-

**DAB402 002- CAPSTONE PROJECT**

FINAL PROJECT REPORT

***Topic: RECRUITMENT SCAM  
(Predict a potential recruitment scam using machine learning )***

Prepared by

Group Members:

* Bency Eldho (ID:0731915)
* Mehak Rajdev (ID:0732638)
* Nanbal Akila Fompun (ID: 0714107)

**A Model for Online Recruitment Fraud Detection**

This project tries to accomplish privacy and loss of money for individuals and organization by creating a reliable model which can detect the fraud exposure in the online recruitment environments. The detection of Online Recruitment Fraud is characterized by other types of automated fraud detection. A publicly available dataset called Employment Scam Aegean Dataset (EMSCAD) was used to build the model. Pre-processing step had been applied before the choice and classification adoptions. Further, the findings presented the main features and important factors in detection purpose include having a company logo, title, In-balanced\_dataset, function and industry feature. We also applied Natural Language Processing for text features for building the model and the results showed the best accuracy of 98.12% obtained for using the text features.

1. **INTRODUCTION**:

In modern organizations, there's a good use of the web and social media deployed in employee recruitment. Recently, the cloud was integrated to the procedure of recruiting new members, where the managed cloud services or solutions are employed by human resource managers. Nevertheless, there are many violating risk threats increased by scams and frauds along side the wide interest and adopting such embedded software Cybercrime is one among this risky crime that face the planet and threaten the individuals and organizations security causing substantial losses.

Depending on the 2021 survey of cyber security, the world's estimate of cybercrime damage is about $6 trillion per annum. We also desperately need information protection to ensure confidentiality, honesty and availability (CIA) to fight against these crimes. It is often done through the implementation of known information security strategies like prevention, detection, and response. Online Recruitment Fraud(ORF) is considered as one type of cybercrimes that has appeared recently It violates the privacy and financial funds of people and organizations by exploiting Internet technology and web service. It allows non-legitimate users to harm the reputations of the organizations. Data mining methods have added to data analysis for prediction and detection of cybercrime. ORF interrupts the privacy of job seeker and bothers the reputation of organizations. Moreover, it causes loss of money for individuals. It occurs when criminals post fake Job advertisement misusing the automation recruitment to trap job seekers.

This is about the research that has observed the online fraud issues and developed a new solution for detection Some researchers like Vidros, Kolias, Kambourakis, & Akoglu (2017) added many features of ORF to the overall public dataset (EMSCAD). The researchers recommended to seek out a reliable detection model of ORF. Thus, we'd like to get a replacement of the reliable model to reinforce the performance of classification supported pre-processing and have selection phase.

The Employment Scam Aegean Dataset (EMSCAD) is a free dataset with 17,880 real-life job advertisements. The Dataset contains17,014 real and 866 fake job ads published between 2012 to 2014.Each record is represented as a group of structured and unstructured data. There are group of fields of various types and a binary class field indicating whether this job ad entry is fraudulent or not. Fields are often of 4 types, namely string as within the job title, HTML fragment just like the description, binary like the telecommuting flag, and nominal as in the employment type (e.g., full-time, part-time).

**Dataset Description is shown below**:



1. **RELATED WORKS**

There is rich literature about cybercrime detection models in several fields. However, there are only two studies one descriptive. One of them is an analytical analysis that has addressed the online recruitment abuse and scams. The related works often studied data processing techniques for various other detection purposes.

In 2019, a project titled “Fake News Detection Using Natural Language Processing” was carried out by Jasmine Vasandani. This project showed how NLP could be used to distinguish fake news from authentic posts on Subreddit. The goal of this project was to have the highest possible outcomes of True Positives and True Negatives minimizing the outcomes of False Positives and False Negatives. The classification models applied were the Logistic Regression model and the Multi-nomial Naïve-Bayes Classifier using the Count Vectorizer and the TF-ID Vectorizer for each classification model. Based on coefficient interpretability, the Logistic Regression model with the Count Vectorizer produced the best results with a training score of 96% and a test score of 87%.

1. **METHODS:**

This is a binary classification problem in which we will be predicting whether a job advertisement is fraudulent (t) or not fraudulent (f). The “fraudulent” column is our target (dependent) variable all other variables are the feature (independent) variables. We will be using feature selection for non-text data to select a subset of relevant features for use in model construction.For text Also we are using Count Vectorizer and the TF-ID Vectorizer to deal with text fields prior to model building. Various classification models were used such as K Nearest Neighbour, Logistic Regression, Decision Tree, Naïve-Bayes Classifier, Random Forest Classifier, Gradient Boosting Classifier and Support Vector Machine and check various classification metrics such as accuracy, precision, recall etc., to compare the models.

