Comparisons of machine learning techniques for detecting fraudulent criminal identities

Hassan Kazemian[a] **(**h.kazemian@londonmet.ac.uk**), Subeksha Shrestha**[b] **(**sus0641@my.londonmet.ac.uk**)**

Intelligent Systems Research Centre, School of Computing and Digital Medial, London Metropolitan University, United Kingdom

[a,b] London Metropolitan University, 166-220 Holloway Rd, London, United Kingdom, N7 8DB

Corresponding Author Email address: h.kazemian@londonmet.ac.uk
First author: h.kazemian@londonmet.ac.uk
Second author: sus0641@my.londonmet.ac.uk

Comparisons of machine learning techniques for detecting fraudulent criminal identities

Hassan Kazemian[a], **Subeksha Shrestha**[b]

Intelligent Systems Research Centre, School of Computing and Digital Medial, London Metropolitan University, United Kingdom

[a,b] London Metropolitan University, 166-220 Holloway Rd, London, United Kingdom, N7 8DB

Abstract

This paper focuses on applications of various machine learning techniques on an anonymized policing dataset used in EU SPIRIT Horizon 2020 project to identify fraudulent identities and help Law Enforcement Agencies (LEAs) in their investigation in finding potential criminals and identity resolution. Lack of qualitative data and appropriate methodology to carry out research on criminal fraudulent identities is a common reason for fewer research in this area. Additionally, it is a very sensitive data to work with and minor inaccuracy in prediction of result causes massive impact in the society as genuine people could be questioned whereas criminals could be sent free. Both of these issues are addressed in this paper by application of 39 million records from policing dataset and working towards higher accuracy while building the model. Various machine learning approaches are applied to train the dataset to make predictions and the research focus on being able to predict the 5 suspected fraudulent identities out of 39 million records in the policing dataset. One of the applied machine learning techniques include TensorFlow along with Keras model which has seldomly been applied by researchers in detection of criminal data. To compare the results and test accuracy of TensorFlow model, other machine learning techniques such as Support Vector Machine, Naïve Bayes and K-nearest Neighbours are also applied to have a comparative study on the obtained outcomes from each model. The goal of this research is to find fraudulent IDs amongst all the anonymized IDs in the criminal dataset using TensorFlow and three other machine learning models and select the most optimal model out of them. Since the model is comparing two names so string-matching techniques such as Levenshtein edit distance, Hamming Distance, Jaro-Winkler and Soundex were applied to select an effective approach first before building the model and analysing the results. TensorFlow model demonstrated highest accuracy with relatively least execution time and the only model to successfully predict all the 5 suspects from the policing dataset.

Keywords: Identity Resolution, Policing Dataset, TensorFlow, Support Vector Machine, K-Nearest Neighbour, Naive Bayes

Highlights:

- Machine learning approaches to resolve identity issues.
- Implementing TensorFlow to investigate policing dataset.
- Predictive models using ML techniques to identify fraudulent or genuine IDs.

Corresponding Author Email address: h.kazemian@londonmet.ac.uk
First author: h.kazemian@londonmet.ac.uk
Second author: sus0641@my.londonmet.ac.uk

1. Introduction

This section outlines the reasons behind the research and brief discussion on existing Identity Resolution techniques applied to other dataset rather than just to crime dataset.

The radical shift from registering information on papers to electronic form in the past decades has rose abundance of errors such as misspelled data, variation in naming order, case sensitive data, considering abbreviations as different terms and more. These occur due to inadequate data validation, duplication of data and/or flaws in data for criminal record's entry, and unintentional manipulation of data, which misrepresents the data and creates ambiguous identities (Xu, Wang, & Li, 2007). This project will incorporate various machine learning algorithms to detect fraudulent criminal data using a methodology combining social features, personal details and criminal history. The identity of a person is a peculiar aspect that differentiates one person from another but intentional manipulation of data to create multiple or imprecise identities is a very likely approach adopted by many criminals (School, 2021). This ultimately leads to increase in ambiguity of records in a real-time scenario which can lead to extremely critical conditions where criminals might be refrained from being convicted (Zhang, Ma, Yuan, & Fang, 2022).

