Fake Job Post Prediction

Kambam Bindu[1], Dr. K Santhi Sree[2]
[1]*Student, M. Tech, Department of Information Technology, Jawaharlal Nehru Technological University Hyderabad*
[2]*Professor, Department of Information Technology, Jawaharlal Nehru Technological University Hyderabad*

Abstract - Because of the improvement of web-based entertainment and contemporary advancements, posting job opportunities has turned into a boundless issue in the present society. The issue of fake job posting prediction will consequently be of significant interest to everybody. Predicting fake job postings, like many other contractual issues, poses many challenges. To determine whether a job advertisement is real or fake, the company is expected to use a variety of data mining techniques and classification algorithms, including ANN, decision trees, support vector machines, naive Bayes classifiers, random forest classifiers, multi-layer perceptron, and deep neural networks. We conducted our tests using the Employment Scam Aegean Dataset (EMSCAD), which contains 18,000 examples. The exhibition of the deep neural network classifier is magnificent for this order test. For this DNN classifier, three thick layers were utilized. As far as predicting a fake job post, the prepared classifier has a classification accuracy (DNN) of practically 98%.

Keywords: - Job fraud detection, classification, DNN, KNN, SVM, Naive Bayes, Decision Tree, Random Forest, data mining, EMSCAD dataset.

I. INTRODUCTION

The spread of web-based entertainment and contemporary innovation has radically changed the job publicizing scene lately, simplifying it than any time in recent memory for organizations to post job opportunities and for contender to find positions. In any case, with this advanced turn of events, there is all one stressing aftereffect that has arisen: the multiplication of fake job postings. As well as burning through work searchers' time and cash, these misleading notices convey critical perils, including the chance of wholesale fraud and monetary misfortune. Subsequently, expecting and spotting deceitful work postings has turned into a pivotal endeavor in the fields of cybersecurity and data mining.

The point of this task is to appropriately predict if a job ad is fake or real by using various DM procedures and classification algorithms. We use K-Nearest Neighbor (KNN), Decision Tree, Support Vector Machine (SVM), Naive Bayes, Random Forest classifier, Multi-Layer Perceptron (MLP) and Deep Neural Network (DNN) on the Employment Statistics Aegean Sea Dataset (EMSCAD), which contains 18,000 samples.
The Deep Neural Network (DNN) classifier performs particularly well in this arrangement challenge, making it stand apart among different strategies. With a model engineering comprising of three thick layers, the DNN classifier predicts false work postings with an accuracy of around 98%. This serious level of accuracy features how DL might be utilized to troublesome order issues where recognizing genuine and misleading things might rely upon minute examples in the information.
This work offers significant bits of knowledge into the genuine execution of these methods in true conditions, as well as exhibiting the viability of complex machine learning calculations in the battle against online misrepresentation. Working on the veracity of job postings utilizing such prescient calculations is turning out to be increasingly more significant as the worldwide job market grows carefully, ensuring more secure and more solid communications for job searchers.

II. LITERATURE SURVEY

Fraud detection, especially in web based selecting and news conveyance, has developed more significant in the computerized period. Utilizing a few methodologies from machine learning and deep learning has showed guarantee in recognizing deceitful action. This writing audit examines current advances in fraud detection and message arrangement procedures.

Vidros et al. (2017) investigated the programmed distinguishing proof of internet selecting extortion, zeroing in on highlights and approaches. They gave a public dataset, setting the system for future concentrate in this space [1]. Alghamdi and Alharby (2019) recommended an astute model for web based enlisting extortion recognition, featuring the need of utilizing keen calculations to balance creating false methods [2].

Huynh et al. (2020) explored job prediction using deep neural network models, exhibiting the utility of modern calculations in expecting business designs [3]. Zhang et al. (2020) proposed FAKEDETECTOR, a deep diffusive neural network for successful misleading news ID, giving light on the battle against disinformation [4].