We divided the entire analysis process to following stages:

1. Data Cleaning
2. Data Exploration
3. Data Preparation/ Preprocessing
4. Feature Selection
5. NLP Processing
6. Modeling
7. Model Evaluation

Each step is explained below in details.

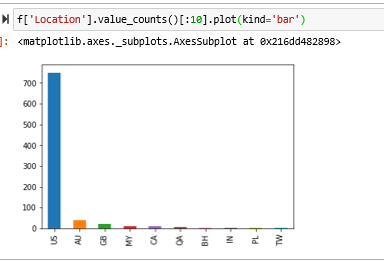
* 1. **DATA CLEANING**

Data cleaning is one of the most important phases of any machine learning project. Without proper data our model will not work well. The original data contains 18 features and 17880 observations and was cleaned following several procedures which are shown below;

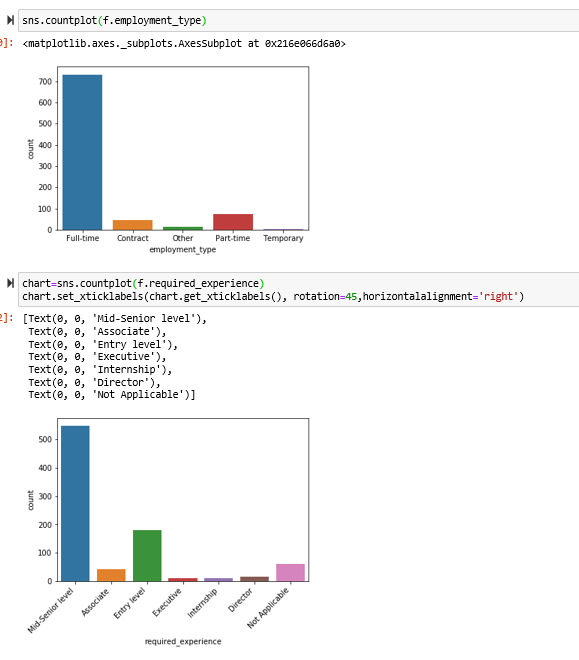
* **Splitting the “Location” column** using the criteria of delimiter (left most). This was done to make the categorization of the data in that column easier when label encoding is applied to the data. This split was done using Microsoft Excel.
* **Dropping the columns with more missing data:** In the data, it was discovered that there were features which had over 60% of its observations missing. The decision was made to drop those columns had insufficient data to be deemed useful for model building. The columns dropped were department and salary range.
* **Imputation of some of the columns with missing values using the maximum count values:** For some other features which had less than 60% of its values missing we used the maximum value counts (i.e. the mode of the categories in that column) to fill for the missing values. The columns affected were location, employment\_type, required\_education and required\_experience, function and industry as shown in the figures below;
* **Imputing some of the columns missing values as “Not specified’:** A new category called ‘Not specified’ was created and used to impute for missing values in the remaining features like requirements, benefits , company\_profile, description
  1. **DATA EXPLORATION**

Exploratory data analysis (EDA) is the technique to analyze data sets in which their features are summarized, most of the times using visual methods. A statistical model may be used or not, but essentially EDA is for seeing what we can derive from the data beyond the proper modeling or hypothesis testing problem. In this project, we will be isolating the two categories in the fraudulent column. Our focus is exploring the ‘t’ (job ads which are fraudulent) category in the dataset and all its observations with univariate analysis for each feature. Below are some interesting findings which are conspicuous from our analysis

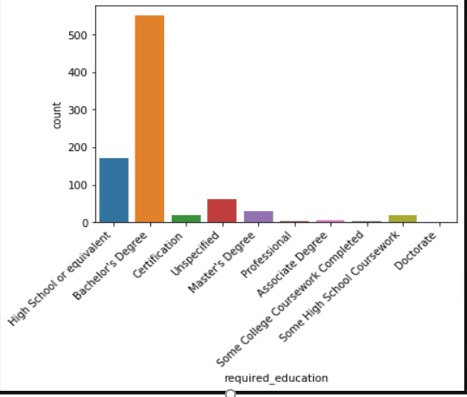
* There were more fraudulent job ads in the US than in any other Location by a very wide margin with over 700 ads being fraudulent as shown in Figure below



