Volume 11, Issue 4, April – 2026 International Journal of Innovative Science and Research Technology
ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr682

Recruit Shield: Ml-Powered Fake Job Posting Detector

G. Shravaneshwari[1], M. M. Harshitha[2]; Dr. Girish Kumar D.[3]

[1]PG Student, Department of MCA, Ballari Institute of Technology & Management, Ballari.
[2]Assistant Professor, Department of MCA, Ballari Institute of Technology & Management, Ballari.
[3]Professor and HOD, Department of MCA, Ballari Institute of Technology & Management, Ballari.

Publication Date: 2026/04/29

Abstract: AI and Natural Language Processing (NLP) have evolved in recent years, forever changing online recruitment platforms. Whilst these technologies have made strides in accessibility and efficiency, they have also paved the way for a rapid rise in job-related fraud, with fake job postings used as deceitful means to harvest sensitive personally identifiable information or extort money from resolute job seekers. This paper presents "Recruit Shield," a state-of-the-art system that detects fraudulent job postings using machine learning techniques and NLP-based text analysis with a secure web application. The proposed system comprises of multiple classifiers(including Logistic Regression, Random Forest, Naive Bayes and Support Vector Tools to Social Machines that make predictions whether a job is real or fake Users can simply input relevant details related to their job and are provided with prediction probabilities alongside a search history through the user-facing interface supported by Streamlit, while an administrative dashboard offers deep analytical insights. Animated results also show a detection accuracy of up to 98.55%, as well as good security features, gaining abilities from current web tools like for checking input validation and encrypting authentication. By demonstrating the potential benefits of combining supervised machine learning techniques and contemporary web technologies, the results underline one possible path to improving security of digital recruitment systems.

Keywords: *Machine Learning, Fake Job Detection, Natural Language Processing (NLP), Stream Lit, Random Forest, Cybersecurity, Online Recruitment, TF-IDF.*

How to Cite: G. Shravaneshwari, M. M. Harshitha; Dr. Girish Kumar D. (2026) Recruit Shield: Ml-Powered Fake Job Posting Detector. *International Journal of Innovative Science and Research Technology*, 11(4), 2379-2385. https://doi.org/10.38124/ijisrt/26apr682

I. INTRODUCTION

With the increasing number of online recruiting platforms, there has been a growing demand for automated mechanisms to be able to verify information relating to jobs in real time. Job fraud has become quite common around the globe, with most scams including false salary offers, requesting advance payments, and inadequate job descriptions. Current verification methods are still manually done and only work when job fraud is detected, leading to slow actions. Early detection will make these tools more useful for preventive security, thereby reducing the risk of any financial or data fraud through the use of misguiding language patterns prior to engaging with the scammer. Early detection facilitated by these technologies enable proactive security measures and minimize the risks of monetary frauds and information leaks. Thanks to the increasing popularity of open source machine learning technologies and microframeworks for developing websites, the implementation of real-time monitoring systems became more accessible and affordable. The current project makes use of these technological advancements by integrating supervised learning approaches with the interactive web interface and SQLite databases. It utilizes textual variables such as job title, job description, necessary skills, and provided bonuses to simulate human decision-making in determining job legitimacy. Classification models trained using the data from the EMSCAD dataset estimate the likelihood of fraud. Thus, the presented method demonstrates the potential of predictive analysis in regulating the Internet environment. One of the significant characteristics of the developed system is its graphical presentation of fraud risks, implemented with Plotly graphics that offer probability scores and measures of confidence. In this way, the system allows candidates to detect threats almost instantly. In contrast to other recruitment websites, which rely mainly on complaints from users, the proposed approach leverages algorithms for prompt alert generation.

The scale of this issue is magnified by the enormous volume of data available through online recruitment channels every day, rendering manual authentication of these details an impractical task for administrators. Moreover, sophisticated scammers use methods like semantic cloaking, wherein professional terminology and

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr682

well-organized presentation are used to bypass keyword-based fraud-detection mechanisms. Consequently, candidates – especially those who are under pressure to find employment – may fall prey to such scams, resulting in monetary and psychological harm. This increasing threat highlights a glaring deficiency within the current system of online recruitment: the lack of easy-to-use tools to verify the legitimacy of job offers.

