ONLINE JOB SCAM DETECTION USING SMART SYSTEM

A DISSERTATION

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE AWARD OF DEGREE
OF
MASTER OF TECHNOLOGY
IN
COMPUTER SCIENCE & ENGINEERING

Submitted by:

PANKAJ PANWAR
2K20/CSE/14

Under the supervision of
Dr. Rajesh Kumar Yadav
(Assistant Professor)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

MAY, 2022

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi - 110042

CANDIDATE'S DECLARATION

I, Pankaj Panwar, Roll No. 2K20/CSE/14 student of M. Tech (Computer Science and Engineering), hereby declare that the project Dissertation titled "ONLINE JOB SCAM DETECTION USING SMART SYSTEM" to the Delhi Technological University's Department of Computer Science & Engineering, I have submitted my Master of Technology thesis, which is original and not reproduced from any other source without due credit.Before this effort, no degree, diploma, associate ship, fellowship, or other equivalent title or honour could be awarded for it.

Place: Delhi

Pankaj Panwar

2K20/CSE/14

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi - 110042

CERTIFICATE

I hereby certify that the Project Dissertation titled **“ONLINE JOB SCAM DETECTION USING SMART SYSTEM”** which is submitted by Pankaj Panwar, 2K20/CSE/14 Department of Computer Science & Engineering, Delhi Technological University, Delhi Master of Technology students' project work is documented here in order to satisfy a portion of their degree requirements.Unless I'm mistaken, no component of this work has ever been submitted to this university or any other for consideration for an academic degree or diploma.

Place: Delhi

Dr. Rajesh Kumar Yadav
Assistant Professor
Delhi Technological University

Date: 26 May 2022

ACKNOWLEDGMENT

Many individuals and the organisation are needed for this project to be a success. I owe a debt of gratitude to each and every one of you who contributed to making this project what it is today. I express my sincere thanks to **Dr. Rajesh Kumar Yadav**, my project guide, for giving me the chance to work on this project with his help and supervision.My mentor's consistent support and encouragement have helped me understand that the process of learning is more important than the final outcome.I owe a debt of gratitude to the panel faculty members for their assistance, persistent monitoring, and encouragement to finish my work throughout all of the progress reviews.Throughout the process, they supplied me with fresh ideas, gave me the information I needed, and pushed me to finish.Thank you to everyone who has helped me along the way, including my classmates and my family.

PANKAJ PANWAR

2K20/CSE/14

ABSTRACT

In light of the growing world breaches of data that occur on a daily basis. With every successive day, the number of job seekers who fall prey to a becomes dramatically more and larger. The bulk of jobseekers are recruited by corporations and fraudsters use the approaches, with the majority of them coming from digital job-posting in Indeed website. We want to apply Machine Learning in the future to reduce the incidence of similar frauds. Candidates would be able to maintain their vigilance and make smart decisions, when necessary, hence lowering the amount of such frauds that take place in the first place. Natural language processing will be sometimes used investigate the attitudes and patterns in the job advertisement, and to do so (NLP). Later, models like Logistic Regression, Naïve Bayes and some other classification algorithms were used to classify the data into fraudulent or non-fraudulent job posting. Recently, some people have also used regular artificial neural networks and LSTM along with Bert or 200 dimension Glove for word embedding. All these models have achieved very good accuracies.We have implemented Logistic regression and Naïve Bayes as our baseline models to see how they perform on our unbalanced data.To make the data more balanced, we oversampled it and used it to test our baseline models.We noted that the model has improved its performance . To improve the accuracy we have chosen to work with CNN models on our oversampled data and used Google's pre-trained model Glove 300d to perform word embedding. We have evaluated all our models and found out that CNN model works the best by achieving the highest accuracy.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

1. NLP: Natural Language Processing
2. CNN: Convolutional Neural Network.
3. LSTM :Long Short Term Memory.
4. TF-IDF: Term Frequency Inverse Document Frequency.
5. POS : Part of Speech
6. EMSCAD : Employment Scam Aegean Dataset.
7. SVM : Support Vector Machine.
8. BOW : Bag of Words.

CHAPTER 1

INTRODUCTION

1.1 A BRIEF OVERVIEW

Humans can interact with each other with their natural language. They can speak, transfer their ideas, and understand each other's opinions, but the machine cannot do so. The machine systematically needs instructions to work. We need to make the computer understand the natural language so that humans and machines can interact with each other.

In Computer Science we have a Area of Artificial Intelligence Under which we have Natural Language Processing that makes Humans and machine to have interaction with one another. It involves an interaction between natural language and machine language. It teaches the computer or device how to communicate with humans using speech or text. It makes the machine read, understand, and generate meaning for human language.Speech recognition, comprehension, and generation are all part of natural language processing.

1.2 PRE-PROCESSING

Machine Learning, as we all know, requires numeric data.To convert text to a numeric vector, we used encoding techniques (BagOfWord, Bi-gram, n-gram, TF-IDF, Word2Vec).However, before encoding, we must first clean the text data; this process of preparing (or cleaning) text data before encoding is known as text preprocessing, because it is the way of resolving NLP problems.

1.2.1 tokenization

Tokenization is the technique of making the substituting for a token for the delicate data which can be crucial for any entity i.e credit card number..Sensitive data must still be safely maintained in a centralised location for future reference and must be surrounded by robust safeguards. As well as the security of the sensitive data themselves, a tokenization approach's security is determined by the method and methodology used to create the surrogate value and map it to its original value.

1.2.2 stop word removal

In many NLP applications, removing stop words is a frequent strategy.The objective is to eliminate all terms from the corpus.The majority of stop words include articles and pronouns in them.. Some NLP tasks, such as knowledge discovery and segmentation, have no meaning for these words, signalling that they are not especially discriminative..Stop word removal, on the contrary, will have little influence on some NLP applications.In a Language Stop Words are typically a well-curated list of terms that appear frequently across corpora..Stop word lists are available online for most languages.

1.2.3 lemmatization and stemming

It is possible to remove affixes from words in order to get at the most basic form.It's like chopping all of a tree's branches down to the trunk and putting them back together.As an example, we may look at the root of the terms eat, eat, and eaten.. Search engines utilize the stemmingtechnique for the indexing words.As a result, rather of keeping all variations of a word, a search engine might just save only the tems.Stemming decreases the amount of the index and, as a result, increases retrieval performance.The stemming process is similar to lemmatization.Instead of a root stem, which is the result of stemming, lemmatization produces a "lemma," which is a "root word." .We will receive a legitimate term that signifies the same thing after lemmatization.The WordNetLemmatizer class in NLTK is a lightweight wrapper over the WordNet corpus.To find a lemma, this class employs the WordNet CorpusReader class's morphy() method.

1.2.4 part-of-speech tagging

Classifying words in a text (corpus) according to their meaning and context is a common strategy in Natural Language Processing (NLP).We know that every word has a lexical phrase associated with it, but as the database increases, remembering these entire words for every text analysis becomes a chore.We employ a shorthand notation known as "tags" to express the categories since the POS tags are utilised to characterise the lexical phrases found in our text..

1.2.5 rules-based system

Rules created by humans are used to store, organise, and alter data in a rule-based systemIn this respect, it is like a human brain.When using rule-based systems, facts or data must be collected and the rules for manipulating them must be established. These sentences are referred to be "If statements" because they seem to mimic the form of "IF X occurs THEN do Y."

The stages could be described this way:

- The data or new business event comes first.
- Analyze, and the system compares the data to its own set of criteria.
- Automated measures for follow-up are also an option.

1.2.6 introduction to semantic analysis

The concept of natural language is understood in Semantic Analysis.Understanding Natural Language may appear to be an easy procedure to us as humans. However, because to the complexity and subjective nature of human language, machine translation is a difficult task.To understand a text, semantic analysis takes into consideration context, phrase logical structure, and grammatical roles.Natural Language Semantic Analysis is divided into two sections:

1. Lexical Semantic Analysis is the process of evaluating the meaning of each individual word in a text.It basically means obtaining the dictionary meaning of a term in the text.

2. Understanding the context of each and every word in the text is critical, but it was not enough to completely appreciate the context of the text.

Take a look at the following two sentences:

Sentence 1: GeeksforGeeks is popular among students.

Sentence 2: GeeksforGeeks adores scholars.

Despite the fact that both statements 1 and 2 have the same basic terms (student, love, geeksforgeeks), their meanings are entirely different.As a result, Compositional Semantics Analysis seeks to understand how individual word combinations build text meaning.The basic phases of Semantic are as follows:

Understanding a sentence's intended meaning requires analysis.:

- Word Sense Disambiguation.
- Relationship Extraction.

1.2.7 word sense disambiguation:

According to Natural Language, a word like this might have a different meaning according on the context in which it's used.The strategy to understanding the context of a phrase based on its context in a text in Word Sense Disambiguation. For example, the term "bark" can apply to both a dog's bark and the outer covering of a tree. In the same way, the term "rock" might imply "a stone" or "a kind of music," depending on the context and use in the passage.To put it another way, the term Word Sense Disambiguation refers to the capability of a computer to discern the context of an expression based on its use and context.

1.2.8 relationship extraction:

Extraction of relationships is an essential part of Semantic Analysis as a whole.

First, the different components of the text are recognised and then the linkages between them are extracted.

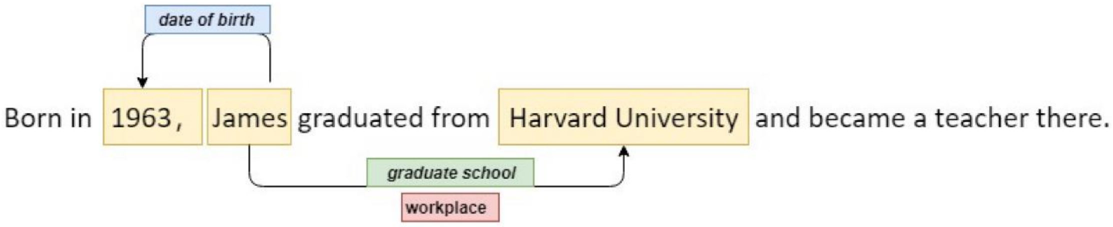

date of birth
Born in 1963, James graduated from Harvard University and became a teacher there.
graduate school
workplace

Figure 1.1 Relationship Extraction Example

Natural Language processing necessitates a thorough examination of the following aspects of Semantic Analysis:

Hyponymy: A hyponymy is a sentence that is an instance of a generic word.The class-object analogy can help you understand them.Color, For instance, is a hypernymy, yet grey, blue, red, and so on are hyponyms.Two or more words that have the same spelling but have different meanings are called homonomy.There are homonymous words that have the same spelling but distinct meanings, such as 'rose,' which may mean "the past form of rising," or "a flower."

Synonymy: occurs when 2 or more than 2 lexical expressions with different spellings have the same or similar meaning. As an example, (Job, Occupation), (Large, Large) (Stop, Halt).A pair of lexical phrases having opposing meanings that are symmetric to a semantic axis is referred to as an Antonymy.

Polysemy: Many words in the English language have the same spelling but different meanings, which is known as polysemy.Homonymy, on the other hand, requires that the words' meanings be closely connected, which is not the case here.The lexical word "man" is a polysemy since it has several meanings.An instance of Meronomy is one in which one word serves as the building block of a broader concept.

1.2.9 meaning representation

While it is relatively easy for humans to interpret the meaning of textual material, this is not the case for robots.As a result, in order to comprehend the meaning of the text, robots tend to express it in certain forms.Meaning representation refers to the formal framework utilised to grasp the meaning of a document.It is critical to understand the building components of such representations in order to achieve Meaning Representation in Semantic Analysis.The following are the fundamental units of semantic systems:Entity: An entity is a specific unit or individual, such as a person or a location.For example, GeeksforGeeks, Delhi, and so on.A concept can be thought of as a generalisation of entities.It includes a wide range of separate components.For instance, Learning Portals, Cities, and Students.Relations: Relations aid in the formation of relationships between distinct entities and concepts.For instance, 'GeeksforGeeks is a Learning Portal,' 'Delhi is a City,' and so on.Predicate: Predicates reflect the sentence's verb structures.