The issues relating to identity resolution henceforth have risen in today's world of Online Social Networking (OSN) where personal details can be fabricated or manipulated easily for personal comfort or benefits (Yadav, Adwitiya, & Kumar, 2019). Accuracy of data is difficult to maintain, particularly in a scenario where the same criminal might attempt to portray a different identity to a police officer upon arrest and/or interrogation. By providing false details, a criminal can hide their identities and end up misleading the existing police records with new identity of themself and ultimately have multiple identities for the same person. This issue broadens when the data of criminals are used in investigation or are required for reference to the past records of any current suspect to look for traces of their involvement in any previously committed crimes. Handling criminal record is also typically difficult because their attributes such as name, date of birth, gender or phone number are not sufficient enough by itself for matching records to uniquely identify a criminal and find their match out from millions of rows (Li & Wang, 2015).

Information on the distance of criminals' home location and the crime site, role in the crime and even the offence committed are essential to investigate in a detailed manner and compare any two records to ensure whether or not they are a match. The policing dataset for this project has records where criminals have intentionally manipulated their details to disguise themselves and hide their original identity, so the research is to investigate on comparing every two strings that have higher similarity scores of names and then apply the ML algorithms to identify the similar false identities. There are many other significant researches carried out on customer's behaviour, credit card fraud, trace traveller's history, patient's records, for anti-money laundering, collecting KYC (Know Your Customer) details for secured transactions at banks and finances or details of politically exposed people (Cui, Wang, Xue, & Wang, 2021). A study conducted by Bartunov et al., authors investigate on different online social networks such as Twitter and Facebook to resolve issue of having multiple profiles of a single person (Bartunov, Korshunov, Park, Ryu, & Lee, 2012). Ahmed et all studied on merging different crime data from three sources and applied XGBoost classifier to study relation of human trafficking and drug related crimes (Ahmed, Gentili, Sierra-Sosa, & Elmaghraby, 2022). Another research conducted on probabilistic Naive Bayes approach to analyse and identify deception detection also includes study on identity resolution but the research is only limited to analyse the findings to improve staff's job performance (Chen, et al., 2006). In the work of Phua et al., a multilayer crime detection system is developed that studies social relationships of the suspects and also amplifies the duplicate records to increase the suspicion score (Phua, Smith-Miles, Lee, & Gayler, 2010). Kou et al. study has a very interesting approach feature selection and optimization of classification performance in two stages for models to predict

bankruptcy (Kou, et al., 2020). Although the research was carried out by comparing several models similar to our research but due to lack of quality data an in-depth relationship between variables and bankruptcy could not be obtained.

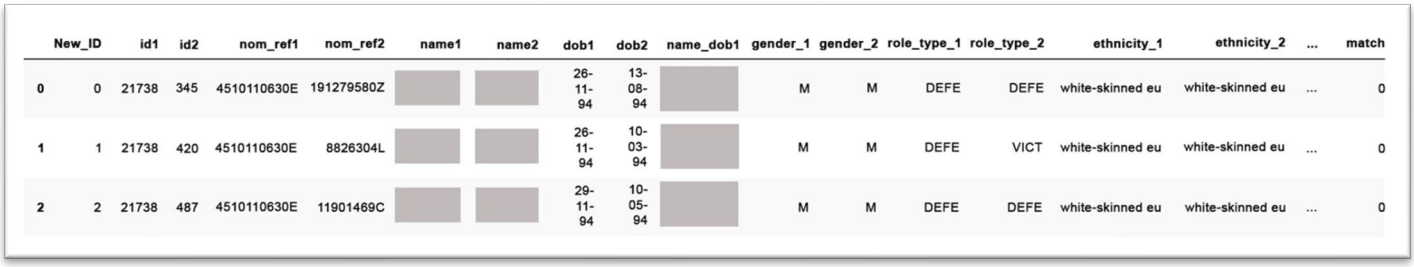
However, there is comparatively less research implementing identity resolution in criminal records. Integration of machine learning algorithm is implemented to not only enhance prediction but to ease the task of obtaining result without human intervention after training a model efficiently (School, 2017). This is where identity resolution plays an integral role to significantly train a model in the beginning and then work on test data to look for similar data for possible identity matches and distinguish between obvious and non-obvious relationships in a selected dataset (Furtado, 2009). Implementation of identity resolution avails the future of decision-making process and optimizes tedious tasks of skimming each row to find out only the fraudulent identities in a huge dataset (IBM, 2006). On a precise scale, identity resolution is a mechanism which is designed to detect similar data that differs in a spelling, variation in naming, abbreviations or nicknames and more.

2. Policing Dataset Analysis

This section includes introduction to the policing dataset, compare and contrast with existing research and details on ML techniques and various libraries, APIs and framework that are used to develop the predictive model.