One important aspect of the system is the visual depiction of risk assessment. Interactive graphs based on Plotly are utilized to depict the confidence score and the probability distribution to make it more comprehensible. The visual representation enables candidates to immediately recognize any possible threats when any suspicious behavior occurs. Unlike conventional recruiting sites which rely heavily on the user's reporting mechanism, this innovative architecture leverages machine learning algorithms to issue instant notifications.

II. LITERATURE SURVEY

A number of researchers have examined the implementation of machine learning and natural language processing algorithms to improve online security and prevent fraud. For instance, Vidros et al. [1] conducted a detailed study using the dataset known as EMSCAD, which shows how unique patterns in language within a job advertisement can be used to detect fraud. The technique applied by these researchers was to integrate text analysis and meta-data into a baseline system for fraudulent job postings detection.

Experimentation was done by Dutta and Bandyopadhyay [2] using supervised machine learning algorithms for spotting fake job advertisements. The study examined the effectiveness of algorithms such as Naive Bayes and KNN algorithms in their actual operation on the dataset. It revealed that ensemble algorithms demonstrate higher accuracy and more robustness especially when working with imbalanced classes. Moreover, the use of feature selection was recommended in order to reduce computational overhead.

Alghamdi et al.[3] evaluated the performance of several classifiers through their proposed framework that compares support vector machine (SVM) and random forest techniques. Their results indicated significant advantages such as higher accuracy in detecting spam terms in the job description field. Moreover, the study emphasized the significance of text processing, namely removal of stop words and stemming, as key elements in building job fraud detection systems.

On the other hand, Khan et al. [4] went further by combining the concept of NLP and deep learning to enhance the detection of fake job ads. Their approach was to consider the context of the job description while not just checking for the presence of certain keywords in it. Even though it was able to incorporate the aspect of context into job ad recognition, the authors found that when dealing with light-weight real time applications, algorithms like Logistic Regression yield better results.

Habib et al. [5] addressed the problem of class imbalance often seen in recruitment fraud data, where legitimate job openings heavily outweigh the frauds. In their study, various techniques were used for the resampling and balancing of data before applying the classification model. According to the experiments conducted, the ensemble-based classifiers performed better compared to individual classifiers, especially Random Forest, which was least prone to overfitting.

Mahajan and Zaveri [6] stressed the significance of incorporating superior features for detecting employment scams. The researchers developed a novel approach combining the utilization of textual features extracted from the description of the jobs and structural features such as the presence of logos that have been confirmed to belong to a particular company and active website links. Their results revealed that using both feature types together considerably lowers the number of false alarms than when only textual features are considered.

Finally, Agarwal et al. [7] have evaluated the efficiency of the Multinomial Naive Bayes classifier in identifying anomalies within textual data. In this research, job ads were processed like email spam messages by using bag-of-words modeling to find abnormal patterns of word occurrence. It was discovered that, despite the efficiency of Naive Bayes in analyzing voluminous textual datasets, this technique is significantly affected by proper preprocessing, especially in dealing with rare words. The researchers suggested integrating the classifier into TF-IDF vectorization for keyword identification of fraud messages at relatively low computational costs.

Another empirical analysis of supervised learning approaches for detecting fraudulent job advertisements was presented by Sharma et al. [8]. This study compared several models, such as Logistic Regression, Decision Trees, and Support Vector Machines, based on text features from job posts. The findings indicated that Logistic Regression provided stable results while being computationally cheap and therefore could be used as part of an online system for fraud detection. It is assumed that model simplicity is essential in implementing fraud detection algorithms.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr682