1.2.10 approaches to meaning representations:

Now that we've covered the fundamentals of Meaning Representations, let's glance for one of the more prominent techniques to meaning representation:

1. First-order predicate logic (FOPL)
2. Semantic Nets
3. Frames
4. Conceptual dependency (CD)
5. Rule-based architecture
6. Case Grammar
7. Conceptual Graphs

1.2.11 semantic analysis techniques

A number of approaches to semantic analysis may be used, depending on the final aim.

In Semantic Analysis, two of the most prevalent methods are as follows:

1.2.12 text classification

In-Text Classification is used to categorise text based on the information that may be gleaned from it.Sentiment Analysis, for example, prefers to categorise texts based on their primary emotional content.It's a great tool for enhancing customer evaluations.Our data is organised into categories under Topic Classification.Finding out whether anything is in Physics, Chemistry, or Mathematic.We use Intent Classification to try to figure out what a text message is trying to say.For example, determining if a customer service email is a question, a complaint, or a request.

1.2.13 text extraction

Textual information may be extracted via In-Text Extraction.Using Keyword Extraction as an example, we are attempting to identify and extract the most important phrases that accurately characterise the full text.The goal of Entity Extraction is to extract all of the documents' entities.

1.2.14 significance of semantics analysis

Semantic analysis is a critical component of Natural Language Processing (NLP).It is vital for organisations to gain insights from textual information in this increasing era.Semantic Analysis helps robots to understand texts and extract important information, giving critical data while reducing human labour.Chatbots, for example, employ Semantic Analysis to let them answer to user queries without the need for human intervention.

1.3 FEATURE EXTRACTION TECHNIQUES – NLP

NLP techniques for analysing text similarities using basic feature extraction techniques. When it comes to teaching computers how to understand and utilise large amounts of human language data, this field of computer science is known as Natural Language Processing (NLP).NLP refers to a computer's ability to recognise natural language.To create output for the test data, machine learning algorithms use a pre-defined set of features from the training data.In order to use machine learning techniques in language processing, raw text must be provided as a training set.We need particular Feature Extraction Methods in order to transform text into vectors of features.The following are a handful of the most used methods for extracting features.:

- Bag-of-Words
- TF-IDF

1.3.1 bag of words:

The fundamental method for transforming tokens together into set of characteristics is known as Bag-of-Words.In BOW Model every word is utilised as for the training of the Classifier , which is used for document categorization. Words like "great," "exceeding," and "amazing" are often used to describe a favourable review, whereas "annoying" and "bad" are more commonly used to describe a negative review.BoW models are built in three stages:

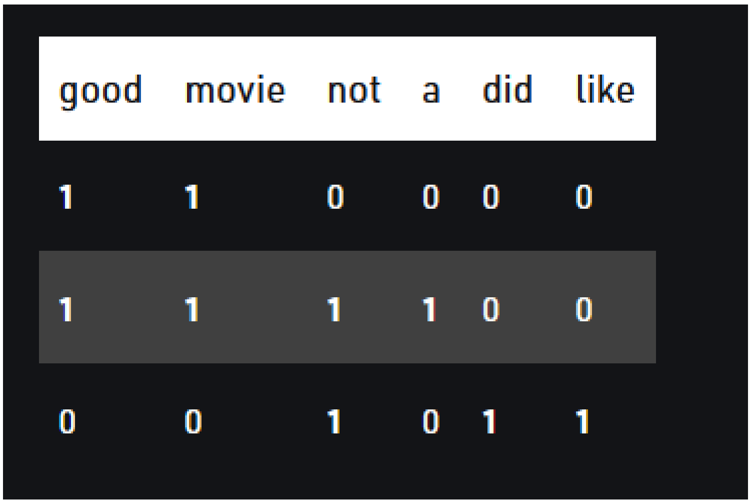
- Transforming everything to lower case and deleting any extra punctuation or symbols is step one of the text processing process.
- A vocabulary including all of the corpus's unique terms must be created next. Hotel reviews are often written in the third person.

Consider the following three examples of reviews:

1. good movie
2. not a good movie
3. did not like

Now we'll create a vocabulary using all of the unique terms from the prior reviews, which will look like this:
{good, movie, not, a, did, like}

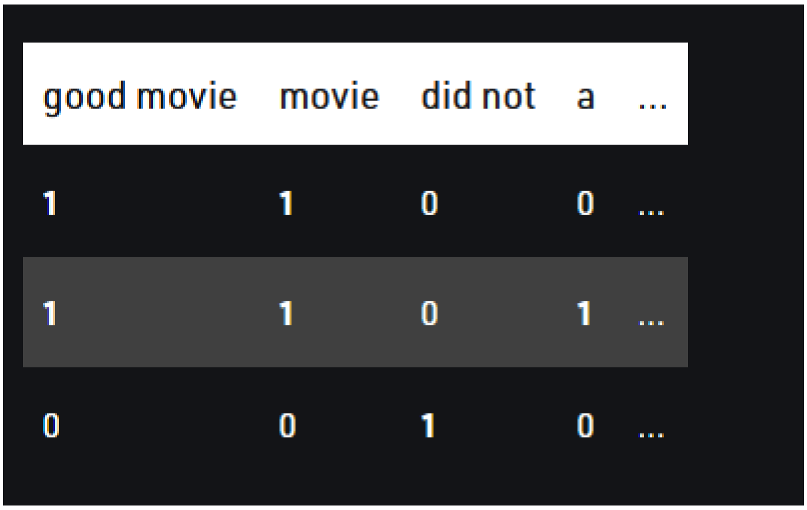
We generate a vector of features in the third phase by allocating a separate column to each term and a row to each review which is known as Text Vectorization. In the vector of Features each item indicate each term in the review. If the term appears in the review, we assign a 1 or else, we assign a 0.The matrix of characteristics for the example above will be as follows:



good	movie	not	a	did	like
1	1	0	0	0	0
1	1	1	1	0	0
0	0	1	0	1	1

Figure 1.2 Matrix of characterstics

Because we construct a matrix of tokens in random order, the order of occurrences of words is erased while using this method.Instead of individual words, we can explore N-grams (mostly bigrams) to solve this problem (i.e. unigrams).This keeps the original word order.The table below shows the results of evaluating all probable bigrams from the reviews:



good movie	movie	did not	a	...
1	1	0	0	...
1	1	0	1	...
0	0	1	0	...

Figure 1.3 Bigrams for the review example

This database, however, will be rather large because all possible successive word pairs might result in a large number of potential bigrams.FurthermoreIt may be difficult to compute with a big vocabulary if N-grams are used since the matrix will be huge and sparse.As a result, certain N-grams will have to be omitted from the text.All articles have N-grams with a high frequency of occurrence, so we can always rule them out.

Similarly, because to their extreme rarity, low frequency N-grams might be omitted from consideration (appearing in only one or two reviews). These N-grams are almost always typing mistakes..Furthermore, there are a few N-grams in our corpus that are incredibly uncommon but can indicate a particular issue. Say, for example, that a user's review says, "Wi-Fi fails often."However, the frequent N-gram "Wi-Fi breakdowns" point to a core problem that must be addressed..Because the frequency of such N-grams is so low, our BoW model would miss them.Another model, the TF-IDF Vectorizer, is required to address this sort of problem, which we will look at next.

1.3.2 tf-idf vectorizer

TF-IDF has uncovered an issue that isn't frequent in our corpus but is really significant.Words appear more often in papers with higher TF–IFD scores, and the number of documents featuring them reduces as the number of occurrences rises.It consists of the following two sections:

1. Term Frequency (TF)
2. Inverse Document Frequency (IDF)

1.3.3 term frequency(tf)

The frequency of a phrase in a text is determined by the term.It may be likened to the probability of finding a word in a manuscript.When looking at a review, it evaluates how many occurrences of a word wi there are in comparison to its total word count.To put it simply, it looks like this:

$$tf(w_i, r_j) = \frac{No.\,of\,times\,w_i\,occurs\,in\,r_j}{Total\,no.\,of\,words\,in\,r_j} \tag{1.1}$$

Log normalisation is a distinct method of computing tf. It's also written as

$$tf(t,d) = 1 + \log f_{t,d} \tag{1.2}$$

1.3.4 inverse document frequency(idf)

Every document in the collection is analysed by the IDF to see whether a certain phrase is unusual or frequent.Words having a high IDF score, or keywords that occur only in a limited number of texts throughout the corpus, are highlighted.In order to get the overall term's logarithm, you divide the total number of documents in the collection by the total number of documents that include the word t.

$$idf(d, D) = \log \frac{|D|}{\{d \epsilon D : t \epsilon D\}} \tag{1.3}$$

Where,

$f_{t,D}$ document D's total number of instances of the term "t".

$|D|$ it's the total number of items in the collection.

{d∈D : t∈D is the number of items in the collection that include the term "t" anywhere in the text of the item.

The magnitude of IDF >= 0 because the proportion within the IDF's formula must always be >= 1.The ratio within the logarithm becomes 1 when a term occurs in a large number of papers, yet the IDF is getting closer and closer to 0. combination of TF and IDF results in the TF-IDF hybrid.It may be stated as follows

:

$$tfidf(t, d, D) = tf(t, d) * idf(d, D) \tag{1.4}$$

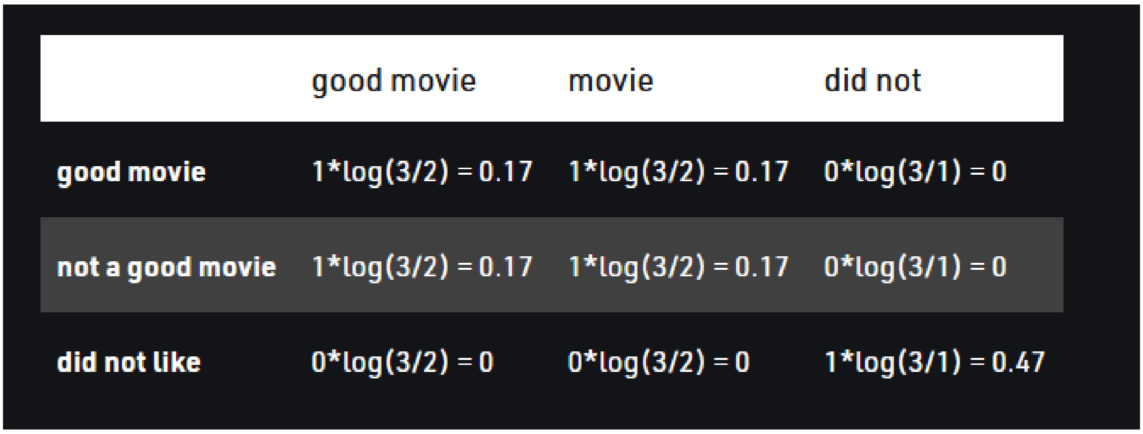
A higher TF-IDF score is assigned to a word that appears in a large number of documents but appears in a smaller number of documents within the corpus.The TF-IDF value is getting closer to 0 for a term that appears in almost all of the texts, which means that the IDF value for it is becoming closer to 0.

If both the IDF and TF values are high, the TF-IDF score will also be high; this indicates that the term is unusual within the text as a whole yet frequent inside it.

