**Ensemble ML Modelling to classify Real/Fake jobs**

"Spotting the Fakes: Data Science Takes on Fraudulent Job Postings"

Introduction:

Ensemble ML Modelling to classify Real/Fake jobs centers around the development of a machine learning model for detecting fraudulent job postings within a comprehensive dataset of job listings. Through meticulous data preprocessing and exploratory analysis, key features were engineered to enhance the model's predictive capability. Leveraging classification algorithms including Logistic Regression, Random Forest Classifier and K-Nearest Neighbors, various models were trained and evaluated for their ability to accurately identify suspicious job listings. After rigorous evaluation, the Random Forest Classifier algorithm emerged as the most effective tool for this task, achieving a high level of accuracy.

Libraries:

**Pandas** facilitates seamless data handling through its DataFrame structures, simplifying data cleaning, manipulation, and structured data analysis. **NumPy** plays a foundational role, providing support for multidimensional arrays and a wide range of mathematical functions vital for numerical operations. **Matplotlib** and **Seaborn** are pivotal for data visualization, with Matplotlib offering diverse plot types and Seaborn enhancing aesthetics. **Regular** **Expressions** **(re)** are applied for advanced string manipulation and pattern matching, particularly beneficial for text data processing. The **NLTK (Natural Language Toolkit)**, a comprehensive library for natural language processing (NLP), is used for tasks such as tokenization and stemming in text analysis. The **Collections** module introduces practical alternatives, including the Counter class, which efficiently counts element frequencies, advantageous for word frequency analysis. **Scikit-learn (sklearn)** empowers the project with machine learning capabilities, encompassing various algorithms, data preprocessing, model evaluation, and selection tools.

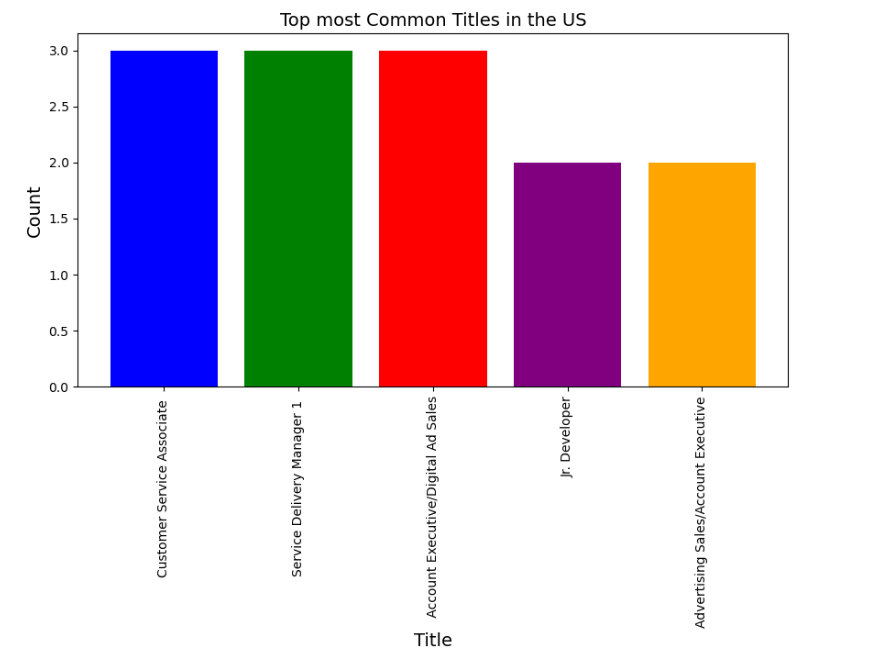
Dependencies:

"Fraudulent" is the target feature, X and y are used to represent the independent and dependent variables, respectively, in a dataset. The independent variables, denoted as X, encompass a range of features including job details like title and location, information about the company such as its profile, as well as specific requirements and benefits associated with the job posting. Additionally, it includes binary indicators like 'telecommuting', 'has\_company\_logo', and 'has\_questions', along with categorical variables like 'employment\_type', 'required\_experience', 'required\_education', 'industry', and 'function'. On the other hand, the dependent variable, represented by y, is labeled as 'fraudulent'.