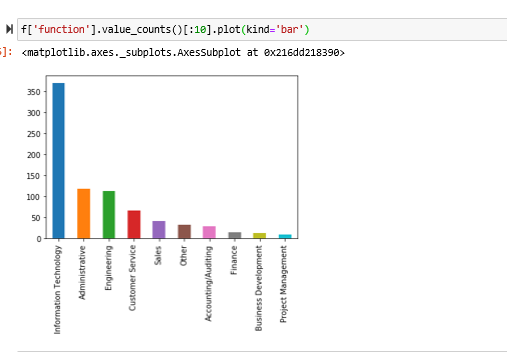
* The most common employment type being classified as fraudulent were full-time jobs with over 700 jobs and for Mid Senior Level experience time jobs with over 500 jobs being classified as fraudulent This is shown in Figure below.



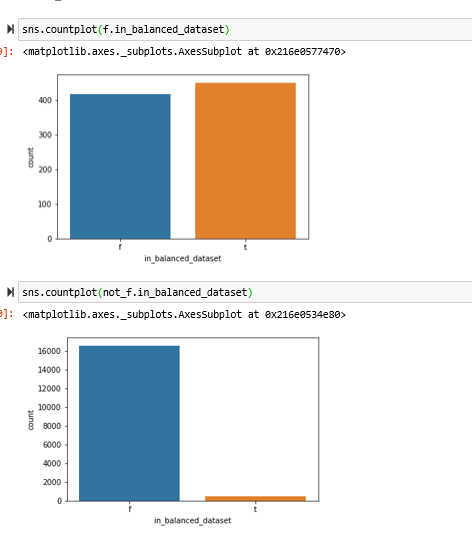
* The required education with the most fraudulent job is the Bechelor’s Degree with over 600 jobs classified as fraudulent as shown in the below Figure;



* The information Technology Industry has more fraudulent job ads than any other industry by some distance with over 400 jobs being classified as fraudulent. This is reflected in the visualization below;



* In the case of in\_balanced\_dataset column for fraud data there is a bit of balance but for not fraud data the false value is significantly higher than the true value.



**3.3 DATA PREPARATION/PRE-PROCESSING**

Data preprocessing could be a technique data mining where we transform the information into a clear format. "Real data is commonly imperfect, unpredictable and/or missing in certain behaviours or patterns and is feasible to contain several errors." So, data preprocessing has proven to be a way to unravel these challenges. In this project, we have been applied a few data preparation techniques some of which include;

* **Separating of text data from non-text in our cleaned data:** We have separated the text data from non-text data. The reason behind this is to be able to carry out further preprocessing on the text data using Natural Language Processing (NLP) for modelling. Also, to enable us build models for our non-text data and make comparisons between both approaches.
* **Label Encoding:** Label Encoding is the conversion of categorical labels into numeric form in order to make the labels machine-readable. Machine learning algorithms can then choose in an improved way how those labels should be operated upon. It is a vital pre-processing step for structured data in supervised learning. In the previous step we split the data into text and non-text data. We will be applying label encoding to the non-text data as they are all categorical features. For binary variables we used binary coding where we convert false to 0 and true to 1 and for other variables, we are using label encoding where we convert each values in a column to a number.
* **Preprocessing of text data:** Before applying the NLP method, it is necessary to clean the text. For that we select all the columns except the non-text columns and did cleaning for each column such as removing HTML tags, accented chars, special characters and merge all the columns to a single column prior to the process.

**3.4 FEATURE SELECTION**

In every dataset, it is fairly common to have columns that are nothing but noise. It is easier to get rid of these variables which will cost, particularly in large datasets, due to the memory space they occupy, the time and computational resources. The method where we select those features which contribute most to your prediction variable during which you're interested by automatically or manually is called the feature selection. The advantage for this process is that it reduces the complexity, it is easier to interpret, improves the accuracy, reduce overfitting and so on.

The various techniques we used for performing the feature selection are

**1. Univariate feature selection:** There is a statistically significant relationship between each feature and the target. they only consider each feature individually. Univariate tests are often very fast to compute, and don’t require building a model. On the other hand, they are completely independent of the model

**UFS:** Statistical tests can be used to select those features that have the strongest relationship with the output variable. The scikit-learn library provides the [SelectKBest](http://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectKBest.html) class that can be used with a suite of different statistical tests to select a specific number of features.