Let’s take the same example to understand this better:

1. good movie
2. not a good movie
3. did not like

This illustration has many documents, one for each assertion.Through application of the bigram model, we arrive at the final TF-IDF values for each bigram.



	good movie	movie	did not
good movie	1*log(3/2) = 0.17	1*log(3/2) = 0.17	0*log(3/1) = 0
not a good movie	1*log(3/2) = 0.17	1*log(3/2) = 0.17	0*log(3/1) = 0
did not like	0*log(3/2) = 0	0*log(3/2) = 0	1*log(3/1) = 0.47

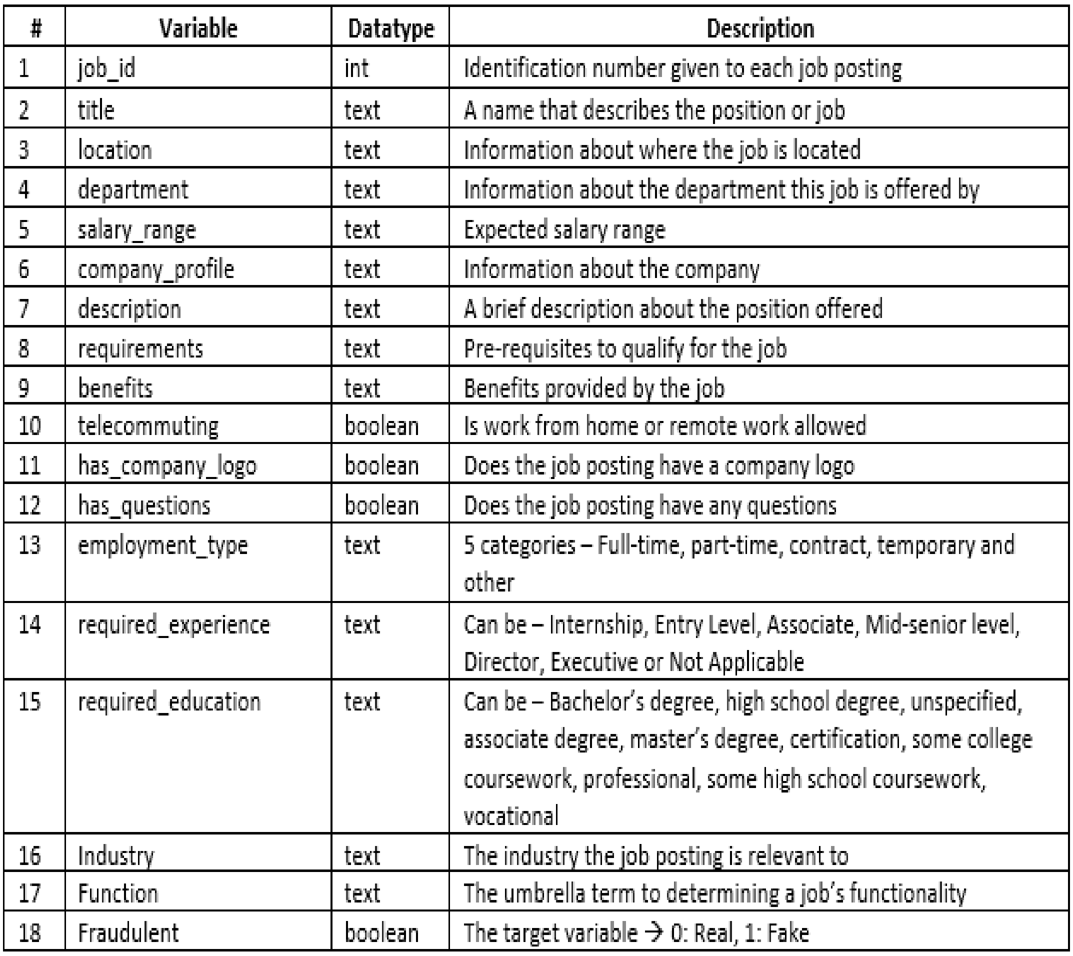
Figure 1.4 TF-IDF Score for Bigrams

In this case, we can see that the bigram is not as rare as other tokens, and so has a greater tf-idf value. To summarise, while Bag-of-Words is among the most essential approaches in feature extraction and text vectorization, it falls short of capturing some difficulties in the text. However, A feature extraction technique, TF-IDF Vectorizer, tackles this issue by catching some of the severe defects that are not widespread in the corpus.

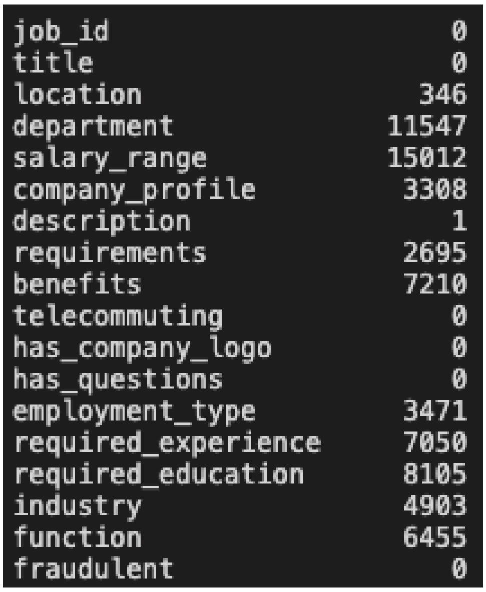
1.4 DATA SOURCE

Publicly available dataset comprising 17,880 real-life job advertisements, the Employment Scam Aegean Dataset (EMSCAD) is intended to provide the scientific community with an overview of the problem and act as a valuable testbed for scientists working in the area. MDPI Future Internet Journal has published our first article online.In We chose to utilise the EMSCAD (EMployment SCam Aegean Dataset) dataset, which comprises real-world employment adverts posted by companies [2]. In the EMSCAD Dataset, there are 17,880 real job advertisements and 866 bogus ones.There is a lot of information contained in the data: the job ID, job title, the company's name and location (if applicable), the company's profile, and so on. This dataset comprises categorical and descriptive pre-processed to make it usable for model training. LinkedIn, for example, was used as a source of relevant and real-world market exposure for our research.The web scraping was used to do this.Converted from json to csv, the data is now available.Integer, binary, and text datatypes are all used to store the data.Below is a list of all the variables.Because most of the data is either Boolean or text, there is no need for a summary statistic.It's a job id, which has no significance in this situation.Null values are discovered through additional investigation of the dataset. Even on the most reputable job-hunting websites, you'll find a slew of legitimate job listings.The so-called recruiters, on the other hand, begin asking for money and bank account information as soon as candidates have been selected.A large number of applicants fall prey to their ploys, resulting in both financial and employment losses.Consequently, it's a good idea to figure out whether or not an employment posting on the website is genuine or a scam.It's very difficult, if not impossible, to make the correct identification by hand.We can train a model for fake job categorization using machine learning.It may be taught to distinguish between legitimate and fraudulent job postings using past examples. The (EMSCAD) will be used to develop a machine learning classifier that will be able to recognise fraudulent employment adverts.To begin, we will create a visual representation of the insights gained from both the false and the actual job post.

Table I   Table of Variables



#	Variable	Datatype	Description
1	job_id	int	Identification number given to each job posting
2	title	text	A name that describes the position or job
3	location	text	Information about where the job is located
4	department	text	Information about the department this job is offered by
5	salary_range	text	Expected salary range
6	company_profile	text	Information about the company
7	description	text	A brief description about the position offered
8	requirements	text	Pre-requisites to qualify for the job
9	benefits	text	Benefits provided by the job
10	telecommuting	boolean	Is work from home or remote work allowed
11	has_company_logo	boolean	Does the job posting have a company logo
12	has_questions	boolean	Does the job posting have any questions
13	employment_type	text	5 categories – Full-time, part-time, contract, temporary and other
14	required_experience	text	Can be – Internship, Entry Level, Associate, Mid-senior level, Director, Executive or Not Applicable
15	required_education	text	Can be – Bachelor's degree, high school degree, unspecified, associate degree, master's degree, certification, some college coursework, professional, some high school coursework, vocational
16	Industry	text	The industry the job posting is relevant to
17	Function	text	The umbrella term to determining a job's functionality
18	Fraudulent	boolean	The target variable → 0: Real, 1: Fake



```
job_id                     0
title                      0
location                 346
department             11547
salary_range           15012
company_profile         3308
description                1
requirements            2695
benefits                7210
telecommuting              0
has_company_logo           0
has_questions              0
employment_type         3471
required_experience     7050
required_education      8105
industry                4903
function                6455
fraudulent                 0
```

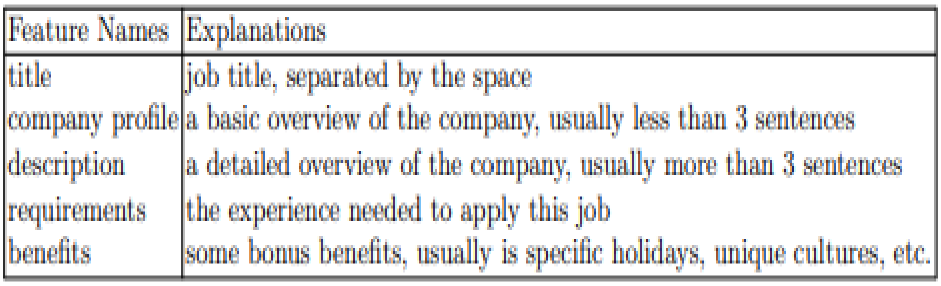
Figure 1.5   Missing values

There are several missing values in variables like department and salary range.These columns will not be analysed any further.Due to the fact that these job postings were culled from several countries, it was revealed after a first assessment of the dataset that they were written in numerous languages.This project relies on data from the United States, which accounts for more than 60% of the dataset.Having all the data in English makes it much easier to understand.

1.5 EXPLORATORY DATA ANALYSIS

we will explore the some basic statistics in this dataset. Theresponse variable (indicator of fradulent) is extreme unbalanced, which contains17200 non-fraudulent and 800 fraudulent.The Fig.1 indicates that the salary range, department info, required experience and education, function, and industry have a lot of missing values. As ourobjective is to mainly deploy the NLP in the text related features, we will drop these columns and keep only the features listed in the Figure 1.4 in our models'training.

Table II Feature Description



Feature Names	Explanations
title	job title, separated by the space
company profile	a basic overview of the company, usually less than 3 sentences
description	a detailed overview of the company, usually more than 3 sentences
requirements	the experience needed to apply this job
benefits	some bonus benefits, usually is specific holidays, unique cultures, etc.

The Table 1.2 takes a brief look on the required experience level. Among all the job posts, we can notice that most of them requires the entry level for the graduates, or the mid-senior level and associate level for those employees leave their old companies. This is very intuitional since these three groups do play the main role in the online job applications.Besides, are most often used words generated by WordCloud(MIT licenced). Even though there doesn't exist a clear pattern among these two

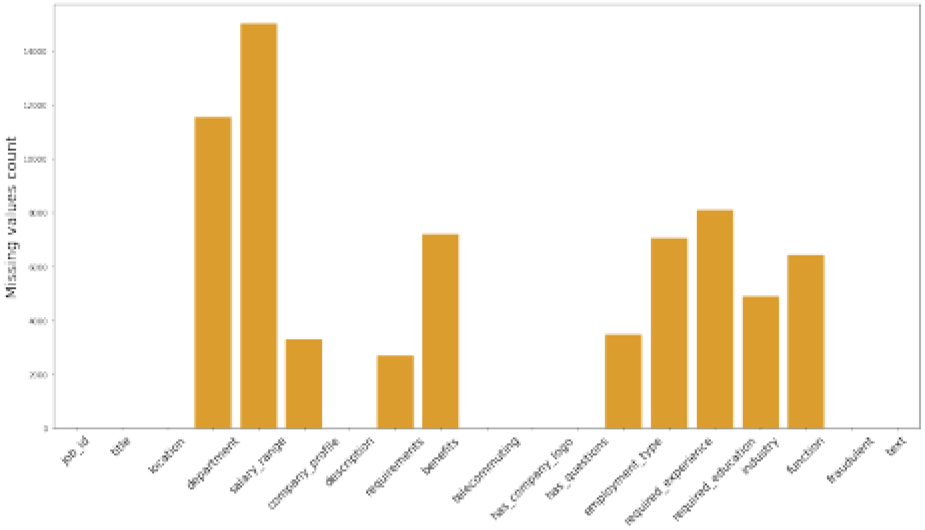


Figure 1.6   Missing values in different features of Dataset

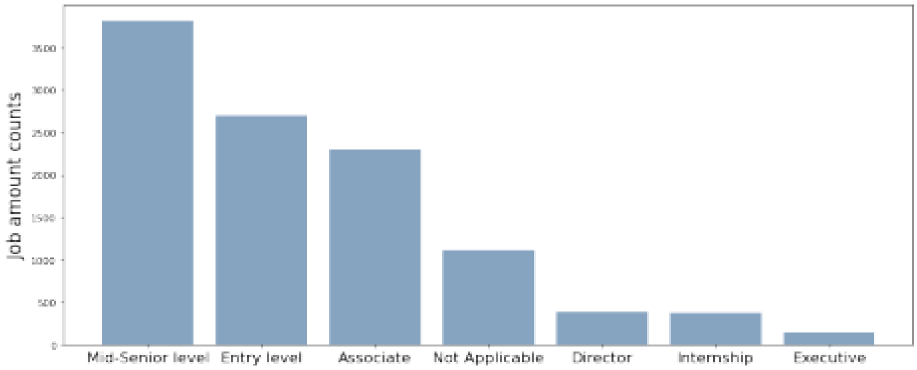


Figure 1.7 count of Different Job Designation

1.5.1 Correlation Matrix

For this project's visualisation, we begin by creating a correlation matrix to see how the numerical data is related.