Dataset Questions:

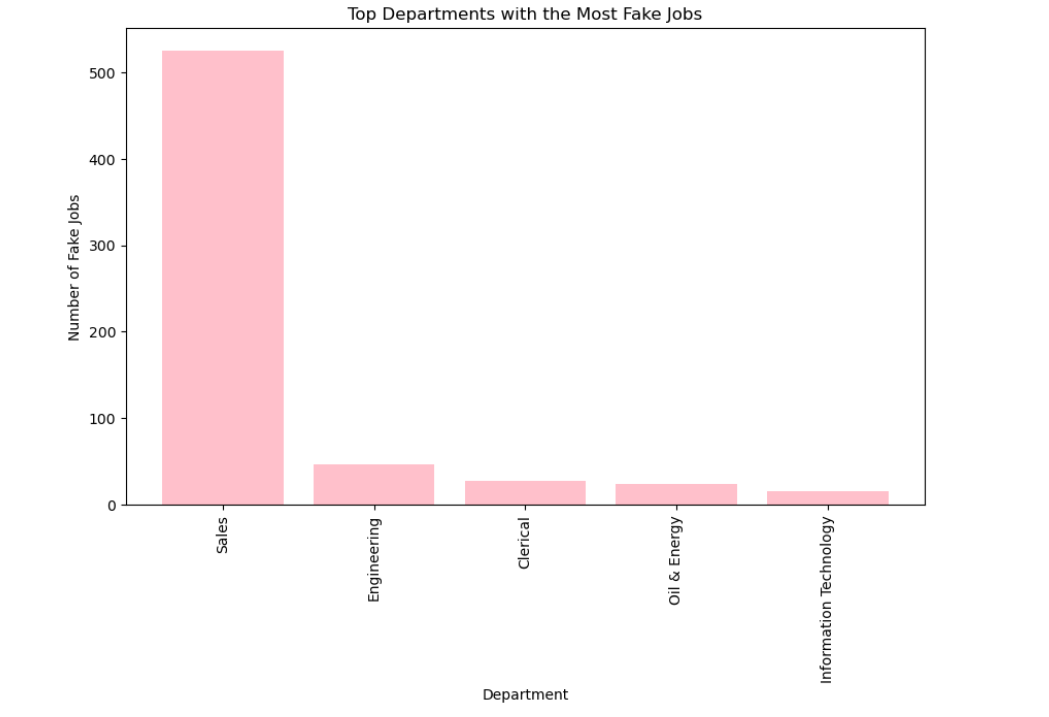
Q1) What are the most common title used in jobs in the US?

Answer: 'Customer Service Associate '



Q2) Which department has the most number of fake jobs?

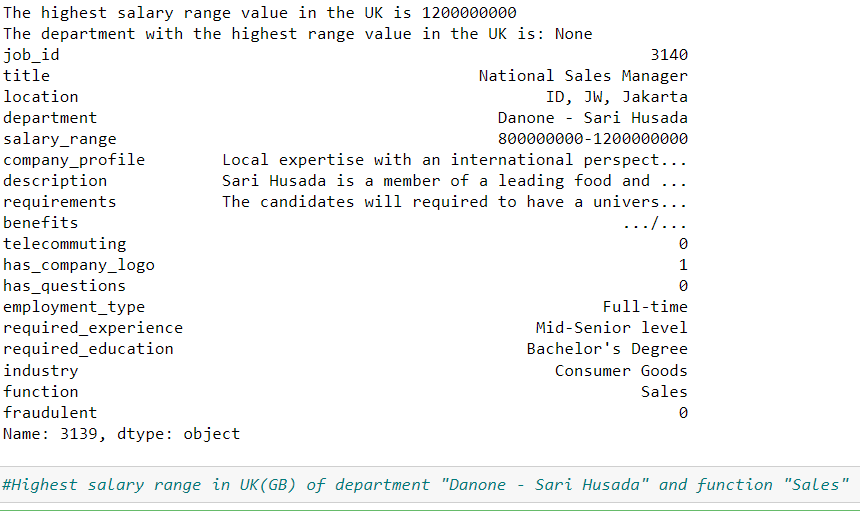
Answer: 'Sales'



Q3) Which department or function has high-paying jobs in the UK?