Scanlon and Gerber (2014) explored the mechanized distinguishing proof of digital selecting by vicious fanatics, underlining the need of utilizing innovation to forestall risky web-based action [5]. Kim (2014) created convolutional neural networks (CNNs) for state order, laying the structure for future text arrangement issues [6].

Huynh et al. (2019) researched disdain discourse ID on Vietnamese web-based entertainment text and showed the proficiency of Bi-GRU-LSTM-CNN models in perceiving unsafe material [7]. Wang et al. (2016) introduced semantic development methodologies to further develop short text order precision by word inserting grouping and CNNs [8].

Li et al. (2018) detailed a superior BiLSTM-CNN model for news text arrangement, showing progress in utilizing DL designs to sort literary information [9]. Remya and Ramya (2014) researched the weighted larger part voting classifier mix for connection order in organic writing, showing group learning procedures for specific regions [10].

All in all, flow research has shown impressive advances in fraud detection and text arrangement draws near. From the utilization of DL models, for example, CNNs and LSTM to the examination of ensemble learning draws near, scientists are continuously endeavoring to work on the viability of distinguishing fake activities and appropriately ordering printed information. These drives are basic in safeguarding web stages and ensuring the honesty of advanced information in the present connected world.

III. METHODOLOGY

Modules:

- ADMIN: - We may examine the analysis of several algorithms in the admin module.
- USER: - Predictions are made in the user module using the parameters that the user specifies.
- FRONT END: - HTML, CSS is used for front end templates.

A) System Architecture

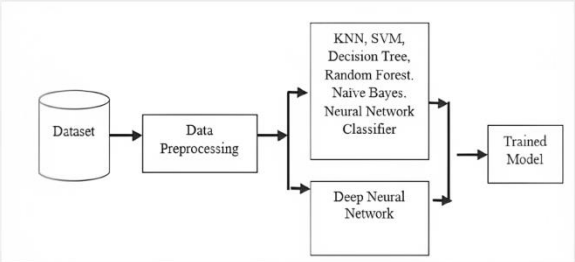


Fig 1: System Architecture

Proposed work

The proposed system utilizes data mining and classification algorithms to battle fake job postings. KNN, Decision Tree, Support Vector Machine, Naive Bayes, Random Forest, Multilayer Perceptron, and Deep Neural Network are trained and evaluated using the Employment Scam Aegean Dataset (EMSCAD), which has 18,000 examples.

To safeguard job searchers from frauds, the framework should recognize genuine and fake job notices. Because of its better presentation, the Deep Neural Network classifier is generally encouraging. Three rich layers in the DNN engineering permit it to get point by point examples and subtleties of fake job ads. A diverse strategy that utilizes various grouping calculations and information mining approaches yields a 98% classification accuracy utilizing the DNN classifier. Clients might make taught decisions and decrease the risk of fake business chances thanks to the framework's high exactness rate in spotting fake job commercials.

B) Dataset Collection

The Employment Scam Aegean Dataset (EMSCAD) distinguishes business cheats. It incorporates work advertisements' printed portrayals, business data, work models, and misrepresentation or certifiable names. The dataset assists plan and test with machine learning

models for business fraud detection by uncovering their patterns and properties.

Analysts, information researchers, and network protection experts need EMSCAD to identify and forestall fake job tricks, which damage work searchers and ventures. Specialists can distinguish con artists' techniques, like misleading sets of responsibilities and fake corporate profiles, and make calculations to recognize questionable advertisements progressively by concentrating on the data. The EMSCAD assists battle online extortion and make the occupation with advertising more secure and more dependable.

C) ADMIN

The administrator module controls calculation examination in our framework. We contrast ML calculations with track down the best exact and effective forecast strategy. Using the EMSCAD dataset, we carefully ran K-nearest neighbor (KNN), decision tree, support vector machine (SVM), naive Bayes classifier, and random forest classifier.