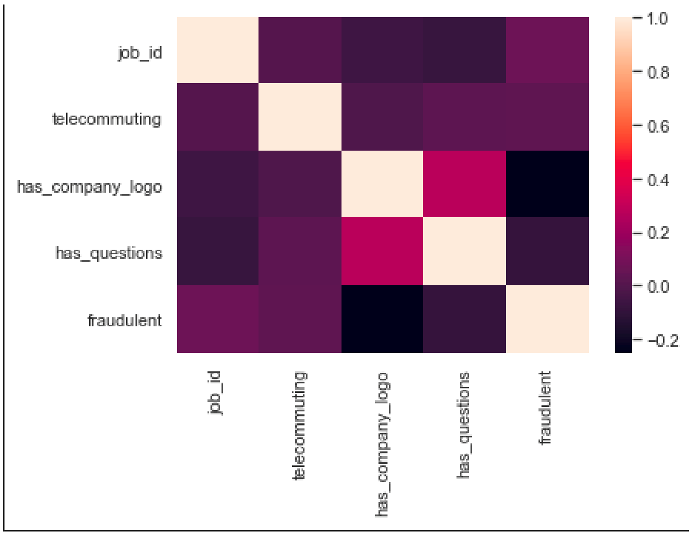


Figure 1.8 Correlation Matrix

There are no statistically significant relationships between the numerical values in the correlation matrix.In the Boolean variable telecommuting, however, a surprising trend was seen.Zero for each of these factors indicates a 92% probability that an item is not authentic.

1.5.2 Word Cloud

Word Cloud is a simple yet powerful visual representation feature for text analysis that presents the most regularly uttered words in bigger, bolder letters and with a variety of colours.In other words, the less important a phrase is, the more diminutive it is.

Word Cloud Applications:

1) Top Hashtags on Social Media (Instagram, Twitter): Because social media is trending across the world for the newest news, we can collect the most trending keywords that people use in their posts from there.

2) It's possible to discover keywords in the headlines of news stories and extract the most popular themes, which yields the desired result: a list of popular media subjects.

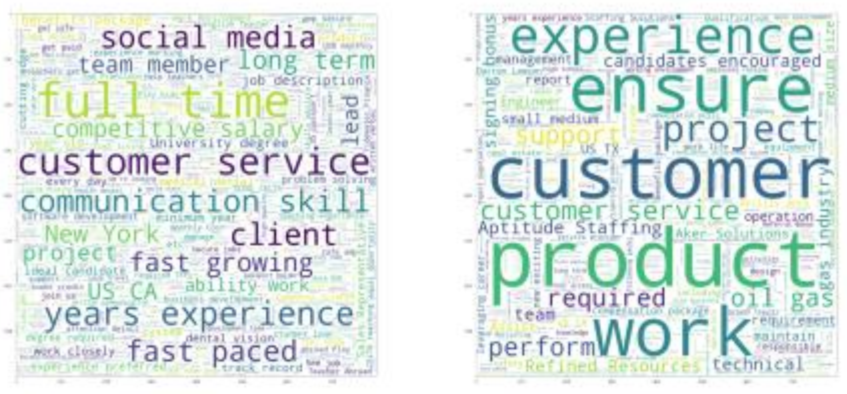


Figure 1.9 Word Cloud of Job Description

WordCloud figures, we can still notice that the fraudulent job posts tend to have more words about promising such as ensure, aptitude, project, which usually gives the candidates a sense of easy to get in. In contrast, the real job posts tend to have more diverse words covering in team work, communication, location, social media, work pace, etc. sizes don’t vary too much, while the fraudulent key words usually have very large proportion for the specific words as the ensure, customer, product have very large sizes.

The text-based categories are integrated into a single field in order to further expand the analysis of text-related data.Combined are the fields of title and location with those of company profile and description with those of prerequisites, benefits, required experience and required education, as well as industry and function.The difference between actual and fraudulent occupations may be shown in a histogram that depicts the character count.Even if both actual and fraudulent jobs have a comparable character count, the frequency of real jobs is greater. This project's implementation is shown in the following diagram.Text, numeric, and y-variables make up the dataset's three sections.For subsequent analysis, the text collection is transformed into a term-frequency matrix.The datasets are then divided into two sets, one for testing and the other for training, using sci-kit learn.Naive bayes and another model, SGD, are trained on the dataset's train set, which is 70% of the dataset's total.It is only when both numerical and text-based models say the same thing about the same piece of data that the job ad is considered to be fake.

CHAPTER 2

PRIOR WORK

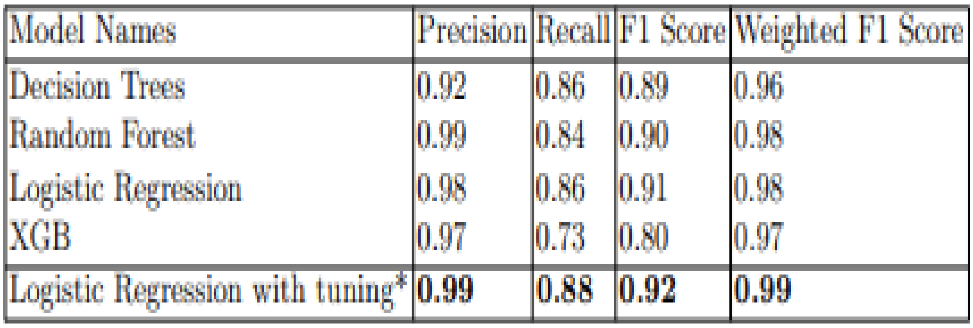
Preprocessing and stemming of words to remove stop words and special characters has been done by Kaggle developers before; however, this is no longer necessary.Later, TF-IDF or Count vectorizer was used to transform the text data into word count vectors or word frequency vectors.Numeric data is the only kind of data that machine learning algorithms could grasp.Logistic regression, Nave Bayes, and other classification techniques were employed to identify fraudulent and non-fraudulent job postings from the data.Others have come up with artificial neural networks and SVMs for predicting the target variable's class.Pre-trained models like Bert or 200d Glove were utilised for word embedding..

We have also come across some works that used Keras framework. . We have used TF-IDF vectorizer on oversampled data and implemented Baseline models on this oversampled data which was not done previously. We also used Glove 300d instead of Glove 200d ,100d or Bert for word embedding and also included embedding layer with convolutional one dimensional layers in models as this was not done earlier. Our main work that distinguishes our work from others is implementing SVM model on oversampled data. We will see how this was done and what results we got in the upcoming sections.

2.1 TRADITIONAL MACHINE LEARNING MODELS

Researchers have previously looked at the following models.Data mining techniques such as decision trees, random forests, logistic regression, and XGB are all examples of this.. we can notice that the among these models, the logistic regression has the best performances in all the metrics, and is less resource demanding in the computation compared to the models like random forest. Thus, we further tune it by the hyperparameter optimization. We use a custom grid search function to find out the optimized hyperparameters, which are l2 penalty, liblinear solver, and C of 4.0.We notice that the recall score, which is mainly affected by the unbalanced classes amounts as the fraudulent data has only 800. To alleviate this effect, we experiment three methods. First, we adjust the prediction threshold on the ROC curve in Fig. 4. Under the threshold of 0.01, the maximum recall value for the True Positive can be the 0.94 and the False Positive is 0.05. However, the threshold is way too low that we don’t think it would generalize in other datasets. Thus, we average the optimized AUC thresholds for both the precision and the recall score, and set the tuning threshold as 0.35.Second, we use the downsampling method by randomly sampling 2000 from the total 18000 non-fradulent dataset. We experiment several downsampling size such as 1000, 5000, 10000 for the non-fraudulent dataset, while the size 2000 tends to have the most balanced effect on the recall and the precision. We notethat the balanced dataset still does not contain 1:1 of these two classes samples since downsampling to around 1000 would make the performances drop signifi cantly, which is probably because of the inadequate training input. Third, we use the class weight that can be set in the model building time. We set the class weight to the corresponded sample amounts in two classes. Table 3. reflects the results of the combination of the second and third methods. We can notice that the recall score increases among all the models and the F1 scorealso increases among the random forest, logistic regression, and XGB.

Table III Models Accuracy



Model Names	Precision	Recall	F1 Score	Weighted F1 Score
Decision Trees	0.92	0.86	0.89	0.96
Random Forest	0.99	0.84	0.90	0.98
Logistic Regression	0.98	0.86	0.91	0.98
XGB	0.97	0.73	0.80	0.97
Logistic Regression with tuning*	**0.99**	**0.88**	**0.92**	**0.99**

We may take a look at the study that has been done to determine whether or not a job ad is authentic.Numerous studies have shown that job scammers use bogus internet job postings to sell their services.Many respectable and well-known companies were found to have posted fake job postings or vacancies for their own gain.The EMSCAD dataset was used to evaluate a variety of classification approaches, including the naive bayes lassifier, random forest classifier, Zero R, One R, and more.With an accuracy rate of 89.5 percent, the Random Forest Classifier was the most successful on this dataset.In this dataset, logistic regression performed horribly, as they found.One R classifier performed well when the dataset was balanced and tested on.Alghamdi proposed a mechanism for identifying fraud vulnerability in an online recruitment system.They experimented with the EMSCAD dataset using a machine learning approach.This dataset has been used by researchers in the past to do work in three different stages: pre-processing of the data, feature selection, and detecting classification fraud.To preserve the basic structure of the text, they removed any extraneous information such as html tags and other forms of formatting.Using pre-processing of the data, they removed noise and html tags from the data while maintaining the overall text pattern they were looking for.Researchers have effectively and efficiently reduced the number of features by using the feature selection technique.An ensemble random forest predictor was used to identify fake job postings in the test data and an SVM for feature selection.Random Forest Classifier looked to be a tree-based classifier that acted as an ensemble classifier using the majority voting method.

This classifier achieved a classification accuracy of 97.4 percent.

