# Prediction of Fake Job Ad Using NLP-Based Multilayer Perceptron Classifier

Shaik Osiha[1], D.Murali[2]

[1]*PG Student, QUBA College of engineering and technology*

[2]*Associate Professor, QUBA College of engineering and technology*

**Abstract: In recent years, due to advancement in modern technology and social communication, advertising new job post has become very common issues in the present world. So, fake job posting prediction task is going to be a great concern for all. Like many other classification tasks, fake job posting prediction leaves a lot of challenges to face. The fraudulent post-detection work draws proper attention to obtaining an automated tool to identify fake jobs and report them to people to avoid applying for such situations. Human judgment can be subjective in some platforms. Most platforms rely on humans to flag suspicious job postings. Users can report ads and users report fake job or genuine. Moderators review them manually. Scammers exploit online job platforms with fake ads, wasting job seekers time and potentially harming them. Misidentifying legitimate ads or failing to detect certain types of scams. Manual review doesn't scale well with the vast number of online job postings. Project used Natural Language Processing (NLP) and a Multilayer Perceptron (MLP) classifier to predict if it's real or fake. Job search platforms like Indeed, Glassdoor and LinkedIn these are the applications. Valid job postings can be done by filtering out fake ad, using a Natural Language Processing technique in combination with TFIDF vectorization and Neural Network. With the help of the MLP job seekers find out their jobs depending on their qualification, experience, suitability etc. Recruitment process is now influenced social media.**

***Index Terms*****—: Fake Job Prediction, NLP, MLP, TFIDF, Neural Network, Machine learning.**

## 1. INTRODUCTION

### 1.1 OVERVIEW

The prediction of fake job advertisements using NLP (Natural Language Processing) and a Multilayer Perceptron (MLP) classifier represents a cutting-edge approach in the fight against employment fraud. This method leverages the power of machine learning and linguistic analysis to scrutinize job postings, distinguishing between legitimate opportunities and deceptive listings designed to exploit job seekers. By analyzing textual data within job ads, such as job descriptions, requirements, and company information, NLP techniques extract and process linguistic features that are indicative of fraud. The MLP classifier, a type of neural network known for its ability to learn and model complex patterns, then evaluates these features to predict the authenticity of job postings. This approach not only enhances the efficiency and accuracy of detecting fraudulent job offers but also significantly reduces the manual effort involved in such verification processes. Consequently, it serves as a crucial tool in protecting individuals from employment scams, thereby fostering a safer job-search environment. As this technology evolves, it holds the promise of becoming even more sophisticated, adapting to new scamming techniques and further safeguarding the integrity of online job markets.

### 1.2 HISTORY

Predicting fake job ads using an NLP-based Multilayer Perceptron (MLP) Classifier involves applying machine learning and natural language processing techniques to distinguish between genuine and fraudulent job postings. While I can't provide a "history" of such a specific application as it's a relatively niche and modern problem, I can outline how such a system might be developed and the steps involved in its creation and application. The process reflects broader trends in applying AI for online security and integrity.

The journey towards using NLP (Natural Language Processing) and Multilayer Perceptron (MLP) classifiers for the prediction of fake job ads has its roots in the broader evolution of machine learning and artificial intelligence over the past few decades.

Initially, the detection of fraudulent online content relied heavily on manual verification and simple rule-based algorithms, which were not only time-consuming but also less effective against sophisticated scams. As the internet became the primary platform for job listings, the volume of postings made manual checks increasingly impractical, necessitating the development of more advanced, automated solutions.

### 1.3 PROBLEM STATEMENT

To address the problem statement of predicting fake job ads using an NLP-based Multilayer Perceptron (MLP) Classifier, let's break down the task into a structured approach, detailing each step necessary to develop, evaluate, and implement such a system. This approach will encompass data collection and preprocessing, feature extraction, model building and training, evaluation, and potential deployment strategies.

### 1.4 RESEARCH MOTIVATION

The motivation for researching the prediction of fake job advertisements using NLP-based Multilayer Perceptron (MLP) classifiers stems from a pressing need to safeguard the integrity of the online job market and protect job seekers from the growing menace of employment scams.