We utilized a 80:20 train-test split for all calculations to keep up with consistency. Investigation measures incorporate Accuracy, Precision, Recall, and F1-Score. After thorough examination and perception, we found that the Random Forest calculation consistently beat its rivals, making it the best prediction technique.

D) USER

In light of the User provided boundaries, the User module makes forecast simpler. Users should pick relevant measures from a dropdown list for this situation. The framework rapidly dissects the client inputted boundaries to decide whether a job promotion is genuine or deceitful by using the prepared Random Forest model. The User is then promptly given the result of the gauge, furnishing them with adroit data.

E) FRONT END

Our framework's front end is unequivocally planned using HTML and CSS for a wonderful and natural user experience. Python Flask associates frontend layouts to backend capabilities. User experience and framework route are improved by this technique.

We guarantee cross-stage similarity and responsiveness by involving HTML and CSS for front end improvement, serving a fluctuated assortment of clients across gadgets. Our framework's adaptability and versatility are improved by utilizing Python Jar to impart between the frontend and backend. All in all, our frontend parts influence HTML, CSS, and Python Flask to give a smooth and connecting with client experience.

F) Algorithms

K- Nearest Neighbor

KNN is a sluggish supervised learning strategy, as it takes more time to prepare. The K Nearest Neighbor working idea depends on doling out weight to every piece of information, known as a neighbor. The K Nearest Neighbor distance is determined for every one of the K Nearest data of interest in the preparation dataset, and characterization is done in view of most of votes. There are three sorts of distances that should be estimated: Eucledian, Manhattan, and Minkowski distance. Eucledian is the most utilized distance computing recipe.

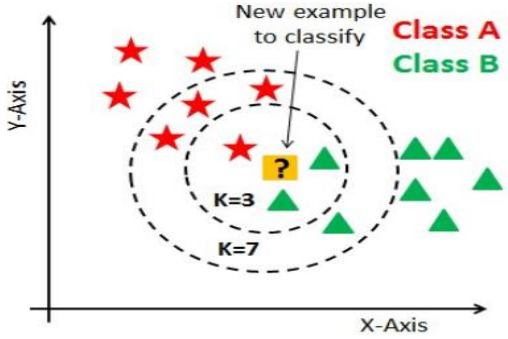
$$Eucledian\ Distance = D(x,y) = (x_i - y_i)_{2k_i} = 1 \quad (1)$$

K=number of cluster

x , y=co-ordinate sample spaces

The means recorded beneath characterize the KNN calculation:

1. The preparation tests are addressed by D, and the quantity of nearest neighbors is shown by k.
2. For each example class, make a super class.
3. Ascertain each preparing test's Euclidian distance.
4. Characterize the example utilizing the neighbor's greater part class as an aide.



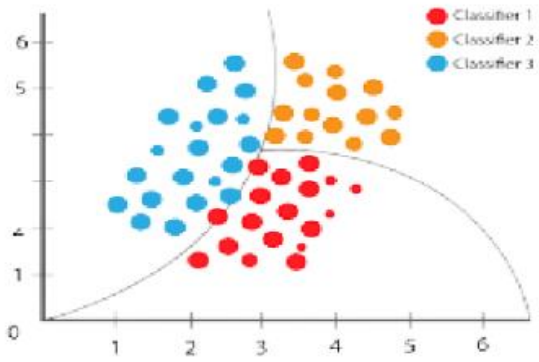
Naive Bayes

Bayes hypothesis supports Naive Bayes classifier. It includes serious areas of strength for a supposition, expecting that a class element's presence or nonappearance is inconsequential to some other class highlight. The Naive bayes classifier might be prepared under oversight. It utilizes maximal comparability. It needs little preparation information.

Order boundaries are assessed. Decide just the variable difference for each class, not the grid. Naive bayes is normally utilized with huge sources of info. It creates more intricate result. Each information characteristic's likelihood is introduced from the anticipated state. Machine learning and data mining use naïve bayes classification.

