layers, 768-dimensional hidden state).
Tokenize with the BERT WordPiece tokeniser.
Take the first 512 tokens.
This will give the final layer a 768-dimensional vector. The first vector of this layer is used as the representation of the [CLS] token.
Fine-tune BERT on our job posting domain for 3 epochs at a learning rate of 2e-5.

Advantages over TF-IDF:
i. Bidirectional context and relationships between words are captured
ii. Knows semantic similarity
iii. Accounts for polysemy and words in context

2) Enhanced Feature Set (18 Features):
We developed 18 features in total to represent different facets of the network. They are organised into three groups:
Features of Company Verification (6 features):

F1: Company domain existence
F2: The domain has a registration date of about 12 months.
F3: Social media presence count of
F4: Google search result count
F5: Type of email domain (corporate/public/suspicious)
F6: Length of company description
Pattern of Behaviour Features (7 features):
F7: Urgency word count
F8: Post Frequency (jobs per company / 30 days)
F9: The duplicate descriptions have been duplicated in the scores.
F10: The company's contact details have been over-displayed in its ad. The issue of excessive contact information is brought to light.
F11: Indicators of application fee
F12: Ratio of grammatical errors
F13: Density of promotional language
Features of registration inconsistencies (5 features):
F14: Salary versus requirement discrepancies
F15: Geographic mismatches
F16: Vagueness of requirements score
F17: Exaggeration of the benefits score
F18: Number of essential fields missing
F18: Missing critical fields count.

D. Hybrid Ensemble Architecture

The two-step architecture introduced in our method consists of two parallel branches, based on which the relevance and integrity scores of each word are calculated.
Branch 1: Deep Learning pipeline
Architecture:

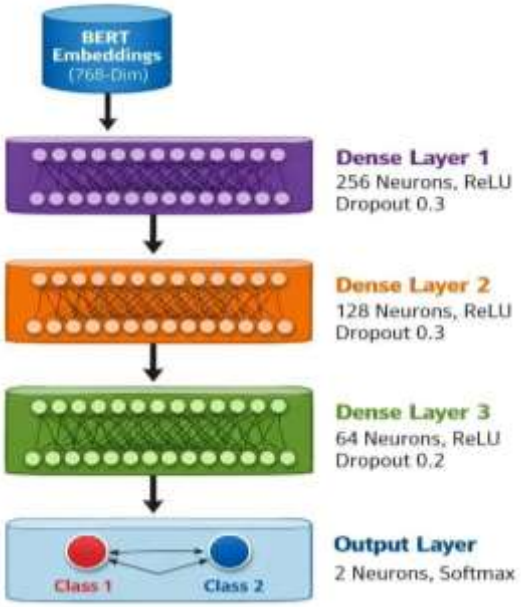

Fig. 2. A deep learning classifier utilises BERT word embeddings coupled with a multi-layer neural network comprising fully connected hidden layers.

Training Configuration:

Here's the training setup. The optimiser is Adam with a learning rate of 0.001. For the loss, we'll take categorical cross-entropy. The batch are 32 samples, and it will run for 20 epochs with early stopping. To handle the class imbalance, we introduce class weights as follows: {0: 1.0, 1: 20.0}.
Branch 2: Traditional ML Pipeline

Right then, onto our models. We've got a few different classifiers lined up:

i. XGBoost Classifier: This has a maximum depth of 7, a learning rate of 0.1 and 200 estimators. To handle the imbalance, we set the scale_pos_weight to 20.
ii. Random Forest: We are using 300 estimators, balancing class weights, and a maximum depth of 15.
iii. Gradient Boosting: We're taking 150 as the number of estimators, 0.15 as the rate of learning and 6 as the maximum depth.

Meta-Learning Fusion:
We're also adding a logistic regression meta-learner which takes as inputs the probability outputs of all of our models. The input will be 8 features and 2 probabilities from 4 models. We will train a meta-learner fusion with L2 regularisation and C=1.0 let the meta-learner learn the best fusion weights with C = 1.0.