It was hypothesised by Huynh that pre-trained text datasets might be used to train models such as Text CNN, LSTM-Bi-GRU-LSTM CNN, or BiGRU CNN.They concentrated on categorising a dataset of IT positions.Convolution, pooling, and fully connected layers were utilised to train a dataset of IT occupations.Data was fed into this model in the form of levels.The flattened weights were then supplied to the fully connected layer.Uses softmax function for categorization.They used an ensemble classifier (Bi-GRU CNN, Bi-GRULSTM CNN) with a majority voting strategy to increase classification accuracy.TextCNN had a classification accuracy of 66%, whereas Bi-GRU-LSTM CNN had a classification accuracy of 70%.An accuracy of 72.4% was achieved by the ensemble classifier in this classification task..The main goal is to let the model study the texts in the job posts, wewill merge the text features from the Table 1. into an mega text column andapply the preprocess pipeline to them. In order to optimize different models'performance, we use different preprocessing methods for different models.For the traditional machine learning models, we utilize the Spacy [5] tokenizerto create our token object, which is equal to linguistic annotations, lemmatizeeach token, remove spaces, convert into lowercase, and then remove the stopwords. We then create the collections of bag of words, which vectorizes the tokens and createsa vector for each of them. Advantages of utilizing this method is that differentvectors can have relationships to others, which implicitly formulate the latent corresponse between the words, i.e. the vectors of 'apple' and 'pear', which belong to the fruit, may have a smaller Euclidean Distance compared to the vector of'tiger'.For the Recurrent Neural Networks or the BERT model, we use the wordembedding instead. The advantage of this method is that most of the pretrainedword embeddings [6] are based on gigantic pretrained dataset, which will be accurate enough . And for other non-pretrained word embeddingsthe built-in functionalities would help the corresponded model better study therepresentation and converge fast.We then setup the pipeline with the help of Bag of Words and combine the abovepreprocess methods and the models below.The models should be generalized in both classes even though the fraudulenthas a extreme small sample. Besides the modification of the dataset columns,we also try the downsampling of the non-fradulence job posts as it is extremeunbalanced. We also apply the ROC curve optimization in order to alleviate the recall score, which corresponded the fraudulent prediction.

CHAPTER 3

TERMINOLOGIES USED

"Machine learning" and "Deep Learning" algorithms were examined as part of this research project.The performance of many models on these datasets was evaluated and the most frequently utilised information for testing and learning algorithms was analysed.. This short essay is meant to assist researchers who are currently working in this subject and desire to do research in this area. Further in-depth study and advances with a variety of different applications will be the focus of future research.

3.1 NATURAL LANGUAGE PROCESSING (NLP)

A computer program's ability to decipher recorded human speech, known as "natural speech," is known as "natural language processing" (NLP).It's a feature of the computer's artificial intelligence (AI).The roots of NLP may be traced back to linguistics, and the field has been around for more than 50 years.You'll find a wide variety of applications, including medical research, search engines, and business intelligence, if you take a look at the scenarios.The NLP is used in computers to do the job of analysing human language.Data from the actual world is processed using "Artificial Intelligence," which uses natural language processing (NLP).Understand, The linguistic form is either spoken or written in the language.Similarly to the many senses individuals have, computers have a variety of sensors like programmes to read and microphones to collect sound.People, on the other hand, are capable of analysing their own data in the same way that computers are able to do so.At some point during the processing, the input is turned into computer-readable code.. NLP is basically can be formed using the two main components i.e "Data PreProcessing" and the other one is "Algorithm Development".To put it another way, "data preparation" is the act of making text data as clean as possible so that computers can study it.Preprocessing is the process of transforming raw data into something that an algorithm can work with.Many options are available to do this, such as:Tokenization is the technique of making the substituting for a token for the delicate data which can be crucial for any entity i.e credit card number..Sensitive data must still be safely maintained in a centralised location for future reference and must be surrounded by robust safeguards. As the surrogate value and its mapping to an initial value are generated using algorithms and procedures, they both have a role in determining how secure a tokenization technique is. In many NLP applications, "stop word elimination" is a common preprocessing technique.As the name suggests, the goal is to simply delete all phrases that present in the corpus.Stop words, such as articles and pronouns, are common in English.Some NLP tasks, such knowledge discovery and segmentation, don't use these terms since they have no meaning, signalling that they aren't highly discriminative.Some NLP applications, on the other hand, will not be affected by the removal of stop words.In most cases, the "stop word" list for a given language contains words that occur often in corpora.The majority of stop-word lists are accessible on the InternetAffixes may be removed from words using the stemming technique, which then reveals the root form.It's like chopping down a tree from the top to the bottom.we may use the example as a consideration of the stem of words eating, eats, and eaten is eat.The "stemming" method is used by search engines to index words.As a consequence, a "search engine" may simply record the stems of a word rather than all permutations of a word.As a consequence, "Stemming" reduces the size of the index and improves search speed..The stemming process is similar to lemmatization.The result of lemmatization is called a 'lemma,' which is a "root word" instead of a root stem, which is the result of stemming.We will receive a legitimate term that signifies the same thing after lemmatization.The WordNetLemmatizer class in NLTK is a lightweight wrapper over the WordNet corpus.To find a lemma, this class employs the WordNet CorpusReader class's morphy() method. For Natural Language Processing, "part-of-speech" (POS) may be used to categorise words in the corpus based on the meaning of the term and its context in respect to a certain "part-of-speech."Because each word has its own lexical term under it, repeating these full terms while doing text analysis may quickly become laborious, particularly as the database grows in size.We use "tags" to denote the categories since the POS tags are used to characterise the lexical phrases in our text. Data is stored, sorted, and manipulated using human-created rules in a "Rule-based system.".In this manner, it resembles human intelligence.It is necessary for a rule-based system to have a data source and a set of rules for modifying that data to work.'If statements' are usually referred to as rules that follow the 'IF X occurs THEN do Y' pattern.

The stages could be described this way:

- The data or new business event comes first.
- Once the data has been entered into the system, it is analysed to ensure that it complies with its standards.
- Any automatic follow-up procedures are then taken.

3.2 K-NEAREST NEIGHBOR(KNN)

Algorithms like the "K-Nearest Neighbor" algorithm, which are used in classification and regression, are classified as "Supervised Learning" algorithms.. The K-NN approach states that every new specific case and prior examples are linked, and the classification of the more similar group is assigned. By comparing new input to previous data, the K-NN algorithm maintains track of everything. This means that utilizing the K-NN approach, new information may be swiftly sorted into the appropriate category.

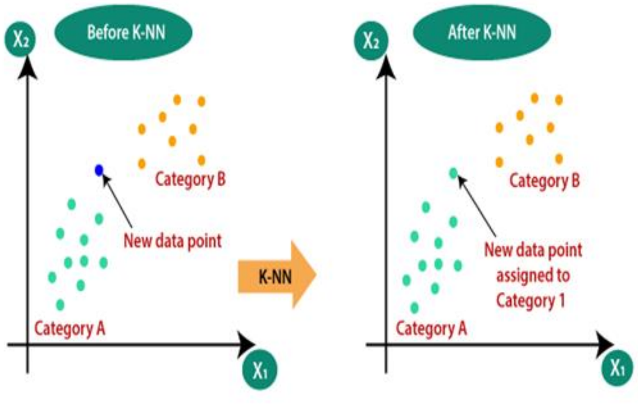


Figure 3.1 KNN Classification

3.3 RANDOM FOREST

Under the ensemble learning method, the "Random Forest" Algorithm is a network of huge trees work together as an ensemble. In the algorithm each network of tress construct a class prediction, in which the class having the majority votes decide the prediction of our model. Any of the unique essential a huge count a committee. The key is the poor correlation between models.

Benefits of applying this algorithm are

□ Accuracy is High

□ Missing Data is held effectively

□ Eradicate the over fitting issues

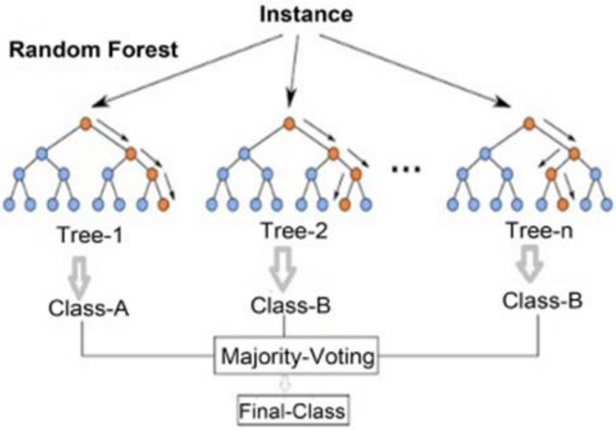


Figure 3.2   Random Forest classification Process

3.4 DECISION TREE

A decision is the method of classifying instances.Presume that everything of the input features have limited isolated areas and that this section only has one goal feature, "classification."A class is the name given to individual component of the categorization domain. Nodes labelled with the target feature's possible values, or that move to a secondary decision node on a different input characteristic, generate arcs.There should be a class labelled on each of the tree's leaves.

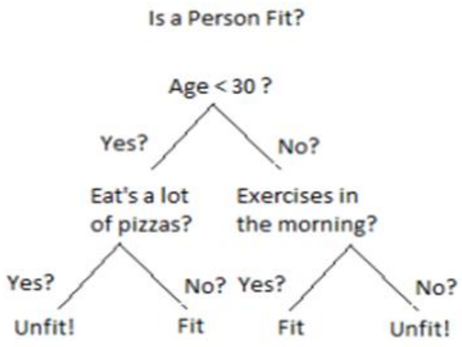


Figure 3.3 Decision Tree Example for Person is fit or Not

3.5 SUPPORT VECTOR MACHINE (SVM)

Under the supervised learning models in "Machine Learning" Algorithm, the SVM Algorithm is there refers to "Support Vector Machine".So that following data points may be put in the correct category, the SVM technique aims to find the ideal location or judgement boundary in n-dimensional space.The optimal selection boundary is denoted by a hyperplane.When deciding on the dangerous locations for the hyperplane, SVM is used.Choose the best decision boundary to help classify the data points in n-dimensional space from many lines/decision boundaries.

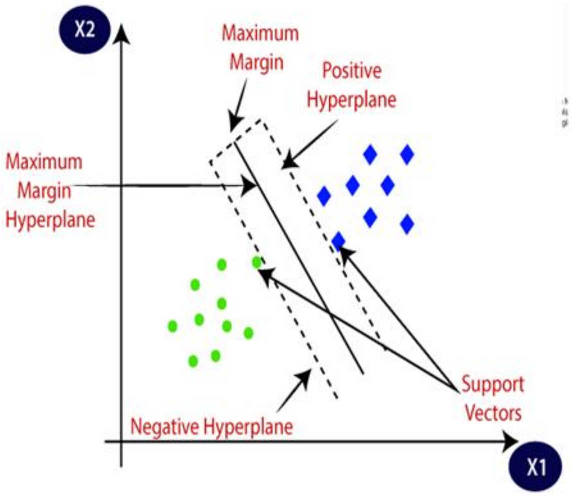


Figure 3.4 SVM Hyperplane and Support Vectors

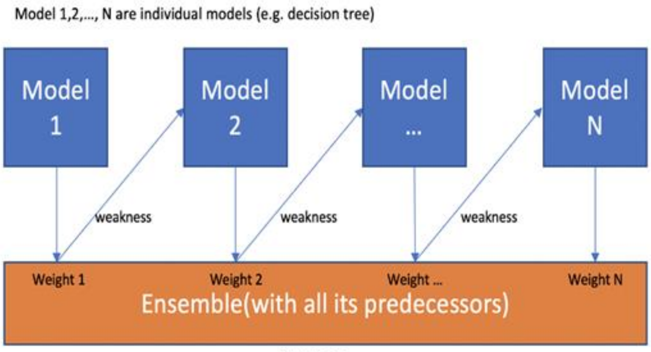
3.6 LOGISTIC REGRESSION

The method of modelling the likelihood of an unique outcome as a result of an input variable is identified as logistic regression.The highly common logistic regression prototypes have got a dual result, It might mean true or false. Logistic regression is a valuable analytical tool for determining if fresh samples practise in placing into a certain category in classification tasks. Because threat identification as well as the diiferent alarming issues of cyber security are classification challenges

$$p(x) = \frac{1}{1 + e^{-(x-\mu)/s}} \tag{3.1}$$

3.7 ADABOOST CLASSIFIER

Under the “statistical classification meta-algorithm",the "AdaBoost" Algorithm is there also known as Adaptive Boosting used in the Ensemble Learning Algorithm in Machine Learning Problems. Every time a mistake is made in the identification of an instance, the weights are redistributed.Adaptive Boosting is the name given to this kind of a training. Boost reduces bias and variation in supervised learning.Progressivism is the guiding principle behind it.Each following student, with the exception of the first, is a creation of previous mature pupils.Another way of putting it is that weak people turn become strong people.AdaBoost may be used in the same way as a boosting strategy with a few tweaks.