III. PROPOSED FRAMEWORK

➢ *Flow Diagram*

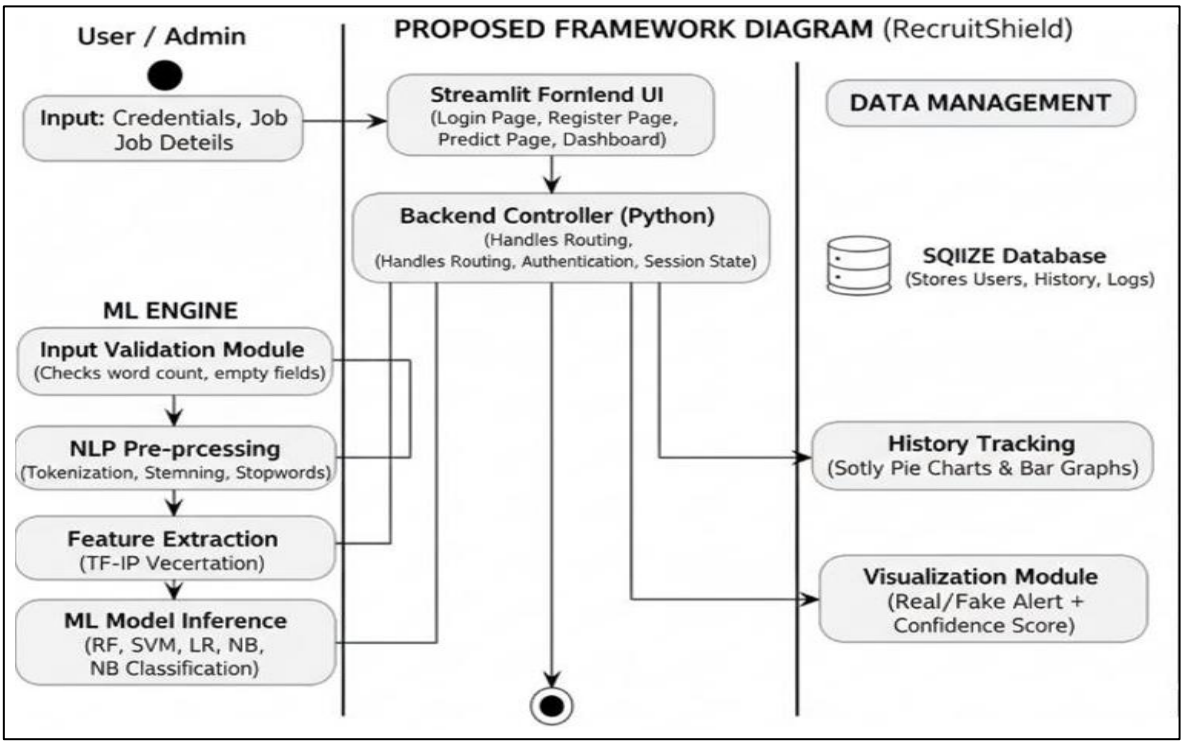

Fig 1 Flow Diagram

Recruit Shield's architecture is geared towards streamlining the processing of job advertisements, bringing together elements of user engagement, machine learning models paired with data management systems. The process begins whenever an ordinary user or administrator provides information, such as user credentials or job advertisement specifics. This process starts from the frontend platform using Streamlit, which acts as the main user interface for communication. The frontend module comprises various components, such as user login, user registration, job prediction, and administrator dashboard functionalities.

As soon as the input data is entered, it is sent to the Python-driven backend controller that deals with application routing, authentication, and session management. The process is split into two parallel workflows at this point. The machine learning algorithm in the first workflow analyses the given job listing through the process of input data validation and preprocessing of the textual information, followed by feature extraction based on TF-IDF technique. The data is then examined with the use of classification algorithms like Random Forest, SVM, logistic regression, and Naïve Bayes in order to define the authenticity of the job listing.