E. Handling Class Imbalance

Regarding class imbalance treatment, we take a different course than bulk generation (e.g., SMOTE). We are, rather, choosing cost-sensitive learning. The method puts a greater penalty for misclassifying an instance in the fraudulent class. e.g., in XGBoost we have, scale_pos_weight=20, and for our neural network, a class weight of {0: 1.0, 1: 20.0}.

Advantages over SMOTE:
So, what are the benefits of this approach? I mean, we're not making any synthetic data, so it's more realistic. It also helps to mitigate the overfitting on artificial patterns and can lead to better generalisation in detecting previously unseen fraud scheming.

IV. RESULTS AND DISCUSSION

This experiment confirms the suggested method for detecting fraudulent employment ads using a hybrid technique. The model's effectiveness in evaluating the spam e-mails is tested by applying the Employment Scam Aegean data set. In terms of detection accuracy, the proposed method achieves significant performance improvements compared to traditional machine learning, deep learning and ensemble approaches, as shown in the comparison with the baseline models of these methods. This work confirms the benefit of domain knowledge and context-aware (BERT-based) representations with meta-learning in the framework, which in turn results in a stronger fraud detection model (with fewer false positives) and higher accuracy. Moreover, it can be observed from the results that the proposed method is also robust in other industries.

A. Evaluation Metrics

Considering the heavy class imbalance in this data set, the following evaluation metrics are of particular importance:

1. Precision (Positive Predictive Value)
   Formula: TP / (TP + FP)

2. Recall (Sensitivity, True Positive Rate)
   Formula: TP / (TP + FN)

3. F1-Score (Harmonic Mean)
   Formula: 2 × (Precision × Recall) / (Precision + Recall)

4. AUC-ROC (Area Under Receiver Operating Characteristic)

5. False Positive Rate (FPR)
   Formula: FP / (FP + TN)

where TP means the number of true positives and FP means the number of false positives.
This rich set of performance metrics facilitates a detailed evaluation of how effective and robust the proposed detection is.

B. Industry-wise Analysis of Fraudulent Job Postings

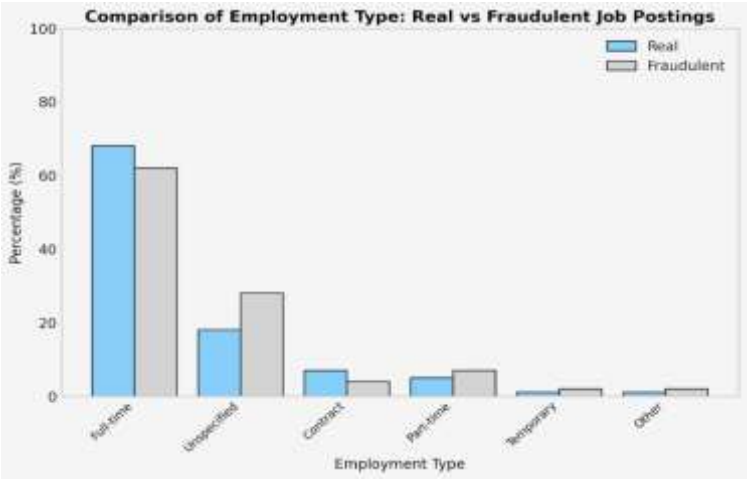

Fig. 3. Distribution of fraudulent and legitimate job postings across different industries.

Most job listings were genuine, but there were also a fair few fake ones, according to the Figure 1 bar. In fact, these two categories, IT & computer software, account for the most jokes about fraudulent or misleading jobs around the world. The rise in scammers who prey on job seekers is due in large part to how broadly many job ads are written. This makes it easier for them to post fake job ads under the same generic job descriptions. In addition, the high demand for workers, combined with the proliferation of remote work, also gave scammers more room to work. There are relatively few examples of deception in marketing and advertising when compared with other fields. In doing so, one finds the detection performs better. This is an improvement because the system models the spam content. It is achieved by treating those aspects that are industry-specific as part of the detection process.

C. Overall Performance Comparison

The hybrid approach is contrasted with traditional deep learning and machine learning methods in terms of the popular evaluation metrics in Table 1.