Model 1,2,..., N are individual models (e.g. decision tree)
Model 1
Model 2
Model ...
Model N
weakness
weakness
weakness
Weight 1
Weight 2
Weight ...
Weight N
Ensemble(with all its predecessors)

Source: Google

Figure 3.5 Ensemble of N Trees

The record that was incorrectly classified is used as input into the following model.This process is carried on indefinitely in order to meet the demand.You can see this in the figure, where the previous model's faults were used to build 'n' models.Once we've mastered the concept of boosting, understanding the AdaBoost algorithm should be a breeze.Let's take a closer look at AdaBoost.A 'n' trees will be added once the algorithm has been applied.Trees with a root node and several leaf nodes are generated.Although there are bigger and smaller specimens of each kind,, there is no set length.However, using AdaBoost, the method only creates a Stump node, which has two leaves

3.8 GRADIENT BOOSTING CLASSIFIER

Gradient boosting includes the three components:

1. An optimization function for a Loss Function.
2. To make predictions, a slow learner.
3. To reduce the loss function, to implement weak classifier.

3.8.1 LOSS FUNCTION

The problem at hand dictates the sort of loss function to be used.Many typical loss functions are provided, as well as the opportunity to design your own.A logarithmic loss is utilised in classification, whereas a squared error is used for regression.Any discrete loss function might be implemented in this architecture since it is so versatile.

3.8.2 WEAK LEARNER

Decision trees are used for the weak learners in gradient boosting.As a result of the use of regression trees with output, subsequent models are able to "correct" any remaining estimations.. Initially, incredibly one split, called decision stumps, were utilised, as has been the scenario with AdaBoost.It is usual to place restrictions on weak students.This is done to guarantee that the students stay weak but may still be built greedily.

3.8.3 ADDITIVE MODEL

When adding trees, approach is to reduce the loss.Gradient descent has traditionally been used to minimise a collection of regression equation coefficients or neural network weights.The weights are changed once the mistake or loss is calculated to minimise the error. To conduct the gradient descent technique after computing the loss, by doing this we achieved right configuring of the tree, changing its characteristics, as well as move towards the direction of reducing the residual loss.

3.9 CONVOLUTIONAL NEURAL NETWORKS (CNN)

Convolutional Neural networks, is by far the most promising ways for creating machine learning models.It excels at picture categorization and computer vision, for example.CNN is typically a neural network with a different convolutional layer than other neural networks.CNN examines every corner, vector, and dimension of the pixel matrix to achieve picture categorization.CNN is more resilient to data in matrix form when it performs with all of the characteristics of a matrix.Convolutional layers provide numerous characteristics, such as detecting edges, corners, and different textures, which makes them a unique tool for CNN modelling.That layer can identify all of the characteristics in the picture matrix as it glides over it.This means that each of the network's convolutional layers can recognise increasingly complicated signals features.The dimension of the convolutional layer must be increased as the feature grows.Text data could be conceived of as data sets, similar to data in a time series, or as a one-dimensional matrix.We must use a one-dimensional convolution layer.The model's concept is virtually identical, but the data format and dimension of the convolution layers have altered.There are both theword embedding layer as well as one-dimensional convolutional network are required to operate with TextCNN.

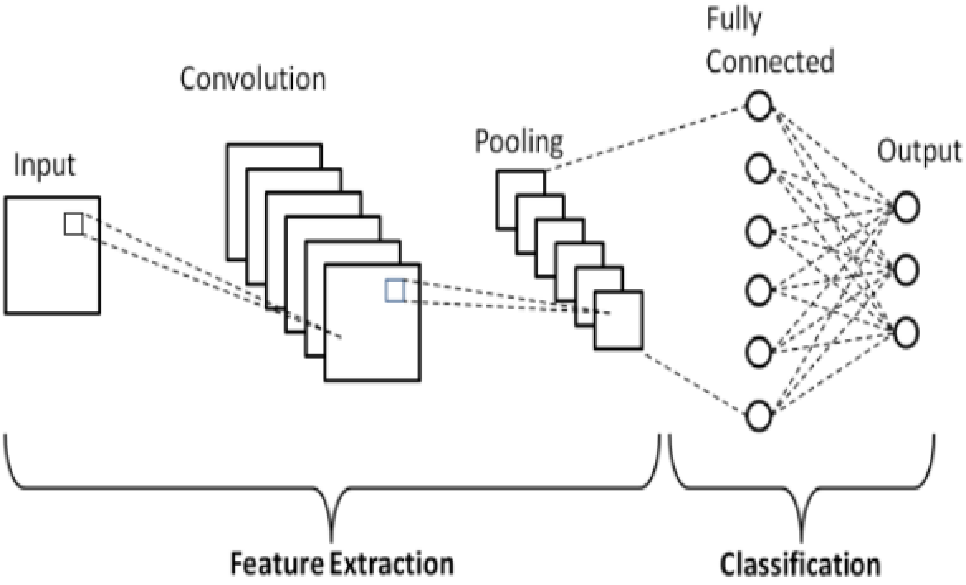


Figure 3.6 CNN Architecture

3.10 LONG SHORT TERM MEMORY (LSTM)

Data may be retained forever using LSTM, which is an extended RNN (sequential network).RNN's vanishing gradients issue may be solved with this technique.Perpetual recollection, RNN, is utilised.At the most basic level,where LSTM is analogous to an RNN cell.This diagram shows how the LSTM network works inside.The LSTM is split into 3 portions, each one serves a specific role, as shown in the diagram below. The previous timestamp's information should be remembered or destroyed in the first sectionIn the second phase, the cell tries to learn new things from the input it receives.

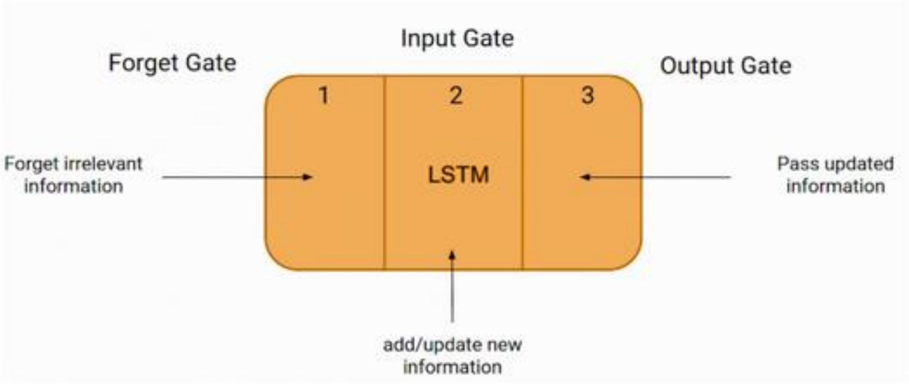


Figure 3.7 LSTM Architecture

Cells finally convert data from one time stamp to another in section three.Gates refer to the three parts of an LSTM cell:This circuit has three gates: a forget one, an input one, and an output one. The output gate comes last.It holds the hidden state as H(t-1) is the previous timestamp's hidden state and H(t) is current timestamp's hidden state, similar to a simple RNN.C(t-1) and C(t) represent the cell state of an LSTM, which may be used to denote the previous and current timestamps, respectively.Cell state is known as long-term memory, but the concealed state is called short-term memory in this circumstance.

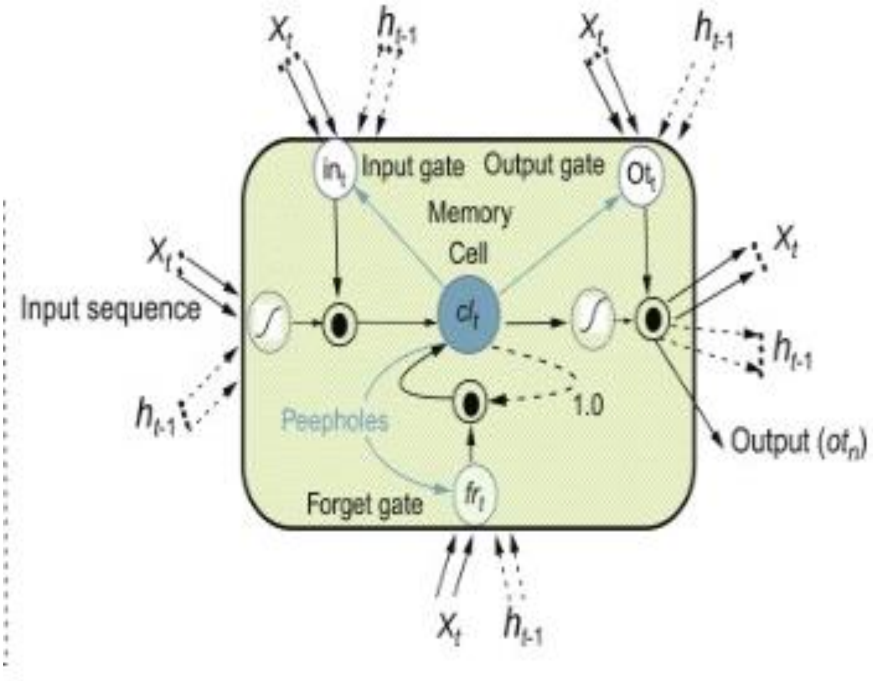


Figure 3.8 LSTM Memory Block

3.11 FAST TEXT

FastText is a library was developed first by Facebook Development Team to help people learn word representation and sentence categorization rapidly.

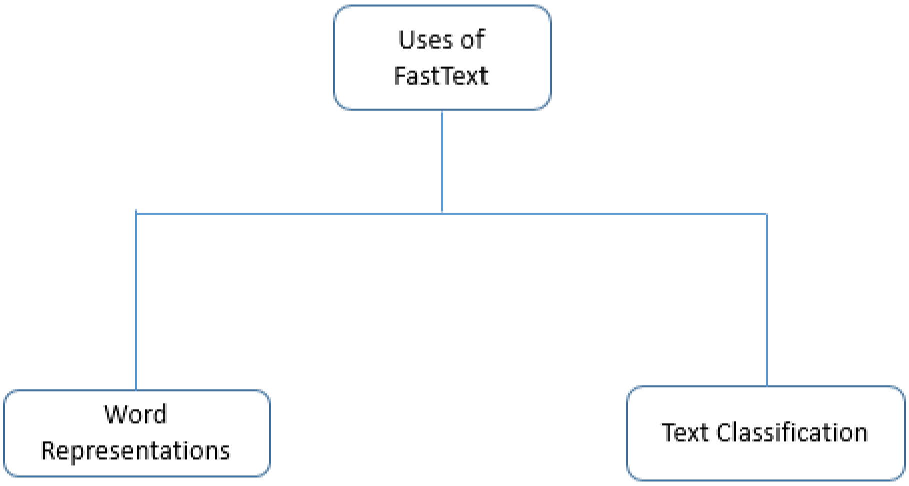
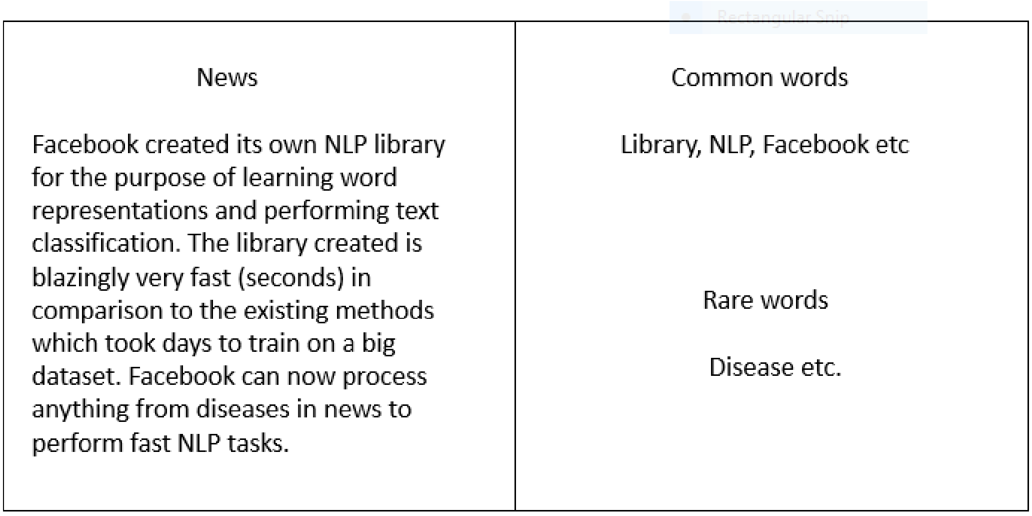


Figure 3.9 Use Cases of Fast Text

This library is a prospective replacement for the gensim package, which contains Word Vectors and other functions.If you're new to Word Vectors and word representations, start here.How does FastText differ from gensim Word Vectors?