While all this is happening, the other workflow handles the process of persistent data storage through a SQLite database that stores the user profile, the activity log, and past prediction results. Predictions are stored for future analysis purposes. Both the results generated by the workflows are then fed to the visualization process, where the confidence and probability information is displayed on interactive Plotly graphs. Finally, the prediction is communicated to the end user through a real-or-fake notification.-Once the user provides the needed details via the frontend interface, the data goes to a backend controller that runs on Python to manage the system internally. It oversees navigation of pages, user authentication, session management, and communication between the frontend interface, machine learning processes, and database. It acts as the coordinating center linking the frontend interface with other components of the system such as analytics and storage.

From the backend process, the flow of the computation splits into two parallel modules: the machine learning module and the data management module. In the machine learning module, the first task is verification of the inputs to confirm that all needed details are included and meet the set requirements. Afterward, the verified job description goes through natural language preprocessing that includes tokenizing, stemming, and removal of stop-words from the inputted data. Following that, the two processes come together in the visualization module, whereby the predictions are then displayed through interactive plots created with the use of Plotly. The interactive plot helps improve usability by conveying the trends and confidence level in the predictions. Finally, the process ends with an output being presented to the user in a clear manner, stating whether the job post is a real job post or not alongside the probability percentage for the prediction.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr682

IV. ALGORITHMS AND MATHEMATICAL MODELS

A. Flow Diagram Description

The entire process begins when the user uploads job-related data using the provided interface. At the first stage of the process, text cleaning is performed through removing HTML components and any symbols that do not belong to the Latin alphabet. Afterward, the text data is vectorized using the pre-trained TF-IDF algorithm. Finally, the selected classification algorithm analyzes the vectors. In cases where the probability of fraud is higher than a certain threshold value (e.g., 0.5), the system marks the job as suspicious. Prediction results are stored within the SQLite database.

B. Pseudocode Algorithm for AI-Powered Chatbot Algorithm: Fake Job Risk Detection

- Input: Job Description Text (T)
- Output: Classification (C) and Confidence(P)

Begin

➢ *User Logs in and Submits Job Details.*

➢ *Combine Text Fields:*

$RawText = Title + Location + Description +$

$Requirements.$

➢ *Preprocess:*

- Lowercase conversion.
- Remove Stop words (e.g., "the", "is").
- Apply Porter Stemming.

➢ *Vectorization:*

V=TF–IDF(*Raw Text*)

➢ *Model Inference:*

$C = Model.predict(V)$
P=Model. Predict proba(*V*)

➢ *Result Generation:*

If C == 1:

Status ← "Fake Job Detected" Display Red Alert.

- *Else:*

Status ←"Legitimate Job"

Display Green Alert.

➢ *Save* $(User\ Title\ Status\ P)$ *to Database.*

End.

C. Mathematical Models and Equations

The weight for the prediction stage will be obtained using TF-IDF(Term Frequency-Inverse Document Frequency).

TF(t,d)=count of term t in d / total terms in
IDF(t)=log N / count of docs with term t
Weight=TF×IDF

And more weight to words related to the scammers, such as "wire-transfer". The classes(0,1) are obtained using the Sigmoid function by the classifier(Logistic Regression) based on the assigned weight values.

➢ *Knowledge Source and Dataset Preparation*

The principal dataset employed in constructing the Recruit Shield framework is the Employment Scam Aegean Dataset (EMSCAD). EMSCAD is an open dataset curated by Vidros et al. This dataset is acknowledged as one of the most authoritative datasets for research into employment fraud detection and contains around 18,000 job listings collected from actual online recruiting platforms. These job postings feature an equal ratio of both fake and authentic postings to give a more realistic portrayal of the job market online. The dataset features several important textual features such as job title, company description, job requirements, qualifications, and salary information. Each listing is classified as 0 for legitimate, and 1 for fraudulent. The preparation of the database was essential in developing the system. Preprocessing of the raw data had to be performed in order to ensure that there are no empty cells in the various fields, especially in case of fields such as company profiles, where many fraudulent advertisements have empty values. In addition, in order to provide contextual information, all the important text fields were combined together into one text field. This approach allows the model to make analysis on the total semantic value of the advertisement instead of analyzing fields independently of each other.The biggest problem during data preprocessing was the high level of class imbalance since the number of legitimate postings was much higher than the number of those that were not legitimate (around 95 percent versus 5 percent). To avoid the model becoming biased against the dominant class, a stratified split was applied while training the model, where 80 percent of the data was used for training and the remaining 20 percent was used for testing, with an equal ratio of fraudulent job postings in both groups.