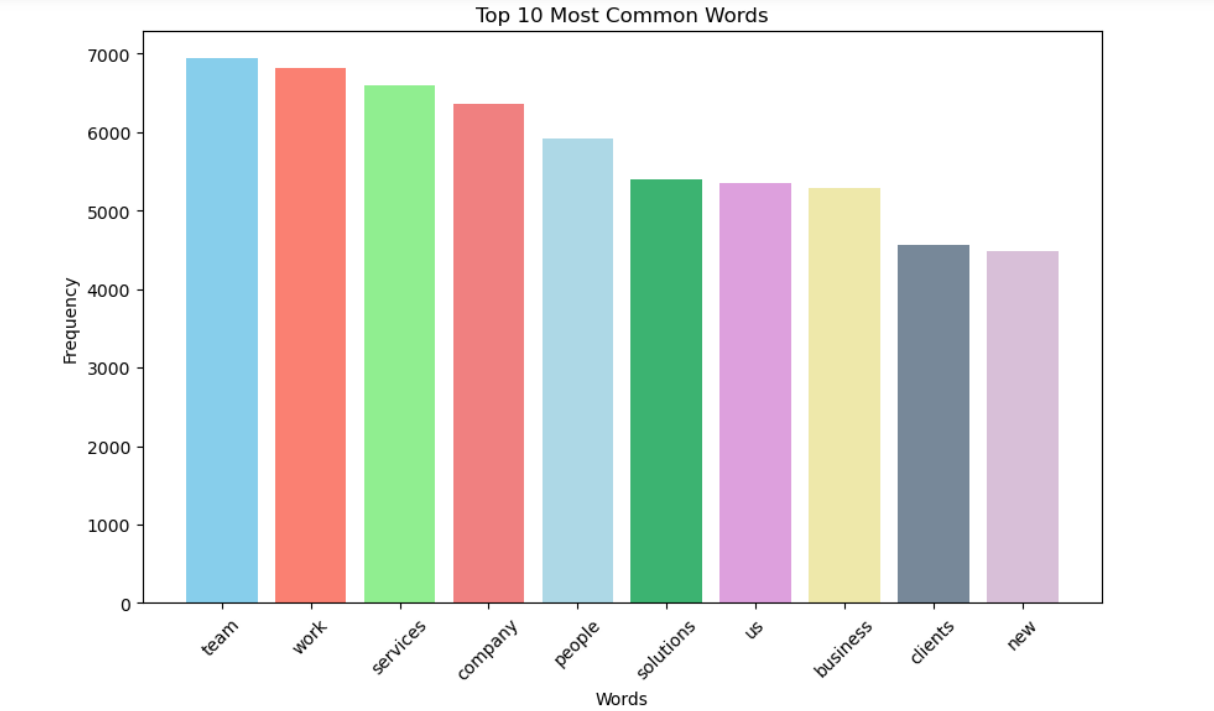
Answer:

* department - Danone - Sari Husada
* function – Sales



Q4) What are the top 3 most commonly used words in Company Profile? (Excluding stopwords)

Answer: [('team', 6934), ('work', 6821), ('services', 6593)]

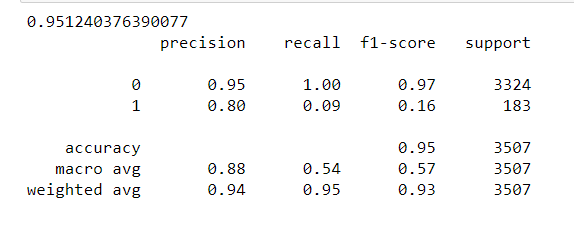


Algorithm:

Logistic Regression

Logistic regression is a widely employed algorithm in machine learning, particularly for binary classification tasks. Its popularity stems from its simplicity, interpretability, and effectiveness in scenarios where the outcome variable is categorical with two classes, as is the case in this dataset where the goal is to classify job postings as either fraudulent or legitimate. Logistic regression models the probability of a binary outcome using the logistic function, which ensures that predictions fall within the range of 0 and 1.