## II LITERATURE SURVEY

### 2.1 LITERATURE SURVEY

Snidhuja, et al. [1], proposed the use of different data mining techniques and classification algorithms like K-nearest neighbor, decision tree, support vector machine, naive Bayes classifier, random forest classifier, and multi-layer perceptron to predict whether a job advertisement was real or fraudulent. They experimented on the Employment Scam Aegean Dataset (EMSCAD), which contained 18,000 samples. A deep neural network, as a classifier, performed greatly for this classification task. They used three dense layers for this deep neural network classifier. The trained classifier showed approximately 98% classification accuracy (DNN) in predicting a fraudulent job ad.

Babu, et al. [2], proposed the use of different data mining techniques and classification algorithms like K-nearest neighbor, decision tree, support vector machine, naive Bayes classifier, random forest classifier, and multi-layer perceptron to predict whether a job advertisement was real or fraudulent. They had experimented on the Employment Scam Aegean Dataset (EMSCAD), which contained 18,000 samples. A deep neural network, as a classifier, had performed greatly for this classification task. They had used three dense layers for this deep neural network classifier. The trained classifier showed approximately 98% classification accuracy (DNN) in predicting a fraudulent job ad.

## III EXISTING METHODOLOGY

### 3.1 K- NEAREST NEIGHBOR

The K Nearest Neighbors (KNN) algorithm is used for both classification and regression problems. It stores all the known use cases and classifies new use cases (or data points) by segregating them into different classes. This classification is accomplished based on the similarity score of the recent use cases to the available ones.KNN is a supervised machine learning algorithm, wherein ‘K’ refers to the number of neighboring points we consider while classifying and segregating the known n groups. The algorithm learns at each step and iteration, thereby eliminating the need for any specific learning phase. The classification is based on the neighbor’s majority vote.

ALGORITHMS

Step 1 – When implementing an algorithm, you will always need a data set. So, you start by loading the training and the test data.

Step 2 – Choose the nearest data points (the value of K). K can be any integer.

Step 3 – Do the following, for each test data –

- Use Euclidean distance, Hamming, or Manhattan to calculate the distance between test data and each row of training. The Euclidean method is the most used when calculating distance.
- Sort data set in ascending order based on the distance value.
- From the sorted array, choose the top K rows.
- Based on the most appearing class of these rows, it will assign a class to the test point.

Step 4 – End

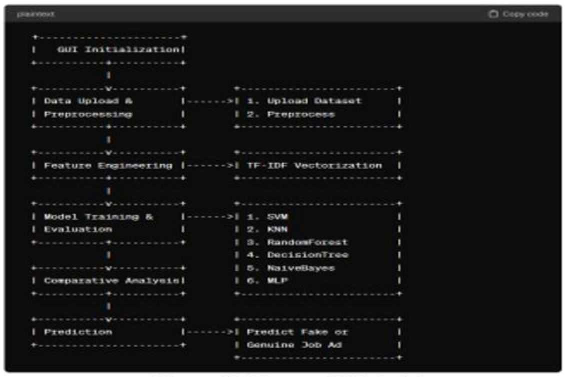
DRAWBACKS

KNN algorithm is that it does not create a generalized separable model. There is no summary equations or trees that can be produced by the training process that can be quickly applied to new records. Instead, KNN simply uses the training data itself to perform prediction. KNN provides no insight about the relative importance of each predictor. Another significant disadvantage of KNN, is that the algorithm is computationally intensive.

## IV. PROPOSED METHODOLOGY

### 4.1 OVERVIEW

The methodology for predicting fake job ads using an NLP-based Multilayer Perceptron (MLP) classifier involves collecting a diverse dataset of job advertisements, preprocessing the text data by cleaning and tokenizing, extracting relevant features through NLP techniques, labeling the data, and training an MLP classifier. The model is then validated, optimized, and evaluated using metrics like precision and recall. The final step includes integrating the model into a system for real-time prediction, with a focus on continuous improvement through monitoring and updates.



```
+--------------------+
|   GUI Initialization|
+----------+----------+
           |
+----------v----------+        +-------------------------+
| Data Upload &       |------->| 1. Upload Dataset       |
| Preprocessing       |        | 2. Preprocess           |
+----------+----------+        +-------------------------+
           |
+----------v----------+        +-------------------------+
| Feature Engineering |------->| TF-IDF Vectorization    |
+----------+----------+        +-------------------------+
           |
+----------v----------+        +-------------------------+
| Model Training &    |------->| 1. SVM                  |
| Evaluation          |        | 2. KNN                  |
+----------+----------+        | 3. RandomForest         |
           |                   | 4. DecisionTree         |
+----------v----------+        | 5. NaiveBayes           |
| Comparative Analysis|        | 6. MLP                  |
+----------+----------+        +-------------------------+
           |
+----------v----------+        +-------------------------+
| Prediction          |------->| Predict Fake or         |
+---------------------+        | Genuine Job Ad          |
                               +-------------------------+
```