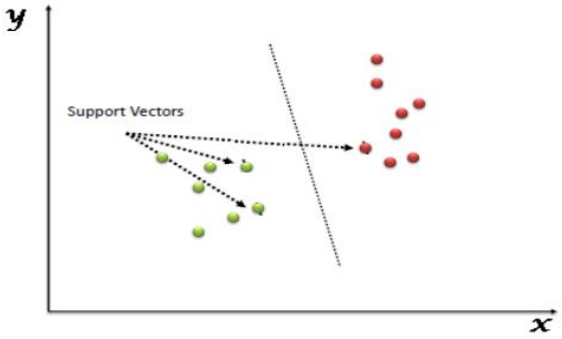
Bayes theorem formula is given by P(H|X) = P(X|H) * P(H) / P(X)

i. Here, P(H|X) is the posterior probability of H given the condition on X
ii. P(X|H) is the posterior probability of X given the condition on H
iii. P(H) is the prior probability of H. P(X) is the prior probability of X.



Support Vector Machine

Support Vector Machine is a very common supervised machine learning technique (with a predefined target variable) that can be used both as a classifier and as an indicator. For classification, she finds a hyperplane that identifies the classes in the element space. The SVM model encodes the target processing data as points in the element space, positioned so that points belonging to different classes are separated by as large an edge as possible. The test information focuses are then planned into a similar region and ordered by which side of the edge they fall.



Decision Tree

The decision tree algorithm is a member of the family of supervised learning algorithms. Unlike other supervised learning algorithms, the decision tree approach is also suitable for dealing with regression and classification problems. The basic objective of using decision trees is to build a decision model that can predict the class or value of a target factor by learning decision criteria derived from previous data (processed data).

Random forest

Our test utilized random forest classifier. Decision tree models are conspicuous in information mining because of their straightforwardness and adaptability in overseeing different information credits. Be that as it may, single-tree models might be delicate to explicit preparation information and effectively over-burden. Ensemble calculations total individual decisions to further develop exactness contrasted with single classifiers. One group method, random forest, utilizes a few tree indicators with a similar dissemination and an irregular free dataset for each tree. The strength of each tree and their affiliation decide random forest limit. Random forests perform better with more grounded single trees and less connection between's trees. Trees fluctuate because of bootstrapped tests and haphazardly chose information properties.

Implementation steps are:
i. Import and print the dataset
ii. Select all rows and columns 1 of x from the dataset and all rows and columns 2 of y
iii. Fit a Random Forest regressor to the dataset
iv. Predict the new outcome
v. Visualization of results

IV. EXPERIMENTAL RESULTS

A) Comparison Graphs → Accuracy, Precision, Recall, f1 score

Accuracy: The limit of a test to accurately distinguish feeble and solid occasions is known as exactness. We ought to record the little level of genuine positive and genuine adverse outcomes in completely checked on examples to gauge the exactness of a test. This may be communicated numerically as:

Accuracy = TP + TN TP + TN + FP + FN.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

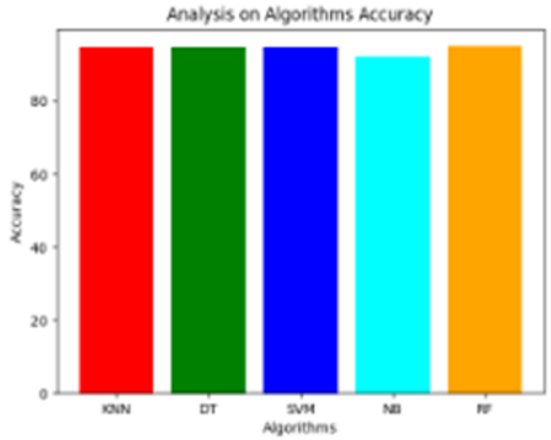


Fig 2: Accuracy Graph

Precision: Precision evaluates the level of accurately arranged examples or events among the up-sides. Thus, the precision not entirely settled by applying the ensuing equation:

Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

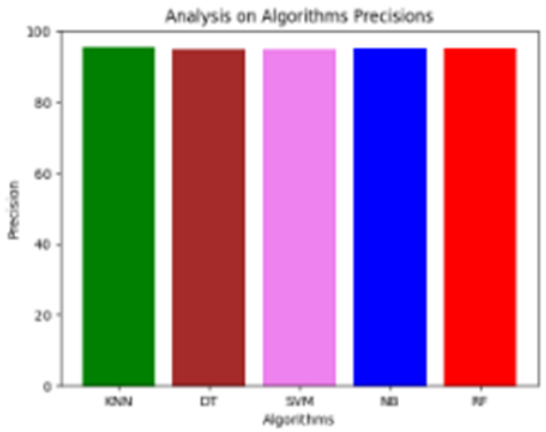


Fig 3: Precision Score Graph

Recall: Recall is a machine learning technique that evaluates a model's ability to recognize all significant examples of a particular class. A small fraction of accurately predicted positive impressions, which provides a solid benefit, gives us data about the model's ability to detect a particular class of event.

$$Recall = \frac{TP}{TP + FN}$$

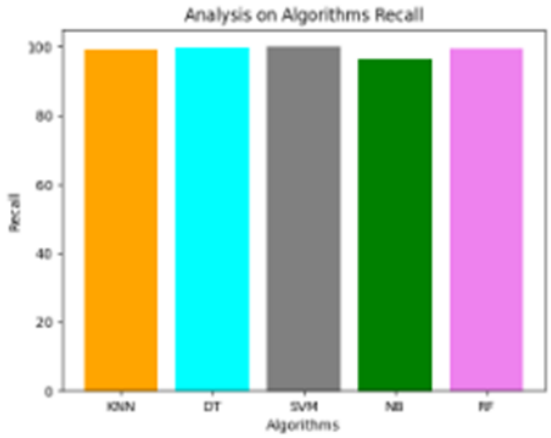


Fig 4: Recall Score Graph

F1-Score: The F1 score is an ML evaluation metric that measures the accuracy of a model. It combines the model's precision and recall values. The precision estimate calculates how often the model made successful predictions across the entire data set.

$$\textbf{F1 Score} = \frac{2}{\left(\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}\right)}$$