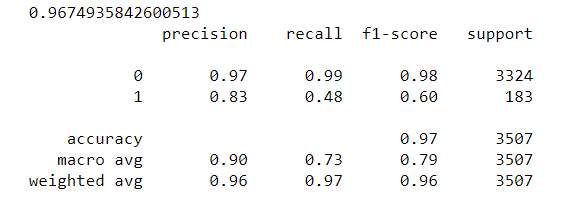
Accuracy = 0.95



K-Nearest Neighbors

The K-Nearest Neighbors (KNN) algorithm is a versatile choice for classification tasks, especially when the underlying data may not follow a specific mathematical model. KNN operates on the principle of similarity, where it classifies data points based on the majority class of their neighboring points. In the context of this dataset, KNN could be beneficial for job posting classification. It allows for a flexible decision boundary, accommodating non-linear relationships and irregularly shaped clusters of data. The attributes of neighboring job postings, KNN can make accurate predictions about the legitimacy of a given posting.

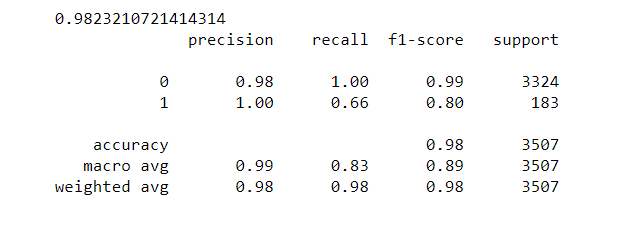
Accuracy = 0.97



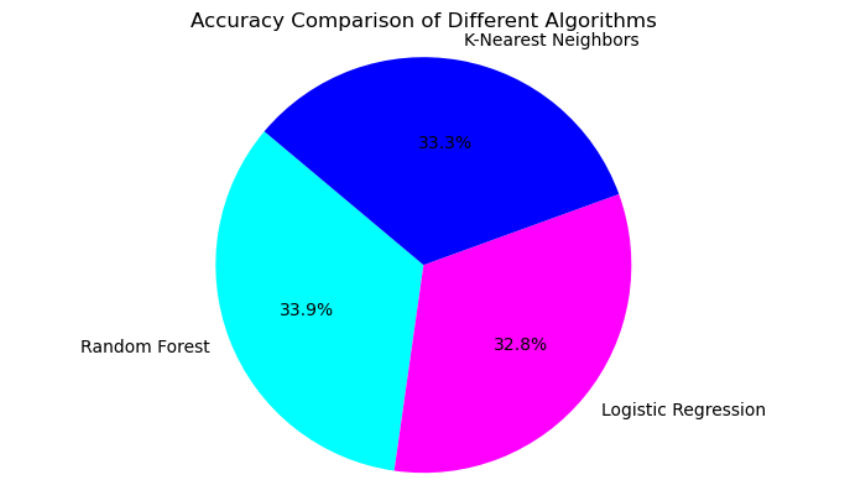
Random Forest Classifier:

The Random Forest Classifier is a formidable choice for classification tasks, particularly when dealing with complex and high-dimensional datasets, as is often the case in job posting classification. This ensemble learning algorithm combines the strength of multiple decision trees to yield robust and accurate predictions. Random Forest excels at capturing intricate relationships and interactions within the data, making it well-suited for discerning fraudulent from legitimate job postings. Its ability to handle both categorical and numerical features.