FastText varies from word vectors, commonly known as word2vec, in that it assumes a word is made up of n-grams of characters.It identifies unusual word vectors.Medical terms like diseases may be unique for a news-trained model.OOV words may be split into character n-grams for embedding.Glove and word2vec don't vectorize non-dictionary words.



News	Common words
Facebook created its own NLP library for the purpose of learning word representations and performing text classification. The library created is blazingly very fast (seconds) in comparison to the existing methods which took days to train on a big dataset. Facebook can now process anything from diseases in news to perform fast NLP tasks.	Library, NLP, Facebook etc Rare words Disease etc.

Figure 3.10   Fast Text Example

CHAPTER 4

PROPOSED WORK

4.1 PROBLEM STATEMENT

It's difficult to build a machine learning model with a classification job that may provide several results.In order to improve prediction accuracy, classifier ensemble approaches combine many model outputs.Ensemble techniques range from simple to complex.However, the majority voting technique is used in our implementation.Text categorization issues responded well to this approach, despite its simplicity.

4.2 PROPOSED METHOD

In this research, we created four Deep Neural Network Algorithms models for Online Fake Scam Job classification using TextCNN, two combination models (Bi-GRU-CNN and Bi-GRU-LSTMCNN), and our recommended ensemble model.These models were also used to implement the two pre-trained word embeddings.

4.2.1 TEXTCNN MODEL

TextCNN, a programme for categorising job descriptions built by, was the first source of inspiration for us.When it comes to NLP use cases like emotion identification, sentiment analysis, and question categorization, TextCNN consistently outperforms other deep learning algorithms.For text categorization, the convolution layer, pooling layer, and fully connected layer (FCL) are the three most important components.We employed 512 filters in the Kernel - the Convolution layer - to collect elevated features and generate convolved feature maps, totaling three kinds of filters of varying widths.Through dimensionality reduction, they are then transferred to the Pooling layer, which decreases the size of the convolved feature and the amount of computational power required to analyse the data.In the Convolutional Neural Network's eighth layer, the convolutional layer and the pooling layer are combined. '

A standard neural network will then be used to classify the final output using the softmax method, which is then smoothed.

4.2.2 BI-GRU-LSTM-CNN MODEL

When Huynh et al. proposed the use of this model in the VLSP Share Task 2019, it finished in sixth place on the public test scoreboard.The diagram depicting our proposed network layout is displayed here.At its core, CNN-1D represents the model's architecture.Rather of using two parallel Bi-GRU networks, the Bi-LSTMCNN model makes use of both Bi-GRU and Bi-LSTM networks in parallel, as does the Bi-GRU-LSTMCNN model.Accordingly, Instead of going through the complete model, we'll focus on the Bi-LSTM part of our proposed ensemble for predicting phoney employment scams.In both directions, the LSTM Using two Bidirectional Long Short Word Memory (Bi-LSTM) blocks, the model sends input sequences to the LSTM in two distinct directions.With an input gate, output gate, and forget gate, the LSTM is a recursive neural network variation.LSTMs with 112 units each were used in parallel in our experiment.We used sigmoid and tanh for recurrent activation and hidden units, respectively.

4.2.3 A SIMPLE ENSEMBLE MODEL USING MAJORITY VOTING

Within the scope of this investigation, we make use of the Majority Voting methodology in order to enhance the prediction capabilities of our classification model.In order to implement this method, the results of n different models will be used..The problem's ultimate categorization result is a voting-based mixture of the following results.For each expected point, cast a vote.The most popular tag will be used as the output tag. When we look at the data, we use the three most accurate m classification models to forecast each job description's y label based on majority (plurality) vote of each Ci classification model.

ˆy = mode {C1 (x), C2 (x), . . . , Cm (x) }

For example, we have a sample result as follows:

- Model 1 -> Label 1
- Model 2 -> Label 0

- Model 3 -> Label 1

ˆy = mode {1, 0, 1 } = 1

Class 0 will be determined by a simple majority vote on the sample.Because each model for each label may be distinct, if we can't find the label result after voting, we'll use the last label as the model's label.Whichever of the three models receives the most support garners the best classification results.

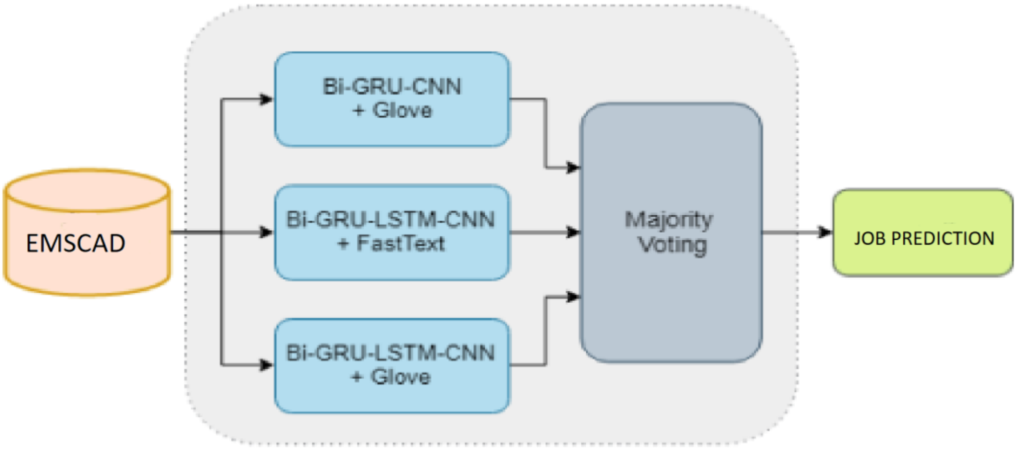


Figure 4.1 Architecture of Voting Model

When evaluating the effectiveness of a model's performance, it is necessary to use certain metrics in order to validate the evaluation.The following metrics are taken into consideration in order to deliver the most appropriate issue-solving method that is also capable of tackling the problem in an effective manner.However, you must not take into consideration circumstances that have been foreseen wrongly if you want the model to be accurate.When a post that has been tampered with in any way is seen as genuine, a significant problem occurs.As a direct consequence of this, it is necessary to handle compensating scenarios for inaccurate classification.When doing this analysis, you need to take into account both the accuracy and the recall. In this study, Natural Language Processing (NLP) and Machine Learning (ML) techniques for identifying Online Fake Job Predictions were investigated.The effectiveness of various models while they were working on these datasets was analysed, and research was conducted on the datasets that are most often used for testing and training models.The purpose of this short essay is to aid both active researchers and those interested in undertaking research in this field who are interested in performing research.The emphasis of future research will be on doing

CHAPTER 5

EXPERIMENTS AND RESULTS

5.1 PRE-PROCESSING

In this work, we use the following pre-processing techniques to ensure that the model's input is accurate:

- The job descriptions are being converted into lowercase strings.
- Delete special characters like #, & *, $, and so forth.
- Job descriptions are divided into a series of terms.
- In the Job descriptions, removing the stop word.
- Using pre-trained word embedding sets to represent words as vectors.

5.2 EXPERIMENTAL SETTINGS

At random, we split the dataset into three sets: 10% were used for testing, and 90% were used for training.In the training dataset, we obtain 20% of the validation dataset.In order to assess our model's accuracy, precision, recall, and F1-score, we use four assessment techniques.

5.3 EXPERIMENTAL RESULTS

We've found that the Bi-GRU-CNN model with the word embedding Glove is much more durable than other models (such as the Bi-GRULSTM-CNN and the TextCNN model).The conclusion of this model achieves an accuracy of 72.40 percent, and it also produces exceptional results in other metrics such as 72.38 percent for F1-score, 72.46 percent for precision, and 72.30 percent for recall respectively.When used in conjunction with the CNN model, it has been shown that the Bi-GRU architecture is more successful than the Bi-GRU-LSTM design.Even though it was not the best model in the test set, the Bi-GRULSTM-CNN model obtained great results when combined with pretrained embeddings such as Glove and FastText.It was a significant improvement over Papachristou's performance, which had a maximum accuracy of 66 percent when using the TextCNN model [28], in contrast to the model Bi-GRU-LSTMCNN with the word embedding FastText, which had an accuracy of 71.20 percent. It was a significant improvement over Papachristou's performance.In the end, we tried out the proposed ensemble model that we had developed.The ensemble approach produced the best results, ranking much better than other models in terms of accuracy (72.70 percent), F1-score (72.71 percent), precision (72.83 percent), and recall (72.59 percent).

Table IV Proposed Models Metrics

Models	**Accuracy(%)**	**Recall (%)**	**F1-Score(%)**	**Precision(%)**
TextCNN + FastText	69	68.86	69.21	69.57
TextCNN + Glove	65.30	65.61	65.42	65.42
Bi-GRU-CNN + FastText	70.20	69.94	70.31	70.69
Bi-GRU-CNN + Glove	72.40	72.30	72.38	72.46
Bi-GRU-LSTM-CNN + FastText	71.20	71.07	71.48	71.89
Bi-GRU-LSTM-CNN + Glove	70.30	70.52	70.71	70.91
Ensemble	72.70	72.59	72.71	72.83

5.4 APPLICATION

We may use our best-performing algorithm to construct a few apps based on our findings, such as Fake job Scam prediction apps based on data supplied by online job sites, and Job's screening applications for Online job Portals to assist them in filtering the Fake Job's and helping the job aspirants to not fall in the trap of Job scams. There is a pressing need for action to combat fake job advertisements in the real world.It is the goal of this project to find a possible solution to this issue.Data is pre-processed to give the best results, and important numerical fields are also selected.The best potential outcomes are achieved by combining the output of many models.So that a machine learning model is less likely to favour the dominant group, this is a necessary step.We learned a lot about employment fraud epitomised by certain areas in this study.There are 15 times as many bogus jobs as there are actual ones in Bakersfield, California.This kind of location need further monitoring.Furthermore, the majority of entry-level positions seem to be bogus.Those with a bachelor's or high school education searching for full-time work appear to be the primary targets of fraudsters.Text data preparation proved to be the most difficult.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

In order to address the problem of predicting fake job scams, we used not only the TextCNN model but also more complex models such as Bi-GRU-LSTM-CNN and BiGRU-CNN with a variety of word embeddings.We presented a straightforward and efficient ensemble model that is based on the experimental results of deep neural networks and that leverages the strengths of each model. After doing our research, we discovered that our technique is the most effective for predicting fake jobs.Using our recommended ensemble, we were able to attain the greatest performance possible on the EMSCAD dataset, scoring 72.71 percent on the F1-score scale.In the future, one of our goals is to improve not just the quantity but also the quality of the dataset.The dataset in particular is relatively limited, as there are only 17,880 job descriptions that have been annotated.On this corpus, we also want to experiment with other classical classifiers that have various features and deep learning models that have distinct word representations. Alternatively, we want to combine the two methods.On this dataset, we will compare traditional machine learning with deep learning in order to get a comprehensive understanding of what this study is trying to accomplish..We will also investigate the LSTM versions.In the future, if any company wishes to apply these models to real-world scenarios or other datasets related to job fraud prediction, we recommend using hyperparameters tailored LSTM,TextCNN models with transfer learning. The dataset that is being used for this project has a lot of imbalances in it.The majority of jobs are legitimate, and just a small percentage are scams.Because of this, genuine employment opportunities can now be located with relative ease.Synthetic minority class samples may be created by the use of certain methodologies, such as SMOTE.A more well-rounded dataset should be able to provide more accurate findings.