The **chi squared (chi^2)** statistical test for non-negative features to select 5 of the best features.

This technique is worked by selecting the best feature based on the univariate statistical test. The SelectKBest function removes all but the k highest scoring features from the dataset.

**2. Iterative Feature Selection :**

In this technique a series of models are built, with varying numbers of features. One of the examples for this technique is the Recursive feature elimination (RFE) which starts with all features, builds a model, and discards the least important feature according to the model. A new model is then created using all but the discarded feature, and so on until there is only a present value of features are remaining.

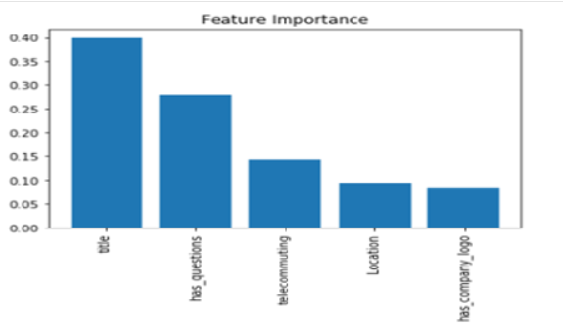
◦**RFE:** works by recursively removing attributes and building a model on those attributes that remain.  
 It uses the model accuracy to identify which attributes (and the combination of attributes) contribute the most to predicting the target attribute.

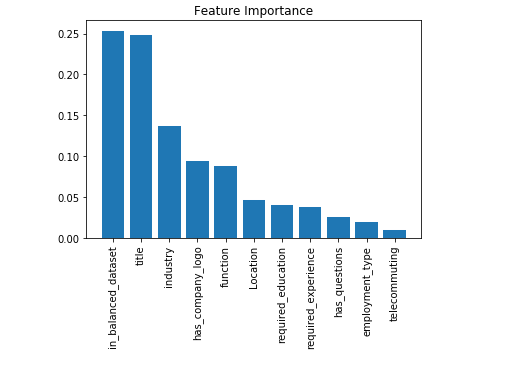
◦In the scikit-learn documentation RFE with the logistic regression algorithm to select the top 5 features. The choice of the algorithm does not matter too much as long as it is skillful and consistent.

**3. Model-Based Feature Selection :**

It uses a supervised machine learning model to judge the importance of each feature and keeps only the most important ones. Decision trees and decision tree–based models provide a feature\_importances\_ attribute, which directly encodes the importance of each feature The SelectFromModel class selects all features that have an importance measure of the feature greater than the provided threshold

As a result of all these techniques , we selected the 5 most important features such as **'title’, 'has\_company\_logo', 'industry', 'function',' in\_balanced\_dataset.** which occurs mostly.





* 1. **NATURAL LANGUAGE PROCESSING (NLP)**

It is how for computers to research , understand, and derive meaning from human language during a smart and useful way. Then use machine learning models to text and language with the help of this process. We used NLP on a text field to predict if the advertisement is a fraud or not. Most NLP algorithms are classification models, and that they include Logistic Regression, Naive Bayes, KNN, SVM, Random Forest, Decision Trees.

Statistical algorithms require data in mathematical form to train the machine learning models .The Text data needs to be converted into some numeric form to be used in training the machine learning models. We used two main approaches to convert text to numbers

1. Bag of words

2. TF- IDF (Term frequency and Inverse Document frequency)

**1. Bag of Words Model Approach**

It is a technique used to preprocess the texts to classify before fitting the classification algorithms on the observations containing the texts. By this method we preprocessed the text by translating it into a bag of words, which keeps a count of the entire occurrences of most often used words. We first preprocess the data, in order to convert text to lower case, removing the stop words , did Stemming which is used to reduce the number of inflectional forms of words appearing in the text using Porter Stemmer” method. The next step is to obtain the most frequent words in our text. For that we declare a dictionary to hold our bag of words and next we tokenized each sentence to words. We now test for each word in the sentence is it exists in the dictionary or not. If it exists, then we increment its count by and if not, we add it to our dictionary and set its count as 1. When processing large texts, the amount of words could reach millions. We don’t want to use all those words. Hence, we select some of the most frequently used words.