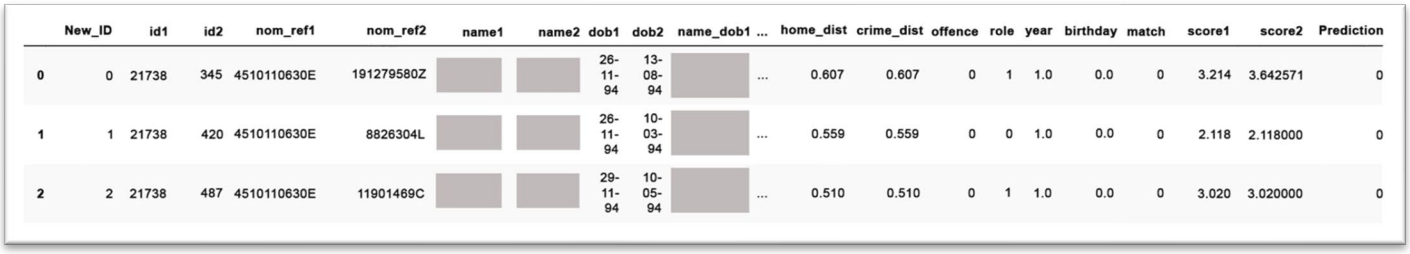
In a dataset related to criminal records manipulating details is generally tempting for criminals as from their end it is a cost-free and easy task to manipulate their personal details and escape out proving to be innocent (Ahmed S. R., 2020). A generic entity resolution-based framework to classify matched and unmatched records based on their features and semantic relations with other entities proposed by (Moir & Dean, 2015). This research was carried out on publicly available dataset, but in the policing dataset we have anonymised genuine dataset from the UK crime records to work for our research. A buffer optimizer-based algorithm alongside with application of K-anonymity based mobile application, reduces the linkage between two dataset and make it difficult for attackers to identity their target (Sakpere & Kayem2, 2018). An interesting aspect of this study is the K-anonymity that ensures the privacy of a person and making it difficult to identity any personal details which ultimately reduces the chance of attack. But, for this research not only machine learning techniques that are commonly applied in practice such as SVM, Naïve bayes and K-nearest neighbours are implemented but TensorFlow and Keras model is acquainted to predict fraudulent identities for criminal dataset. TensorFlow and Keras is very evident in being applied on customer behaviour or credit card fraudulent cases but for criminal investigation or criminal fraudulent identification it is quite novel research. While working on criminal dataset it is not only critical to work with such dataset carefully while building a model but they are prone to many errors which can are absolutely non-negligible as questioning or punishing innocent person based on the results obtained would be a petrifying. So, precision of a model is the key, and it should be utterly high and the most crucial aspect of research with criminal records. This research henceforth presents a unique methodology of matching identity based on various key factors such as grouping many attributed together such as nearby police beat to the location where crime was committed, ethnic group clustering of criminals, their gender, role in a particular crime along with personal details such as their name, date of birth, address, age and more. To pre-process the data numerous phases where completed which includes removal of some outliers and inconsistency in the data from various columns such as the ethnicity, gender, date of birth, etc. Also, to have uniformity across the entire dataset many attributes with similar features were grouped together to reduce huge number of divisions across the same category. For instance, instead of having over 100 different categories for crimes committed, only 16 main crimes were assigned where similar crimes were assigned to a particular group rather than having different types for each. To have a clear of picture of this scenario, crimes such as robbery, theft, burglary, etc were categorised into a single category as ‘theft’. This procedure consequently reduced computation time as well as the analysis of the results were also considerably swift. These 16

main attributes were decided by the research team alongside of the police officers working on the project. Similar approaches have been applied to other columns, such as for the ethnicity and the role in crime. After all these procedures to clean up the data, the final results were working with 39 million records and over 34 columns. For the train and test data split, K-fold validation was utilised to avoid any data leakage between the train and test dataset. This was done to obtain a higher precision and recall value so that there is minimal or absence of incorrect predictions from the model. As a result, approximately 31.2 million records were set as training data and 7.8 million records were test data. The convention is to follow the 80-20 rule to split the data in different subsets or folds where various values of K were tested on the dataset. The best fit was when K was assigned as 10 while performing the K-fold validation. To highlight the features of the policing dataset, the first few rows have been shown in Figures 1 and 2.