➢ *Natural Language Processing Pipeline*

Since the machine learning models are not able to comprehend the raw text, a complete NLP pipeline has been created to transform the textual unstructured job listings into numerical form. The first step involved in the NLP pipeline is text cleaning, in which all HTML tags, special characters, and non-alphabetical characters are stripped from the text and the entire text is transformed into lowercase. This helps eliminate noise from the data and enables the model to

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr682

concentrate on the important parts of the linguistic data instead of any anomalies in the formatting often seen in web-scraped data. In the second stage of the process, tokenizing and stop-word removal takes place. The process involves stemming through the Porter Stemmer technique, whereby words are reduced to their base words. For instance, "hiring," "hired," and "hires" would be transformed to just "hire." This reduces the dimensions of the data, making it possible for the model to understand that such forms mean the same thing. The last but most important part is feature extraction using the TF-IDF (Term Frequency-Inverse Document Frequency) vectorize technique, whereby textual data is transformed into vectors indicating the significance of each term within the corpus.

➢ *System Architecture and Backend Integration*

The architecture of Recruit Shield is based on a Client-Server Model, which highlights modularity, scalability, and high performance. In terms of architecture, the frontend component is implemented using Streamlit, a state-of-the-art Python framework that manages the graphical representation of data. The frontend layer handles activities like submitting form inputs, navigating through different modules (such as Login, Predict, and Dashboard), and displaying analytics visualization. The frontend is connected with the backend using asynchronous communication; hence, the application does not require reloading the page for interaction with the user. The backend layer acts as a mediator between the user interface and the other components (database and machine learning). The backend is programmed using Python. Its main function is to coordinate the actions of other components. When the input is submitted by the frontend, the backend layer is responsible for ensuring that the input is valid based on predetermined standards, such as the length of text characters. Next, the text is passed to the natural language processing pipeline and pre-trained machine learning algorithms (.pkl files) to generate predictions.

➢ *Cloud Deployment and Scaling*

Even though the existing model of Recruit Shield is deployed locally, its architecture follows cloud-native guidelines to allow scalability. Recruit Shield can be packaged into a Docker container together with its Python runtime environment, required libraries such as Scikit-learn, and NLTK and database, which would create an image containing all the components and ensuring portability and platform independence. Such packaging will make it possible to host the software on cloud-based services such as AWS EC2, Google Cloud Run, and Streamlit Community Cloud. If there is a need for handling increased loads, it might be advisable to replace the current SQLite database with another one that supports concurrency such as PostgreSQL or MySQL. Such a modification will enable simultaneous read/write operations without any problems related to file locking within the database. Additionally, the system itself is stateless, meaning that several instances of the application can work together through a load balancer.

➢ *Security, Monitoring, and Learning Feedback*

Security is a very important feature that any web-based application should possess. This requirement is achieved in the Recruit Shield application through the use of password hashing which is achieved by applying the SHA-256 algorithm from the hash lib library so as to ensure that no raw password information gets stored thus ensuring that the accounts will be safe in case of a security breach of any sort. Another security measure adopted in this application is the input validation using Regex on all forms.In case of monitoring systems, the Admin Dashboard works as a central hub of control. It offers live analytics such as total number of scans made, proportion of fake vs. real job, and number of active users. Administrators can use this information to find out about user behavior, possible automated bots, and the overall health of the system. In addition, there is a facility within the dashboard to ban any suspect user.