**2. TF- IDF Approach**

TF-IDF is the combination of two terms called Term frequency and Inverse Document frequency. Each term is calculated as shown below :

**TF = (Frequency of a word in the document)/(Total words in the document)**

**IDF = Log((Total number of docs)/(Number of docs containing the word))**

We are computing the TF-IDF score which is the product of TF and IDF and this will be implemented using TfidfVectorizer() function from the Sklearn library. It is a numerical measure used to determine the importance of the word is to a document in a collection of words or corpus. The value increases correspondingly to the number of times a word appears within the text but is offset by the word frequency inside the corpus. The concept behind the TF-IDF strategy is that terms appearing less in all documents and more in specific documents should contribute more towards classification

**3.6 MODEL BUILDING**

After completing the previous phases, the dataset is now able to build proposed model. Once the model is built it is used as predictive model. In our work, we make model based on different algorithms such as K Nearest Neighbour, Logistic Regression, Decision Tree, Random Forest algorithm etc. and compare it with other machine learning techniques. We build the model for both text and Non-text data. For non-text data we build model with all the features and also the 5 selected features which we selected during the feature selection techniques. For text data we build the model using both the Bag of Word and TF-IDF approach in order to check which method is better to predict the target variable All models received input features, which are then divided into the training and test set The test dataset is used for fraud prediction.

Different Models we used in our project are explained below:

* **K-Nearest Neighbors (KNN) Classifier**: K nearest neighbors is a simple algorithm which stores all available observations and classifies new cases based on a measure of similarity for example; distance functions. A KNN Classifier is a data classification model which attempts to determine what group a data point belongs by looking at the data points around it.
* **Logistic Regression:** It a Machine Learning algorithm which is applied to classification problems. It is a predictive type analysis model which is based on the idea of probability. Both Logistic Regression and Linear Regression model have similarities except that Logistic Regression uses a more complex cost function which is called the ‘Sigmoid function’ or the ‘logistic function’ rather than a linear function. The logistic regression hypothesis tends to limit cost function between 0 and 1
* **Decision Tree Classifier:** It uses a tree-like model of decisions and their possible outcomes such as the chance an event outcome, the resource costs, and utility. It is a way to show an algorithm that only holds conditional control statements. A decision tree classifier builds classification models in the form of a tree structure. This splits the data set into smaller subsets while, at the same time, a new associated decision tree is generated incrementally. The end product is a tree with decision nodes and leaf nodes.
* **Naïve Bayes Classifier:** A Naïve Bayes classifier is a probabilistic machine learning model that is applied to solve classification problems. The root of the classifier is based on the Bayes theorem. Using Bayes theorem, it is possible to find the probability of A happening, provided B has happened. Here, B represents the evidence and A represents the hypothesis. Here, the assumption is that the predictors / features are independent. This means that the presence of a feature does not have any effect on the other. Hence it is called naive.
* **Random Forest Classifier:** The Random forest classifier is a supervised learning algorithm that randomly generates and combines many decision trees into what we would consider as a “forest.” The aim is not to rely on a single decision tree model, but rather the collection of decision tree models to boost accuracy. The key difference between this approach and the standard decision tree classifier is that splitting nodes are generated at random to the root nodes function.
* **Gradient Boosting Classifier:** Gradient boosting classifiers are algorithms merge many weak learning models to produce a stronger predictive model. Decision trees are usually used when implementing gradient boosting. Gradient boosting models are becoming common because of their usefulness in classifying complex datasets.
* **Support Vector Machine (SVM) Classifier:** A support vector machine (SVM) is machine learning model that analyzes data for both classification and regression analysis. SVM is a supervised learning technique which examines data and groups it into one of two classes. An SVM outputs a map of the sorted data with the margins between the two as distant as possible. SVMs have been applied in text categorization, image classification, handwriting recognition and in the sciences.

**3.7 MODEL EVALUATION**

In order to find out the best models or methods it is very necessary to evaluate the model .We used different evaluation metrics such as accuracy, recall, precision,F1 score and can compare with different model in order to find out which model is better to predict whether the advertisement is fraud or not.

The explanation for each metrics is described below:

**Accuracy :** This is the number of correct predictions divided by the number of all samples.

**Accuracy = (TP + TN) / (TP + TN + FP + FN)**

**Precision :** It is used as a performance metric when the goal is to limit the number of false positives

**Precision = TP / (TP + FP)**

**Recall :** It measures how many of the positive samples are captured by the positive predictions:

**Recall = TP / (TP + FN)**

**F1 score :** It the harmonic mean of precision and recall:

**F1 Score = 2 \*(Precision \* Recall )/ (Precision + Recall)**

We also did the cross validation. It is a statistical method of evaluating generalization performance that is more stable and thorough than using a split into a training and a test set. Here the data is instead split repeatedly, and multiple models are trained. The most commonly used version of cross-validation is k-fold cross-validation, where k may be a user-specified number. In stratified cross-validation, we split the info such the proportions between classes are an equivalent in each fold as they're within the whole dataset.