	New_ID	id1	id2	nom_ref1	nom_ref2	name1	name2	dob1	dob2	name_dob1	gender_1	gender_2	role_type_1	role_type_2	ethnicity_1	ethnicity_2	...	match
0	0	21738	345	4510110630E	191279580Z			26-11-94	13-08-94		M	M	DEFE	DEFE	white-skinned eu	white-skinned eu	...	0
1	1	21738	420	4510110630E	8826304L			26-11-94	10-03-94		M	M	DEFE	VICT	white-skinned eu	white-skinned eu	...	0
2	2	21738	487	4510110630E	11901469C			29-11-94	10-05-94		M	M	DEFE	DEFE	white-skinned eu	white-skinned eu	...	0

Figure 1: Raw policing dataset before pre-processing



	New_ID	id1	id2	nom_ref1	nom_ref2	name1	name2	dob1	dob2	name_dob1	...	home_dist	crime_dist	offence	role	year	birthday	match	score1	score2	Prediction
0	0	21738	345	4510110630E	191279580Z			26-11-94	13-08-94		...	0.607	0.607	0	1	1.0	0.0	0	3.214	3.642571	0
1	1	21738	420	4510110630E	8826304L			26-11-94	10-03-94		...	0.559	0.559	0	0	1.0	0.0	0	2.118	2.118000	0
2	2	21738	487	4510110630E	11901469C			29-11-94	10-05-94		...	0.510	0.510	0	1	1.0	0.0	0	3.020	3.020000	0

Figure 2: After performing data cleaning on the policing dataset

Machine learning techniques have been implemented using Python to perform investigation of crime suspects who have previously committed other crimes and have had their identity manipulated. In events associated with crime to have a better understanding of an individual preliminary investigation is carried out to figure out if a person has committed any crime before or not and compare it with the latest date when crime was committed. It is analysed to gather and extract information to enhance the performance and to train the model well (Informatica, 2013). Another issue while working with dataset relating to crime is the situation that prevails not just to improve intentional manipulation of data, but detection of accidental errors is also an important aspect to take into account while awaiting accurate results (Bolton & Hand, 2002). In this research identities of two persons are matched using string-matching algorithms such as Jaro-Winkler to get a similarity score between them to solve preliminary problems observed in identity resolution (Wang, Qin, & Wang, 2017). Since fraudsters are clever enough to manipulate data by creating similar details unidentical to their prior submitted details to avoid being caught, the spotlight of the research hovers around creating a model with higher precision and recall value to resolve this issue (Varmedja, Karanovic, Sladojevic, Arsenovic, & Anderla, 2019).

This project removes any duplicate data and scrutinizes it carefully to enlist the attributes that have been intentionally manipulated. So, a possible solution for the prevailing dilemma would be considered by implementing TensorFlow and Keras and have a comparative analysis with other methodologies implemented to selection an appropriate model. The dataset we are working has massive difference in the percentage of manipulated and genuine

classes which complicates the analysis leaving less room for the model to learn from the available manipulated data and learn patterns from it which is also referred as imbalanced dataset (Makki, Assaghir, Taher, Hacid, & Zeineddine, 2019). It has also been clearly depicted with black dots which are the genuine classes and red dots as the fraudulent or manipulated data where only about 0.75% of the total data represents matched records shown as "1" for 296680 records and the remaining 39367698 records are genuine records shown as "0" in Figure 3. Having a lower frequency of fraudulent cases in the dataset makes it crucial and necessary for the model to work in absolute exactness as a slight error in prediction would result in considering criminals as genuine person or even worse by assuming a genuine person as the culprit so additional measures to balance this imbalanced dataset has been taken such as performing oversampling of dataset.

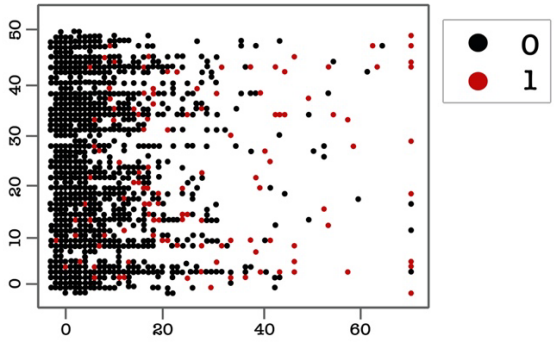


Figure 3: Diagram demonstrating an imbalanced dataset

To overcome this issue while working on imbalance class dataset, different approaches in TensorFlow model has been applied to improve it. Henceforth, TensorFlow and Keras models are implemented which are profoundly efficient and capable models to detect and solve issues related to imbalance data. For optimal solution three additional machine learning techniques are also applied to compare results from each model and select the most optimal outcome and model amongst all.

3. Implementation of Machine Learning Techniques

This section first includes brief introduction to each ML techniques applied to the dataset, background of how each model works with relevant equations and its significance on application to the policing dataset.