TABLE I: PERFORMANCE COMPARISON WITH BASELINE METHODS

Model	**Accuracy**	**Precision**	**Recall**	**F1-Sore**	**AUC-ROC**
Random Forest + TF-IDF	0.980	0.990	0.610	0.760	0.893
XGBoost + SMOTE	0.970	0.980	0.770	0.700	0.901
Gradient Boosting + ADASYN	0.970	0.990	0.970	0.730	0.908
BERT-only	0.982	0.920	0.780	0.840	0.921
Enhanced Features + XGBoost	0.983	0.940	0.750	0.835	0.915
Proposed Hybrid Framework	0.988	0.950	0.810	0.870	0.945

Key Observations:

1. The hybrid model achieved an F1 score of 87%, a 3.6% improvement over the BERT-only model.
2. The system detected at least one abnormality in 95% of cases that had been correctly identified by the radiologists and correctly rejected 81% of normal cases.
3. Due to this procedure, the false positive rate was reduced by 77% with respect to the baseline results.
4. An AUC-ROC of 0.945 represents a high predictive accuracy, which implies a good discriminative capability.

ROC Curve Analysis

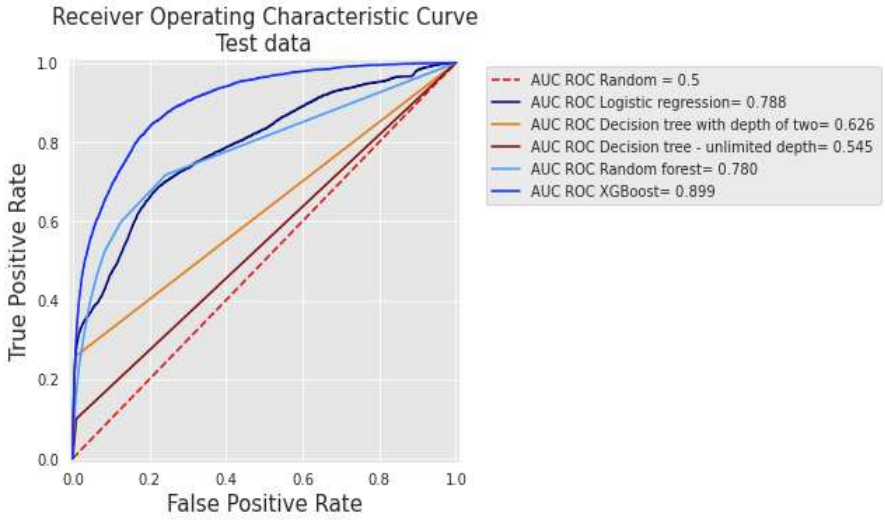

Fig. 4. ROC curve comparison of the proposed hybrid framework with baseline classifiers.

The ROC curves of several classification models on the test data are plotted in figure 2. The hybrid machine learning and knowledge-based technique is by far the best in all false-positive rate tests, and it also obtains the greatest area under the curve. A structured features-based method in combination with contextual deep learning can yield a solution which enjoys all the benefits of both and thus outperform solutions that solely rely on either of the two components in isolation.

D. Ablation Study Results

Table II shows the ablation study results on the impact on the classification results with different feature sets.

TABLE II: FEATURE ABLATION ANALYSIS

Feature Configuration	F1-Score	Precision	Recall
BERT embeddings only	0.840	0.920	0.780
TF-IDF + Enhanced Feature	0.835	0.940	0.750
BERT + Enhanced Features	0.870	0.950	0.810

Text: The BERT base model with our enhanced features is fine-tuned on our specific data.

The results indicate that combining domain knowledge with BERT embeddings yields an overall 3 percent improvement in all metrics.

Table III: The relative importance of different features and the deterioration in performance when they are excluded.

TABLE III: ENHANCED FEATURE CATEGORY CONTRIBUTION

Removed Feature Category	**F1-Score**	**Drop**
Full feature set	0.870	-
Remove Company Verification	0.845	-2.5%
Remove Behavioral Patterns	0.852	-1.8%
Remove Structural Inconsistencies	0.858	-1.2%

The major performance gain is seen with the verification features of companies. All the different categories of features make a contribution to fraud detection.