1. **RESULTS**

To check whether there is any difference between the method we generate a hypothesis test .Since we build 7 models for both the methods the number of samples is small, so we need to do T Statistics. We took the mean and standard deviation of all the models in each method. We tabulated the values and we solved it as shown below.

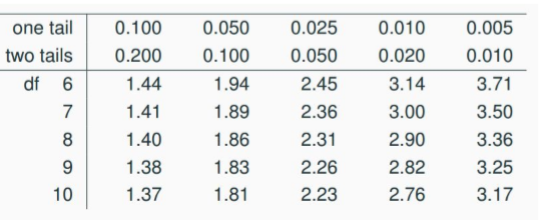
|  |  |  |  |
| --- | --- | --- | --- |
|  | **n** | **Mean** | **Standard Deviation** |
| Method 1 | 7 | 0.9483715 | 0.00368 |
| Method 2 | 7 | 0.9643787 | 0.00507 |

**Null Hypothesis :** There is no difference between the training accuracy for Method 1(Bag of Model Approach)and Method 2(TF-IDF Approach)

**Alternate Hypothesis :** There is difference between the training accuracy for Method 1(Bag of Model Approach) and Method 2(TF-IDF Approach)

H0 : 𝞵diff = 0

HA : 𝞵diff ≠

Tdf = (point estimate – null value) / SE

Point estimate = 𝞵diff = 0.0160072 ;

Sdiff = 0.00139;

df= n-1= 6

SE = Sdiff / √n = 0.0005253 ;

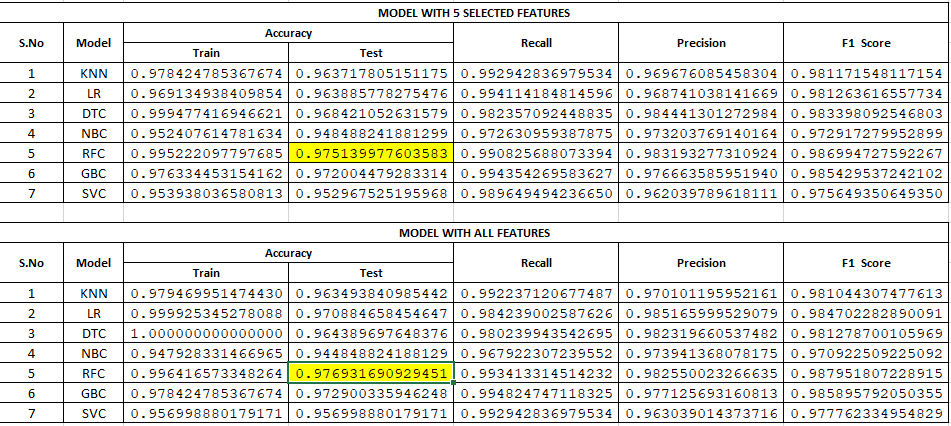
significance level(alpha= 0.05)

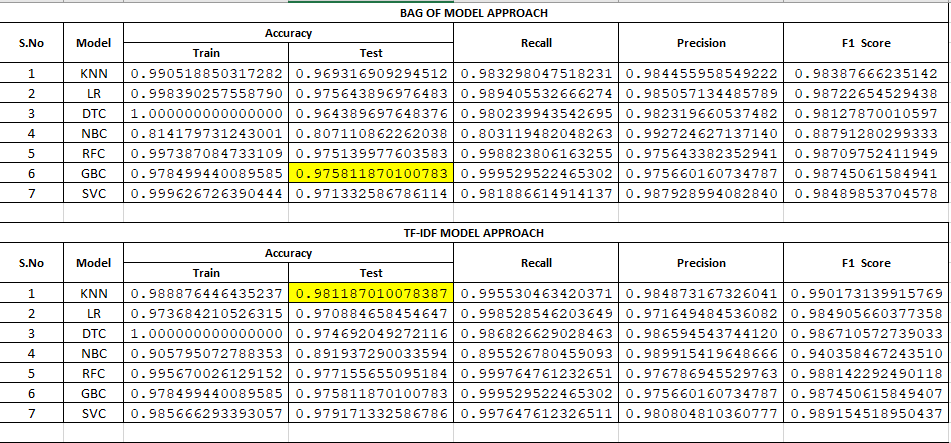
Tdf = (0.00139-0) / 0.0005253 = 2.646

Based on the T distribution table the value obtained is 2.45 < Tdf < 3.14, the value between 0.050 < p < 0.020. So, p value is less than alpha so need to reject Ho