Figure.4.1. Proposed Methodology

### 4.2 DATA PRE-PROCESSING

Data pre-processing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model. When creating a machine learning project, it is not always a case that we come across the clean and formatted data and while doing any operation with data, it is mandatory to clean it and put in a formatted way. So, for this, we use data pre-processing task.

### 4.3 DATA SPLITTING

In machine learning data pre-processing, we divide our dataset into a training set and test set. This is one of the crucial steps of data pre-processing as by doing this, we can enhance the performance of our machine learning model. Suppose if we have given training to our machine learning model by a dataset and we test it by a completely different dataset. Then, it will create difficulties for our model to understand the correlations between the models. If we train our model very well and its training accuracy is also very high, but we provide a new dataset to it, then it will decrease the performance.

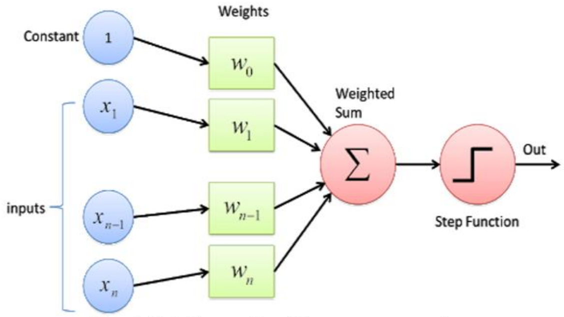


Figure.4.4.1 Architecture of a multilayer perceptron network

## V UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group. The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process also be added to; or associated with, UML.

Goals: The Primary goals in the design of the UML are as follows:

- Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
- Provide extendibility and specialization mechanisms to extend the core concepts.
- Be independent of particular programming languages and development process.

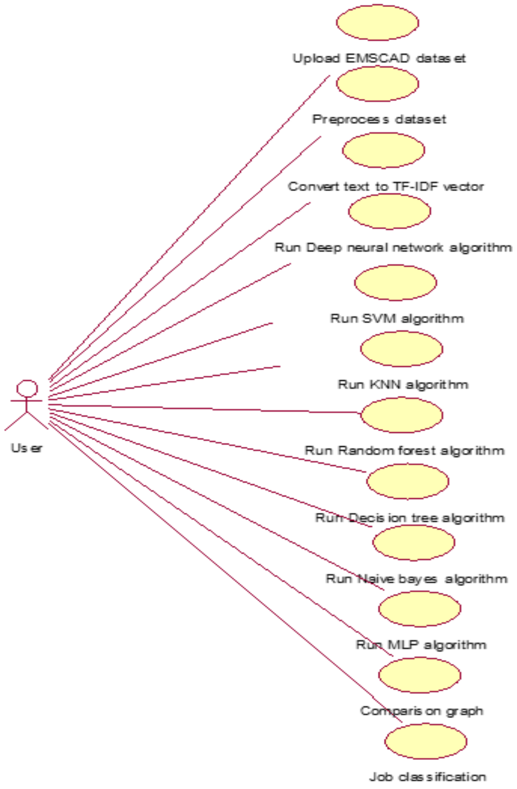

Upload EMSCAD dataset
Preprocess dataset
Convert text to TF-IDF vector
Run Deep neural network algorithm
Run SVM algorithm
Run KNN algorithm
Run Random forest algorithm
Run Decision tree algorithm
Run Naive bayes algorithm
Run MLP algorithm
Comparison graph
Job classification
User

Figure:5.1 Use Case Diagram

## VI. SOFTWARE ENVIRONMENT

### WHAT IS PYTHON?

Below are some facts about Python.