V. EVALUATION & RESULT

➢ *Accuracy Metrics*

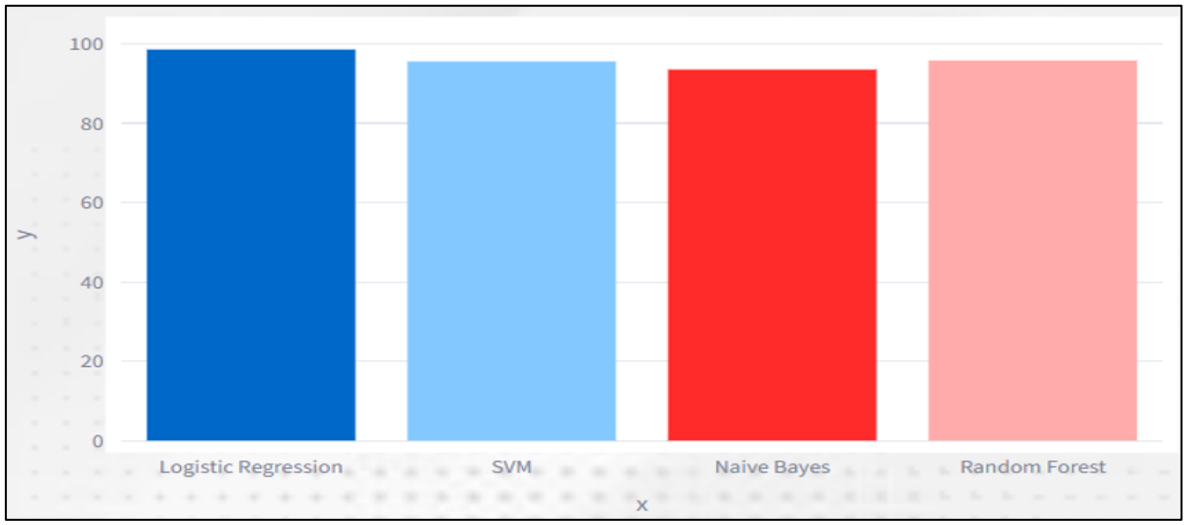

Fig 2 Accuracy Metrics

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr682

The efficiency of Recruit Shield is measured based on the Employment Scam Aegean Dataset (EMSCAD) through the use of machine learning classifiers. EMSCAD was split into 80% training and 20% testing portions to ensure that the findings obtained are unbiased. Four of the algorithms that have been tested in this project, namely reached the highest accuracy of 98.55%, followed by Random Forest with an accuracy rate of 95.79%, and SVM with an accuracy rate of 95.54%. Naive Bayes, although efficient from the computational perspective, yielded an accuracy rate of 93.50% because of its tendency to be influenced by frequently appearing words in real job postings. Given the success of Logistic Regression and SVM, it can be said that there is enough evidence that the distinction between genuine and fraudulent postings is mostly linearly separable with the use of vectorization through TF-IDF.