**Conclusion :** The data provide convincing evidence that there is a difference between the training accuracy for Method 1(Bag of Model Approach) and Method 2(TF-IDF Approach)

Our results show that TF-IDF approach of NLP with the KNN Classifier applied provides the best accuracy with a score on the test set of 0.9811870. With this, we can safely assume that NLP is the better approach to the modelling due to a slightly better scores on the test sets compared to the traditional models with the non-text data.





1. **DISCUSSION**

Our main goal is to find out a good model to predict whether the advertisement is fraud or not and we achieved that while doing the project with best accuracy. We also checked some previous work which are somewhat in line with our findings. One of the main challenges is to clean the dataset properly so that we can make the model efficiently which we did correctly. There are a lot of missing values for variables which make problem when analysing the result so, we overcome it by imputing the values. Regarding the number of features to select for building the model for non-text data, we as a group come with different numbers and finally agreed to stick with 5 features. Since the number of observations is more (17,860), the entire processing of text fields took lot of time to create the corpus and thereafter to produce the result from different model which is one of the biggest challenges we faced during the project.

1. **CONCLUSION**

Online recruitment systems are a promising platform that a lot of companies and enterprises depend upon in their recruitment and hiring process. However, online recruitment systems are mistreated by criminals conducting scams. This report proposes a detection model for online recruitment fraud.

The proposed model used KNN Classifier with the text features after applying the NLP processing with TF-IDF approach and got the best accuracy with a score on the test set of 0.9811870. This research emphasized that the web recruitment contains similarities of previously well-studied scopes like email spam phishing, cyber bullying then forth. Furthermore, this project used the EMSCAD dataset which is the only free available dataset for this scope. The main contribution of this research is that it enhances fraud detection classifiers for online recruitment systems.

In future research, we plan to extend EMSCAD by concentrating on user activity client and network data as well as user-content-IP collision patterns and improve the ruleset. In addition, we'd like to use graph modelling to explore relations between fraudulent job ads, companies, to users. In the end, our goal is to suggest an appropriate method for the identification of job fraud for commercial purposes.

1. **CONTRIBUTIONS:**

Building and keeping an honest team is important to the success of project and makes our own job easier. Everyone is equally participated from the beginning to the end to complete the project successfully. We are dividing the project into different steps and assigned responsibilities as per below so that, all the team members will have the same amount of workload:

* Exploring Data & Model Building: Nanbal
* Data Cleaning and Feature Selection : Mehak Rajdev
* NLP Processing, Model Building, Evaluation and Hypothesis Testing : Bency Eldho

1. **REFERENCES:**

* <http://emscad.samos.aegean.gr/>
* Automatic Detection of Online Recruitment Frauds: Characteristics, Methods, and a Public Dataset by Sokratis Vidros, Constantinos Kolias, Georgios Kambourakis and Leman Akoglu
* An Intelligent Model for Online Recruitment Fraud Detection by Bandar Alghamdi, Fahad Alharby
* Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques by Hadeer Ahmed(&), Issa Traore, and Sherif Saad
* OpenIntro Statistics Third Edition by David M Diez, Christopher D Barr, Mine C¸etinkaya-Rundel
* Introduction to Machine Learning with Python by Andreas C. Müller & Sarah Guido
* <https://www.cnbc.com/2018/11/08/employment-scams-are-on-the-rise-in-tight-labor-market.html>
* [https://github.com/jasminevasandani/NLP\_Classification\_Model\_FakeNews#Methodology](https://github.com/jasminevasandani/NLP_Classification_Model_FakeNews)
* <https://www.bbb.org/scamtracker/us>
* <https://pythondata.com/comparing-machine-learning-methods/>
* <https://scikit-learn.org/stable/modules/feature_selection.html>
* <https://towardsdatascience.com/natural-language-processing-nlp-for-machine-learning-d44498845d5b>

1. **APPENDICES:**

Project code is attached to the report.