- Python is currently the most widely used multi-purpose, high-level programming language.
- Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java.
- Programmers must type relatively less and indentation requirement of the language, makes them readable all the time.
- Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber… etc.
- The biggest strength of Python is huge collection of standard libraries which can be used for the following –
- Machine Learning
- GUI Applications (like Kivy, Tkinter, PyQt etc. )
- Web frameworks like Django (used by YouTube, Instagram, Dropbox)
- Image processing (like Opencv, Pillow)
- Web scraping (like Scrapy, BeautifulSoup, Selenium)
- Test frameworks
- Multimedia

### ADVANTAGES OF PYTHON

Let's see how Python dominates over other languages.

1. Extensive Libraries

Python downloads with an extensive library and it contain code for various purposes like regular expressions, documentation-generation, unit-testing, web browsers, threading, databases, CGI, email, image manipulation, and more. So, we don't have to write the complete code for that manually.

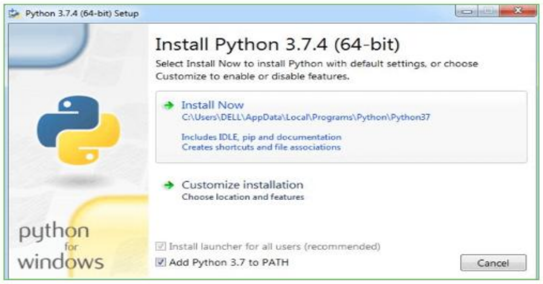
2. Extensible

As we have seen earlier, Python can be extended to other languages. You can write some of your code in languages like C++ or C. This comes in handy, especially in projects.

3. Embeddable

Complimentary to extensibility, Python is embeddable as well. You can put your Python code in your source code of a different language, like C++. This lets us add scripting capabilities to our code in the other language.

Python is a programming language that lets you work quickly and integrate systems more effectively. >>> Learn More


Python 3.7.4 (64-bit) Setup

Install Python 3.7.4 (64-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

Install Now
C:\Users\DELL\AppData\Local\Programs\Python\Python37
Includes IDLE, pip and documentation
Creates shortcuts and file associations

Customize installation
Choose location and features

☑ Install launcher for all users (recommended)
☑ Add Python 3.7 to PATH

## VII SYSTEM REQUIREMENTS

### SOFTWARE REQUIREMENTS

The functional requirements or the overall description documents include the product perspective and features, operating system and operating environment, graphics requirements, design constraints and user documentation.

The appropriation of requirements and implementation constraints gives the general overview of the project in regard to what the areas of strength and deficit are and how to tackle them.

- Python IDLE 3.7 version (or)
- Anaconda 3.7 (or)
- Jupiter (or)
- Google colab

### HARDWARE REQUIREMENTS

Minimum hardware requirements are very dependent on the particular software being developed by a given Enthought Python / Canopy / VS Code user. Applications that need to store large arrays/objects in memory will require more RAM, whereas applications that need to perform numerous calculations or tasks more quickly will require a faster processor.

| | | |
|---|---|---|
| Operating system | : | Windows, Linux |
| Processor | : | minimum intel i3 |
| Ram | : | minimum 4 GB |
| Hard disk | : | minimum 250GB |

## VIII FUNCTIONAL REQUIREMENTS

### OUTPUT DESIGN

Outputs from computer systems are required primarily to communicate the results of processing to users. They are also used to provides a permanent copy of the results for later consultation. The various types of outputs in general are:

- External Outputs, whose destination is outside the organization
- Internal Outputs whose destination is within organization and they are the
- User’s main interface with the computer.
- Operational outputs whose use is purely within the computer department.
- Interface outputs, which involve the user in communicating directly.

### OUTPUT DEFINITION

The outputs should be defined in terms of the following points:

- ➢ Type of the output
- ➢ Content of the output
- ➢ Format of the output
- ➢ Location of the output
- ➢ Frequency of the output
- ➢ Volume of the output
- ➢ Sequence of the output

It is not always desirable to print or display data as it is held on a computer. It should be decided as which form of the output is the most suitable.

### INPUT DESIGN

Input design is a part of overall system design. The main objective during the input design is as given below:

- ✓ To produce a cost-effective method of input.
- ✓ To achieve the highest possible level of accuracy.
- ✓ To ensure that the input is acceptable and understood by the user.