$$\textbf{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

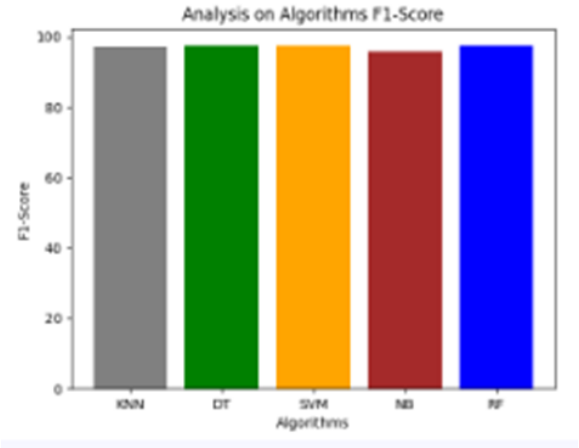


Fig 5: F1 Score Graph

B) Frontend

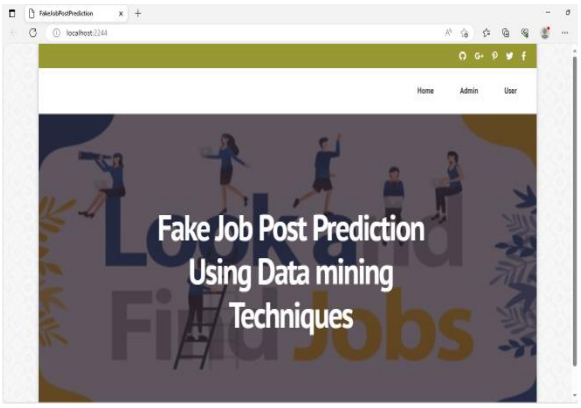


Fig 6: Main Page

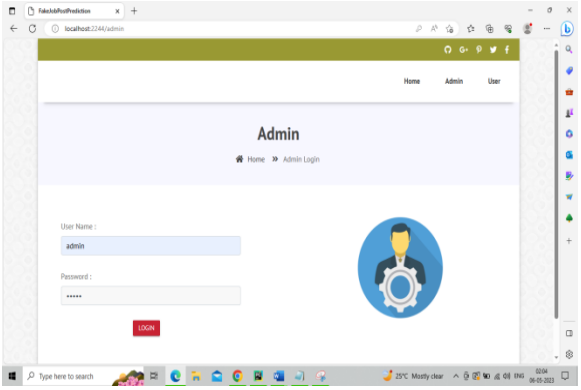


Fig 7: Admin page

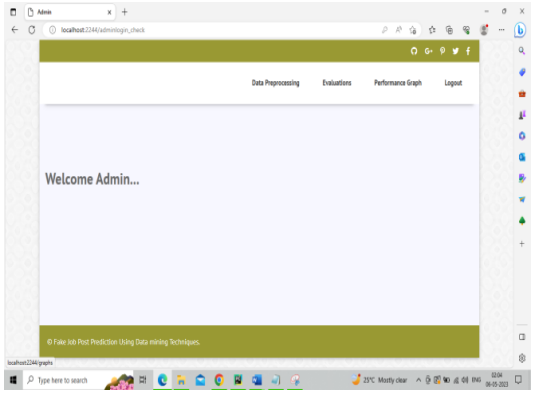


Fig 8: Admin welcome page

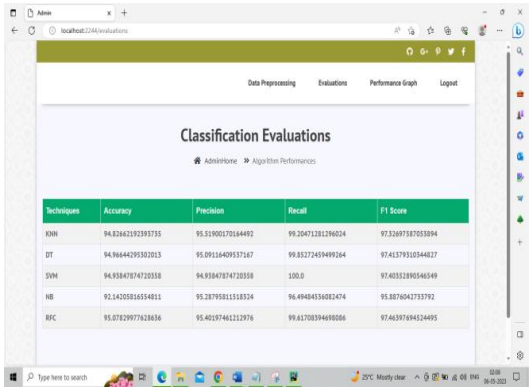


Fig 9: Evaluation or metrics

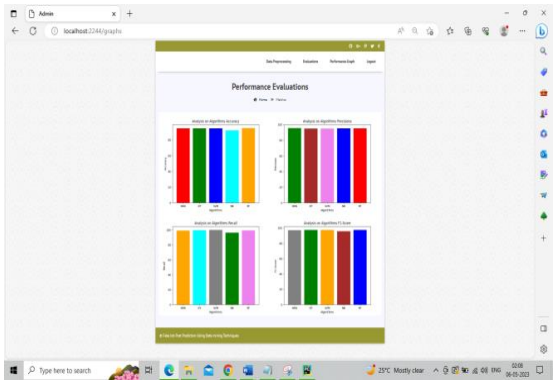


Fig 10: Performance Analysis in Graph

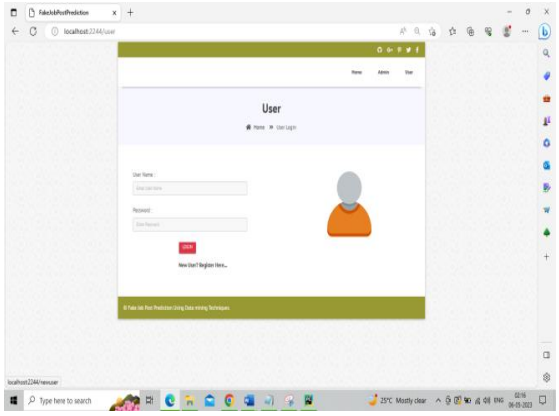


Fig 11: User Login

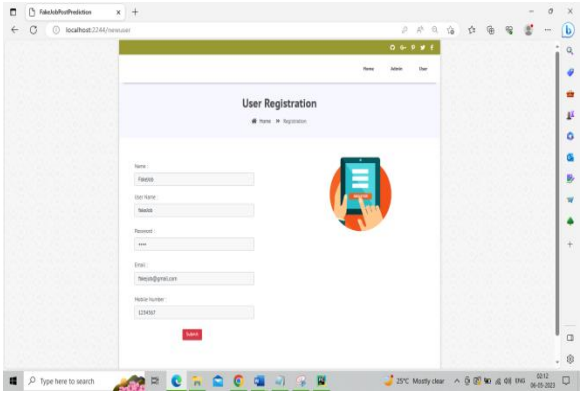


Fig 12: New User Registration

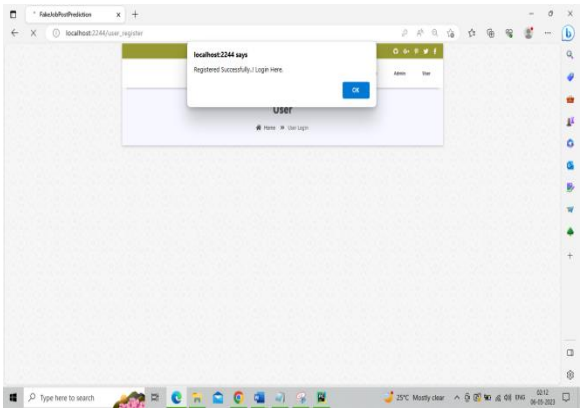


Fig 13: New User Created Alert

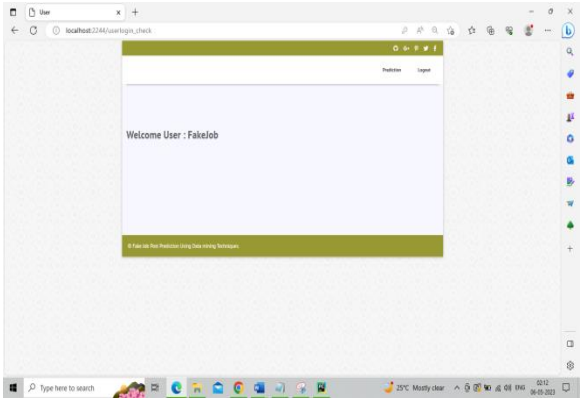


Fig 14: Welcome User