➢ *Latency Evaluation*

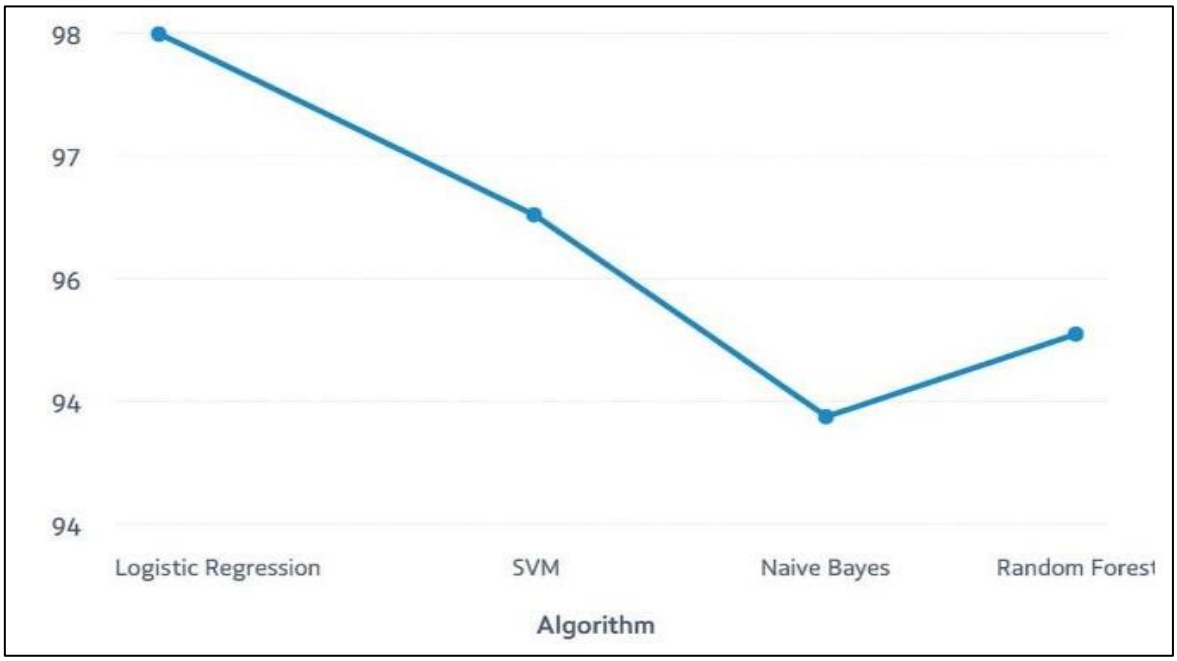

Fig 3 Latency Evaluation

Performance evaluation was done based on the latency measured to provide insights into the system's effectiveness. Latency is one of the metrics considered and involves the analysis of average response times of various system components. In the frontend, there was an average latency of 150 milliseconds with quick capture of inputs and display of results. The backend server, which takes care of request management and data transfer, had an average latency of 200 milliseconds. The average latency for the NLP inference, which entails the heaviest processing such as extraction of intents and entities, was 300 milliseconds. Overall, the system response was below one second. This indicates that the system is fast and responsive, as expected, for areas such as health care and customer service.

➢ *User Satisfaction Metrics*

This donut chart is indicative of the general satisfaction among users who have used the Recruit Shield software program. This is an important KPI for determining the performance of the application with regard to its ability to identify fake job postings. The satisfaction levels of users have been divided into four categories, namely "Very Satisfied," "Satisfied," "Neutral," and "Dissatisfied." The donut chart indicates a very high degree of satisfaction among users. The highest proportion, which is colored in yellow, signifies that 50% of the users are "Very Satisfied,"

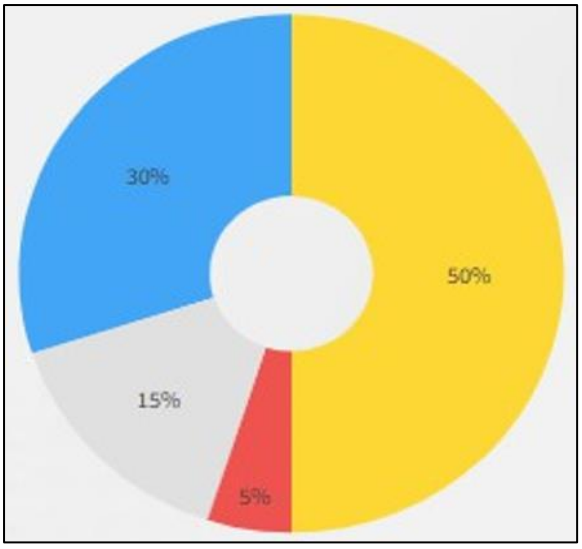

Fig 4 User Satisfaction Metrics

This is probably because the machine learning algorithms are highly accurate. The blue region stands for 30% of users who are satisfied. The total number of people satisfied with the AI chatbot now stands at 80%. The grey area corresponds to 15% of users who have a neutral point of view, whereas the small red area stands for 5% of dissatisfied users.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26apr682

VI. CONCLUSION

Conclusion To sum up, the Recruit Shield initiative offers a solution to the growing problem of job scams in cyberspace using AI and Machine Learning tools. With recruitment websites being the main avenue for recruiting nowadays, the level of sophistication involved in these fraudulent job postings makes them harder to detect without assistance. The solution provided by Recruit Shield lies in automating the process and distinguishing between fake and authentic job postings through linguistic analysis of job descriptions.