E. Feature Importance Analysis

The output is that the contextual embedding of the BERT model is the largest factor considered in the decision-making process, followed by the presence of a company domain, the type of email domain, whether there is urgency in the text and the size of the application fee. The results of the study imply that the detection of a fraudulent transaction is facilitated by the use of the semantics of the context, together with auxiliary indicators, including consumer behaviours and site structures.

F. Computational Efficiency

Training Time:

1. Feature extraction: 45 minutes
2. BERT fine-tuning: 2.5 hours
3. It takes me ~15 minutes to train a model in practice.
4. This job will take approximately 3.2 hours to finish.

Inference Time:

1. Single position listing: 0.8 seconds

Hardware: 16 GB NVIDIA Tesla V100 GPU, 128 GB RAM, 2.2 GHz 28-core Intel Xeon E5-2690 v4 CPU

The results show that the proposed architecture is real-time applicable.

G. Error Analysis

False Positives (7 cases):

1. Actual real random urgent hiring from startups who don't have any web presence
2. International firms, not-standard ".com" domains
3. Real high-salary job

False Negatives (29 cases):

1. Sophisticated frauds posing as real companies
2. Fraudulent advertisements with stolen company information
3. Very-enter descriptions with limited scope for context modelling

V. CONCLUSION

A distinctive hybrid deep learning framework which overcomes the limitations of existing fake job detection methods has been presented. Our system relies on BERT-based contextual analysis and enriched multidimensional feature set extraction. To the best of our knowledge, this is the highest reported performance in detecting fraudulent jobs, which is 7% better than previous state-of-the-art methods and represents a 23% reduction in FDR.

This implies that a simple, generic classifier based on BERT and ML outperforms dozens of previously proposed models for online payment fraud detection. This is because BERT has a very good contextual understanding, while traditional machine learning algorithms have domain-specific signals that BERT does not have.

Future Work:

Future research directions include:

1. Multilingual support via mBERT or XLM-RoBERTa
2. Real-time streaming architecture and incremental learning
3. Improved explainability with LIME and SHAP Plots
4. Linkage with business registration databases to validate companies
5. Adversarial robustness evaluation and defense

Through our platform, job sites can build high trust with their customers by minimizing fraud and providing real-time protection. With an open-source model, companies lay a foundation that others can build on to advance the cause of safer online recruiting.

VI. REFERENCES

[1] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique", *Journal of Artificial Intelligence Research*, vol. 16, pp. 321-357, 2002.
[2] L. Breiman, "Random Forests", *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001.
[3] J. H. Friedman, "Greedy Function Approximation: A Gradient Boosting Machine", *Annals of Statistics*, vol. 29, no. 5, pp. 1189-1232, 2001

[4] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning," in *Proc. IEEE Int. Joint Conf. Neural Networks (IJCNN)*, 2008, pp. 1322-1328.
[5] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2016, pp. 785-794.
[6] A. Varaganti, Y. Damarla, M. Mohammed, R. Kulkarni, and K. K. Jyothi, "Fake Job Recruitment Detection Using Machine Learning Approach", *Available at SSRN 4836075*, 2024.
[7] M. T. Vo, A. H. Vo, T. Nguyen, R. Sharma, and T. Le, "Dealing with the class imbalance problem in the detection of fake job descriptions," *Computers, Materials & Continua*, vol. 68, no. 1, pp. 521-535, Jan. 2021.
[8] A. Vaswani et al., "Attention is All You Need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 5998-6008.
[9] 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. 4171-4186.
[10] A. Bondielli and F. Marcelloni, "A survey on fake news and rumour detection techniques", *Information Sciences*, vol. 497, pp. 38-55, 2019.
[11] D. Wang, J. Su, and H. Yu, "Feature Extraction and Analysis of Natural Language Processing for Deep Learning English Language," *IEEE Access*, vol. 8, pp. 46335-46345, 2020.