REFERENCES

[1] Devsmit Ranparia, Shaily Kumari and Dr. Ashish Sahani "Fake Job Prediction using Sequential Network"- 2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS).

[2] Sultana Umme Habiba, Md. Khairul Islam and Farzana Tasnim "A Comparative Study on Fake Job Post Prediction Using Different Data mining Techniques" -2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST).

[3] Sokratis Vidros, Constantinos Kolia,Georgios Kambourakis and Leman Akoglu "Automatic Detection of Online Recruitment Frauds: Characteristics, Methods, and a Public Dataset"- 2017 by the authors. Licensee MDPI, Basel, Switzerland.

[4] Shawni Dutta and Dr. Samir Kumar Bandyopadhyay "Fake Job Recruitment Detection Using Machine Learning Approach"- International Journal of Engineering Trends and Technology (IJETT) – Volume 68 Issue 4- April 2020.

[5] Bandar Alghamdi, Fahad Alharby " An Intelligent Model for Online Recruitment Fraud Detection "-Journal of Information Security, 2019, 10, 155-176.

[6] G. Kambourakis, "Employment Scam Aegean Dataset http://emscad.samos.aegean.gr" Unpublished, 2017, doi: 10.13140/RG.2.2.12872.72962.

[7] P. Simard, D. Steinkraus, and J. Platt, "Best practices for convolutional neural networks applied to visual document analysis," Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings

[8] J. Pennington, R. Socher, and C. Manning, "Glove: Global Vectors for Word Representation," Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014.

[9] S. Bansal, "[Real or Fake] Fake JobPosting Prediction," Kaggle, 29- Feb-2020. [Online]. Available: https://www.kaggle.com/shivamb/real- or-fake-fake-jobposting-prediction. [Accessed: 6-March-2020.

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

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

[12] Novaković, J.D., Veljovic, A., Ilić, S.S., Papic, Z. and Tomovic, M. (2017) Evaluation of Classification Models in Machine Learning. Theory and Applications of Mathematics & Computer Science, 7, 39-46.

[13] Morgan, S. and Menlo Park, C. (2017) Cybercrime Report from the Editors at Cybersecurity Ventures. Herjavec Group, Toronto.

[14] Gov, K. (2018) Vision of 2030. http://vision2030.gov.sa/en.

[15] ACRON (2018) Australian Cybercrime Online Reporting Network (ACORN). http://www.acorn.gov.au/learn-about-cybercrime.

[16] Armstrong, A. (2006) Handbook of Human Resource Management Practice. 10th Edition, Kogan Page Limited, London.

[17] Hada, B. and Gairola, S. (2015) Opportunities and Challenges of E-Recruitment. Journal of Management Engineering and Information Technology, 2, 1-4.

[18] Prasad, L. and Kapoor, P. (2016) Topic: E-Recruitment Strategies. International Journal of Business Quantitative Economic and Applied Management Research, 2, 80-95.

[19] Panov, P., Soldatova, L. and Džeroski, S. (2013) OntoDM-KDD: Ontology for Representing the Knowledge Discovery Process. 16th International Conference on Discovery Science, Singapore, 6-9 October 2013, 126-140. https://doi.org/10.1007/978-3-642-40897-7_9.

[20] Cios, K.J., Pedrycz, W., Swiniarski, R.W. and Kurgan, L.A. (2007) Data Mining: A Knowledge Discovery Approach. Springer, New York.

[21] Hussain, S. (2017) Survey on Current Trends and Techniques of Data Mining Research. London Journal of Research in Computer Science and Technology.

[22] Sinoara, R., Antunes, J. and Rezende, S. (2017) Text Mining and Semantics: A Systematic Mapping Study. Journal of the Brazilian Computer Society, 23, 9. https://doi.org/10.1186/s13173-017-0058-7.

[23] Diwathe, D. and Dongare, S. (2017) Classification Model Using Optimization Technique: A Review. International Journal of Computer Science and Network, 6, 42-48.

[24] B. Alghamdi and F. Alharby, ―An Intelligent Model for Online Recruitment Fraud Detection," J. Inf. Secur., vol. 10, no. 03, pp. 155–176, 2019, doi: 10.4236/jis.2019.103009.

[25] I. Rish, ―An Empirical Study of the Naïve Bayes Classifier An empirical study of the naive Bayes classifier,‖ no. January 2001, pp. 41–46, 2014.

[26] D. E. Walters, ―Bayes's Theorem and the Analysis of Binomial Random Variables,‖ Biometrical J., vol. 30, no. 7, pp. 817–825, 1988, doi: 10.1002/bimj.4710300710.

[27] F. Murtagh, ―Multilayer perceptrons for classification and regression,‖ Neurocomputing, vol. 2, no. 5–6, pp. 183–197, 1991, doi: 10.1016/0925-2312(91)90023-5.

[28] P. Cunningham and S. J. Delany, ―K -Nearest Neighbour Classifiers,‖ Mult. Classif. Syst., no. May, pp. 1–17, 2007, doi: 10.1016/S0031-3203(00)00099-6.

[29] H. Sharma and S. Kumar, ―A Survey on Decision Tree Algorithms of Classification in Data Mining,‖ Int. J. Sci. Res., vol. 5, no. 4, pp. 2094–2097, 2016, doi: 10.21275/v5i4.nov162954.

[30] E. G. Dada, J. S. Bassi, H. Chiroma, S. M. Abdulhamid, A. O. Adetunmbi, and O. E. Ajibuwa, "Machine learning for email spam filtering: review, approaches and open research problems,‖ Heliyon, vol. 5, no. 6, 2019, doi: 10.1016/j.heliyon.2019.e01802.

[31] L. Breiman, ―ST4_Method_Random_Forest,‖ Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1017/CBO9781107415324.004.

[32] B. Biggio, I. Corona, G. Fumera, G. Giacinto, and F. Roli, ―Bagging classifiers for fighting poisoning attacks in adversarial classification tasks," Lect.

Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 6713 LNCS, pp. 350–359, 2011, doi: 10.1007/978-3-642-21557-5_37.

[33] A. Natekin and A. Knoll, ―Gradient boosting machines, a tutorial,‖ Front. Neurorobot., vol. 7, no. DEC, 2013, doi: 10.3389/fnbot.2013.00021

[34] N. Hussain, H. T. Mirza, G. Rasool, I. Hussain, and M. Kaleem, ―Spam review detection techniques: A systematic literature review,‖ Appl. Sci., vol. 9, no. 5, pp. 1–26, 2019, doi: 10.3390/app9050987.

[35] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, ―Fake News Detection on Social Media,‖ ACM SIGKDD Explor. Newsl., vol. 19, no. 1, pp. 22–36, 2017, doi: 10.1145/3137597.3137600.

[36] Shivam Bansal (2020, February). [Real or Fake] Fake JobPosting Prediction,Version 1.Retrieved March 29,2020 from https://www.kaggle.com/shivamb/real-or-fake-fake-jobposting-prediction.

[37] S. M. Vieira, U. Kaymak, and J. M. C. Sousa, ―Cohen's kappa coefficient as a performance measure for feature selection," 2010 IEEE World Congr. Comput. Intell. WCCI 2010, no. May 2016, 2010, doi: 10.1109/FUZZY.2010.5584447.

[38] Blanzieri, E.; Bryl, A. A survey of learning-based techniques of email spam filtering. Artif. Intell. Rev. 2008, 29, 63–92.

[39] Guzella, T.S.; Caminhas, W.M. A review of machine learning approaches to spam filtering. Expert Syst. Appl. 2009, 36, 10206–10222.

[40] Saadat, N. Survey on spam filtering techniques. Commun. Netw. 2011, 3, 153–160.

[41] Abu-Nimeh, S.; Nappa, D.; Wang, X.; Nair, S. A comparison of machine learning techniques for phishing detection. In Proceedings of the Anti-Phishing Working Groups 2nd Annual eCrime Researchers Summit, Pittsburgh, PA, USA, 4–5 October 2007; ACM: New York, NY, USA, 2007; pp. 60–69.

[42] Potthast, M.; Stein, B.; Gerling, R. Automatic vandalism detection in Wikipedia. In Advances in Information Retrieval; Springer: Berlin/Heidelberg, Germany, 2008; pp. 663–668.

[43] Potthast, M. Crowdsourcing a wikipedia vandalism corpus. In Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Geneva, Switzerland, 19–23 July 2010; ACM: New York, NY, USA, 2010; pp. 789–790.

[44] Chen, Y.; Zhou, Y.; Zhu, S.; Xu, H. Detecting offensive language in social media to protect adolescent online safety. In Proceedings of the 2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing, Amsterdam, The Netherlands, 3–5 September 2012.

[45] Potha, N.; Maragoudakis, M. Cyberbullying Detection using Time Series Modeling. In Proceedings of the 2014 IEEE International Conference on Data Mining Workshop (ICDMW), Dallas, TX, USA, 7–10 December 2013; pp. 373–382.

[46] Heydari, A.; ali Tavakoli, M.; Salim, N.; Heydari, Z. Detection of review spam: A survey. Expert Syst. Appl. 2015, 42, 3634–3642.

[47] Laboratory of Information and Communication Systems, University of the Aegean, Samos, Greece. EMSCAD Employment Scam Aegean Dataset, 2016. Available online: http://icsdweb.aegean.gr/emscad (accessed on 22 February 2017).

[48] Vidros, S.; Kolias, C.; Kambourakis, G. Online recruitment services: Another playground for fraudsters. Comput. Fraud Secur. 2016, 2016, 8–13.

[49] Sculley, D.; Wachman, G.M. Relaxed online SVMs for spam filtering. In Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, 23–27 July 2007; ACM: New York, NY, USA, 2007; pp. 415–422.

[50] Hershkop, S. Behavior-Based Email Analysis with Application to Spam Detection. Ph.D. Thesis, Columbia University, New York, NY, USA, 2006.

LIST OF PUBLICATIONS

[1] Rajesh Kumar Yadav, Pankaj Panwar, "A SURVEY ON ONLINE JOB FRAUD DETECTION". Accepted at the **4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N–22).**

Abstract- — In light of the growing world breaches of data that occur on a daily basis. With each passing day, who fall victim to a grows exponentially larger and larger. The bulk of jobseekers are recruited by corporations and fraudsters use the approaches, with the majority of them coming from digital job-posting in Indeed website. We want to apply Machine Learning in the future to reduce the incidence of similar frauds. Candidates would be able to maintain their vigilance and make smart decisions, when necessary, hence lowering the amount of such frauds that take place in the first place. Natural language processing will be sometimes used investigate the attitudes and patterns in the job advertisement, and to do so (NLP). In this work, we will examine different.We will investigate several techniques to detecting bogus jobs in our research.Finally, depending on their accuracy, I will analyse and contrast the various ML and deep learning models.