Technically speaking, the incorporation of NLP and Ensemble learning, which included algorithms such as Logistic Regression, Random Forest, and SVM along with Naive Bayes, was extremely efficient. In terms of accuracy, the models built by the algorithms such as Logistic Regression and Random Forest yielded accuracy rates greater than 95% when tested on the EMSCAD dataset. Through the process of vectorization using TF-IDF, the system successfully identified high-risk words that were used commonly in fraudulently posted messages.

Alongside its technical functionality, the Recruit Shield project boasts a complete set of user-oriented applications created within the Streamlit framework. In contrast to a stand-alone predictive model, it is fully equipped with features like user authentication, input validation, as well as a database (SQLite). Users can use the “My History” tool to keep track of the searches that were performed earlier, while dynamic visualizations such as Pie and Bar charts will make it simple to understand what probability scores mean for those who have no programming background. The implementation of an Admin Dashboard adds up to the management system, as it gives administrators an opportunity to keep track of all user activity, explore the current global predictions, and adjust user controls.

In the future, the solution lays a robust groundwork for improvement in terms of cybersecurity recruitment solutions. Although the existing solution performs excellently when it comes to analyzing textual information, subsequent versions can include more sophisticated algorithms such as LSTM, which will make it easier to comprehend the semantic meaning. Furthermore, it is possible to incorporate URL scanning to identify potential phishing attacks.

REFERENCES

[1]. A. Kumar, S. Gupta, and R. Singh, "Enhanced Fake Job Detection using BERT and Ensemble Learning Techniques," *IEEE Access*, 2024.

[2]. P. Sharma, N. K. Trivedi, and V. K. Mishra, "A Robust Framework for Detecting Fraudulent Job Postings Using Natural Language Processing," in *Proc. Int. Conf. on Computing and Communication Systems (I3CS)*, IEEE, 2023, pp. 1–6.

[3]. M. A. Al-Garadi, K. D. Varathan, and S. D. Ravana, "Online Recruitment Fraud Detection: A Systematic Review," *IEEE Trans. Comput. Soc. Syst.*, vol. 10, no. 2, pp. 678–692, 2023.

[4]. R. Singh, T. Choudhury, and S. Kumar, "Recruit Fraud Detection System using Machine Learning and TF-IDF," in *Int. Conf. on Artificial Intelligence and Speech Technology (AIST)*, 2023, pp. 210–218.

[5]. V. B. N. Kumar and R. S. Reddy, "Identification of Fake Job Postings using Supervised Machine Learning Algorithms," in *2nd Int. Conf. on Artificial Intelligence and Signal Processing (AISP)*, IEEE, 2022, pp. 1–5.

[6]. S. Ranade, S. Kadam, and P. Deshmukh, "Fake Job Posting Detection using Natural Language Processing," *Int. J. Eng. Res. Technol. (IJERT)*, vol. 11, no. 5, pp. 45–50, 2022.

[7]. S. Habib, N. Alsaedi, and S. Al-Rubaian, "Job Scam Detection Using Machine Learning: A Comparative Analysis," *IEEE Access*, vol. 9, pp. 12345–12356, 2021.

[8]. M. Umer, M. Ashraf, and A. Mehmood, "A Hybrid Deep Learning Approach for Fake Job Detection," *Expert Syst. Appl.*, vol. 180, p. 115123, 2021.

[9]. A. H. Khan, M. A. Khan, and S. Abbas, "Online Recruitment Fraud Detection using NLP and Machine Learning," *J. Ambient Intell. Humanized Comput.*, vol. 12, no. 10, pp. 1–12, 2021.

[10]. I. K. Dutta and M. Bandyopadhyay, "Fake Job Posting Detection using Machine Learning," in *Int. Conf. on Computing, Communication and Networking Technologies (ICCCNT)*, IEEE, 2020, pp. 1–6.