Accuracy = 0.98

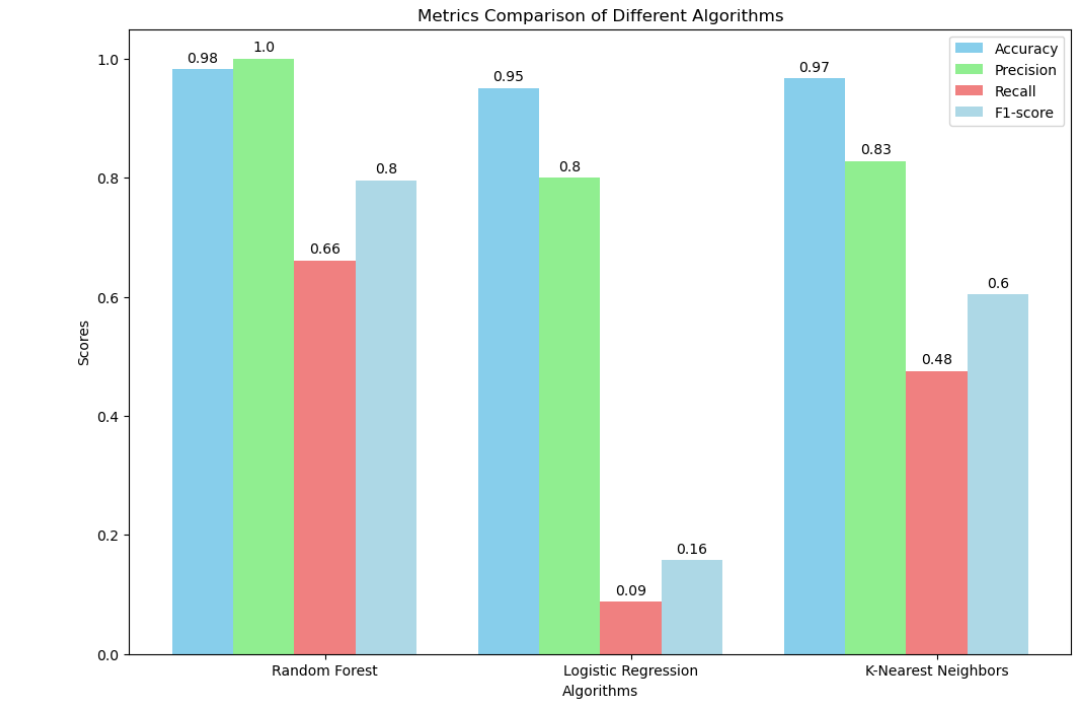


Hence, Random Forest classifier suites the dataset with its highest accuracy compared with Logistic regression and K-Nearest Neighbor algorithms.



**Metrics Calculation:**

Accuracy in a classification model measures overall correctness, calculated as the ratio of correct predictions to total instances. A higher accuracy signifies better performance. Precision emphasizes accurate positive predictions, computed as true positives divided by the sum of true positives and false positives. It's crucial for minimizing false positives. Recall, also known as sensitivity, evaluates the model's capability to identify all relevant instances. It is the ratio of true positives to the sum of true positives and false negatives. Recall is vital for minimizing false negatives. The F1-score, a harmonic mean of precision and recall, strikes a balance between the two. A higher F1-score indicates a better equilibrium between precision and recall.



**Summary**:

The Job Posting Fraud Detection project aims to identify potentially fraudulent job postings using a dataset comprising various job-related features. Several Python libraries were employed, including Pandas for data handling, NumPy for numerical computations, and Matplotlib and Seaborn for data visualization. Natural Language Toolkit (NLTK) aided in processing and analyzing text data, while Scikit-learn provided a suite of machine learning algorithms for classification tasks.

The project's dataset consisted of features such as job title, location, company details, and job description, with the target variable being the 'fraudulent' label. After extensive preprocessing and feature engineering, three machine learning algorithms were implemented: Random Forest Classifier, Logistic Regression, and K-Nearest Neighbors. The models were evaluated using metrics such as accuracy, precision, recall, and F1-score.

Random Forest Classifier exhibited the highest accuracy, making it the preferred algorithm for this particular problem. This project showcases the efficacy of machine learning techniques in detecting fraudulent job postings, which can be invaluable for online job platforms and job seekers alike. The visualization of metrics through bar charts provided a clear comparison of algorithm performance, offering valuable insights for model selection and refinement.