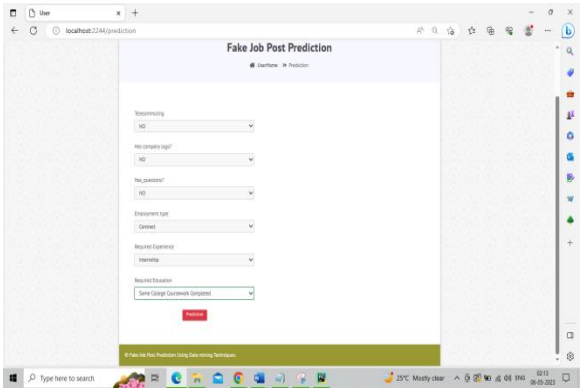


Fig 15: Enter Input Values

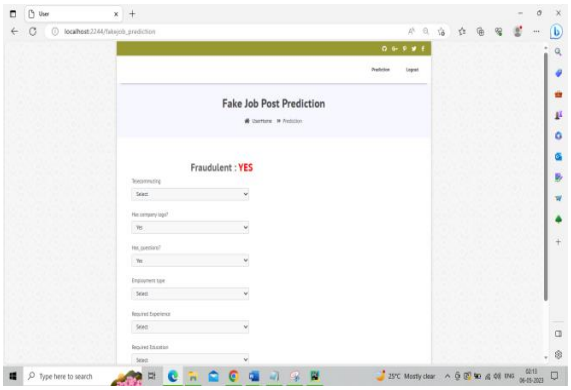


Fig 16: Prediction

V. CONCLUSION

All in all, business fakes are a worldwide issue that requires compelling recognition. We acquired different bits of knowledge from investigating these cheats and trying different things with the EMSCAD dataset. In the first place, fake job promotions confound unwary work searchers and encourage online stage doubt, featuring the requirement for viable ways of forestalling them. Second, our testing of SVM, KNN, Naive Bayes, Random Forest, and MLP shows the intricacy of discovery. Every calculation has upsides and downsides, so it means a lot to utilize various strategies to battle work fakes.