### INPUT STAGES

The main input stages can be listed as below:

- Data recording
- Data transcription
- Data conversion
- Data verification

## IX CONCLUSION AND FUTURE SCOPE

### CONCLUSION

The development of an NLP-based Multilayer Perceptron (MLP) Classifier for the prediction of fake job ads represents a significant advancement in combating online job fraud. By leveraging the capabilities of Natural Language Processing (NLP) and the computational power of neural networks, this approach offers a promising solution to a problem that affects countless job seekers and undermines the integrity of online job markets. The system's ability to analyse and learn from textual data allows it to identify patterns and features characteristic of fraudulent postings, which might be difficult for humans to discern consistently. Implementing such a system involves careful dataset preparation, including collection, cleaning, and labelling, followed by feature extraction using NLP techniques to convert text into a format that can be processed by the MLP. The design, training, and evaluation of the MLP model are critical steps that determine the effectiveness of the fraud detection system. When properly executed, this approach can significantly reduce the presence of fake job ads, protecting job seekers and helping maintain trust in online job platforms.

## FUTURE SCOPE

Model Improvement: Continuous research into NLP and neural network architectures could lead to more sophisticated models that are better at understanding the nuances of language and context, further improving the accuracy of fake job ad detection. Dataset

Expansion and Diversification: Collecting and incorporating more diverse and extensive datasets, including job ads from various industries, languages, and regions, could enhance the model's generalizability and effectiveness across different contexts.Real-Time Detection and Scalability: Future work could focus on optimizing the model for real-time detection and ensuring it can scale to handle the vast number of job postings processed by large online platforms.

Integration with Additional Features: Incorporating other data types, such as metadata about the posting company or user behavior data on job platforms, could provide additional signals for detecting fraudulent postings.User Feedback Loop: Implementing a system where users can report suspected fake ads could help in continuously refining the model.

Cross-Platform Collaboration: Sharing insights and data (while respecting privacy and ethical considerations) between different job platforms could lead to a more robust defense against job ad fraud across the entire online job market ecosystem.

Legal and Ethical Framework Development: As AI and machine learning systems become more integrated into societal functions, developing comprehensive legal and ethical frameworks to guide their application is crucial.

Adaptation to Other Forms of Online Fraud: The methodologies and technologies developed for detecting fake job ads could be adapted to combat other forms of online fraud, such as fraudulent real estate listings, scam e-commerce products, and more.

## REFERENCE

[1] Snidhuja, B., B. Anitha, A. Sowmya, and D. Srivalli. "Prediction of Fake Job Ad using NLP-based Multilayer Perceptron." Turkish Journal of Computer and Mathematics Education (TURCOMAT) 14, no. 1 (2023): 296-310.

[2] Babu, E., Nampelly Naresh, Muda Sai Kiran, Vadla Ruthuja, and Mukkapally Sai Pavan. "ARTIFICIAL INTELLIGENCE FOR FRAUDULENT JOB ADVERTISEMENT PREDICTION FROM EMSCAD."

[3] Thilagam, P. Santhi. "Multi-layer perceptron based fake news classification using knowledge base triples." Applied Intelligence 53, no. 6 (2023): 6276-6287.

[4] Mundra, Shikha, Jaiwanth Reddy, Ankit Mundra, Namita Mittal, Ankit Vidyarthi, and Deepak Gupta. "An Automated Data-driven Machine Intelligence Framework for Mining Knowledge To Classify Fake News Using NLP." ACM Transactions on Asian and Low-Resource Language Information Processing (2023).

[5] Singh, Veeraj R., P. Sampras, and Aryan Dhage. "Fake Job Post Prediction Using Data Mining." Journal of Scientific Research and Technology (2023): 39-47.

[6] Mouri, Ishrat Jahan, Biman Barua, M. Mesbahuddin Sarker, Alistair Barros, and Md Whaiduzzaman. "Predicting Online Job Recruitment Fraudulent Using Machine Learning." In Proceedings of Fourth International Conference on Communication, Computing and Electronics Systems: ICCCES 2022, pp. 719-733. Singapore: Springer Nature Singapore, 2023.

[7] Altheneyan, Alaa, and Aseel Alhadlaq. "Big data ML-based fake news detection using distributed learning." IEEE Access 11 (2023): 29447-29463.

[8] Shaukat, Muhammad Waqas, Rashid Amin, Muhana Magboul Ali Muslam, Asma Hassan Alshehri, and Jiang Xie. "A hybrid approach for alluring ads phishing attack detection using machine learning." Sensors 23, no. 19 (2023): 8070.