Our discoveries additionally show that ML can further develop work fraud detection. These strategies can recognize fake postings utilizing information examples and characteristics. Job tricks are continuously changing, subsequently distinguishing techniques should be refreshed. Our work further accentuates the need of great datasets like the EMSCAD dataset for fruitful examination in this field. Datasets like this help train and assess location models, working on their genuine execution. All in all, business cheats are a central issue, yet our review gives arrangements. By utilizing ML and top notch datasets, we can further develop ways of recognizing and lessen fake job postings, safeguarding position searchers and helping trust in web based selecting stages.

VI. FUTURE SCOPE

Future job fraud detection exploration could improve scholastics and industry. Coordinating complex deep learning procedures like RNNs and CNNs to further develop fraud detection models appears to be

encouraging. Fusion of text, picture, and conduct information may likewise assist with making sense of extortion. Specialists, policing, business partners should cooperate to make powerful discovery frameworks that can adjust to evolving tricks. NLP examination of sets of expectations and client associations could further develop discovery. These models should be refreshed and approved utilizing genuine world datasets to handle the always changing work trick scene.

REFERENCE

[1] S. Vidros, C. Kolias, G. Kambourakis, and L. Akoglu, "Automatic Detection of Online Recruitment Frauds: Characteristics, Methods, and a Public Dataset", Future Internet 2017, 9, 6; doi:10.3390/fi9010006.

[2] B. Alghamdi, F. Alharby, "An Intelligent Model for Online Recruitment Fraud Detection", Journal of Information Security, 2019, Vol 10, pp. 155176, https://doi.org/10.4236/iis.2019.103009 .

[3] Tin Van Huynh1, Kiet Van Nguyen, Ngan Luu-Thuy Nguyen1, and Anh Gia-Tuan Nguyen, "Job Prediction: From Deep Neural Network Models to Applications", RIVF International Conference on Computing and Communication Technologies (RIVF), 2020.

[4] Jiawei Zhang, Bowen Dong, Philip S. Yu, "FAKEDETECTOR: Effective Fake News Detection with Deep Diffusive Neural Network", IEEE 36th International Conference on Data Engineering (ICDE), 2020.

[5] Scanlon, J.R. and Gerber, M.S., "Automatic Detection of Cyber Recruitment by Violent Extremists", Security Informatics, 3, 5, 2014, https://doi.org/10.1186/s13388-014-0005-5

[6] Y. Kim, "Convolutional neural networks for sentence classification," arXiv Prepr. arXiv1408.5882, 2014.

[7] T. Van Huynh, V. D. Nguyen, K. Van Nguyen, N. L.-T. Nguyen, and A.G.- T. Nguyen, "Hate Speech Detection on Vietnamese Social Media Text using the Bi-GRU-LSTM-CNN Model," arXiv Prepr. arXiv1911.03644, 2019.

[8] P. Wang, B. Xu, J. Xu, G. Tian, C.-L. Liu, and H. Hao, "Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification," Neurocomputing, vol. 174, pp. 806814, 2016.

[9] C. Li, G. Zhan, and Z. Li, "News Text Classification Based on Improved BiLSTM-CNN," in 2018 9th International Conference on Information Technology in Medicine and Education (ITME), 2018, pp. 890-893.

[10] K. R. Remya and J. S. Ramya, "Using weighted majority voting classifier combination for relation classification in biomedical texts," International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2014, pp. 1205-1209