3.1 TensorFlow

TensorFlow is an open-source machine learning algorithm to build and train different set of data implementing high level APIs such as Keras to build an easy model that is robust and powerful (Abadi, 2016). TensorFlow works efficiently on classification with text, images, distributed training with Keras, transfer learning and even on classification on imbalanced dataset (Unruh, 2017). And since policing dataset is an instance of imbalanced dataset henceforth TensorFlow was selected for implementation. The phase of training a dataset in TensorFlow is very crucial aspect so various steps such as optimization, setting number of epochs and using early stopping would be considered. Optimization function plays a key role in building a model whereas number of epochs are assigned to improve the model with every iteration of assigned batch size. Additionally, early stopping feature is also initiated which stops the

model early than the assigned number of epochs i.e., 100 for this model, which implies that if there is no significant improvement in the model it stops the model from training further. The early stopping ensures that the model learns new patterns and stops the process when there is nothing new to learn for the training set to avoid biasness and overfitting in the model.

During the phase of building the model, one of the major concerns was to contemplate about taking approached so that both overfit and underfit in the model could be avoided. Overfitting is a condition when the training dataset is very accurate and becomes biased towards a specific prediction for instances resulting as a fraudulent identity or a genuine identity more often and becomes incapable to generalize the test datasets. In contrary, underfitting is the situation when the training dataset is incapable to predict efficiently on the test dataset due to lack of well-trained data. Considering both the scenario, it would be risky specially while working on dataset related to criminal activities or identifying manipulation as it becomes a dilemma to trace the manipulated identities within the huge dataset of 39 million rows. In such cases, even a minor error in prediction can cost a major and long-term effect of punishing an innocent or even worse by setting a criminal free. Hence, in order to avoid such situation weight regularization, adding dropout or increasing the training rate is considered (TensorFlow, 2015).

Another eminent feature while working with TensorFlow is assigning certain number of hidden layers while building the model which is often coined as ANN or Artificial Neural Network which is a dense layer between input and output layer (Figure 4). A number of hidden layers can be assigned to a model depending on the scenario of the dataset and time constraint as number of hidden layers are directly proportional to the time required to execute the code and build a model. Weighted inputs are assigned randomly which are then fine-tuned and is assessed through the process called backpropagation (an algorithm that calculates gradients of the error function with respect to the weight we assigned to our neural network) (Lillicrap, Santoro, Marris, Akerman, & Hinton, 2020).

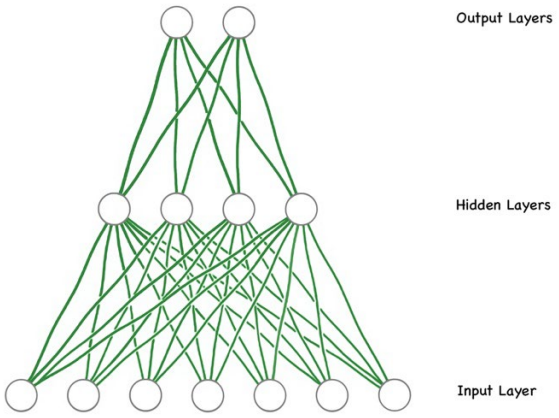


Figure 4: Hidden layers in TensorFlow Model (Valkov, 2017)

Activation function is another aspect that should be carefully considered and applied to the model as it is a mathematically access between an input feeding the current neuron and an output that goes into the next layer. In this model the applied activation function amongst various other functions in Keras API is ReLu (Rectified Linear Unit), which is a proven and efficient way while working with deep neural architecture (Fusinska, 2018). The principle behind the ReLu activation function is to return the value if the input is positive else return a zero (Chen B. , 2021). Selecting an appropriate activation is the most crucial part while building a model because activation functions are the core of any model which determines the efficiency, accuracy and output of any model which can be represented by the following equation (Skalski, 2019) (Equation 1). In the equation $'n'$ represents the exponential power of 2 which begins

as 2^n as 2, 4, 8,16, 32 all the way up to 512 and *'n-1'* represents the combination of various power value for hidden dense layers while building the model.

$$T_f{}^{[n]} = W^{Tf}. \; a_r{}^{[n-1]} + b \qquad (1)$$

Equation 1: The concept behind assigning activation function in TensorFlow, where T_f represents TensorFlow, n represents the exponential power of 2 that will be assigned while building the model, $n-1$ is applied when adding different combinations of dense layers, W represents the weight assigned, a_r represents the activation function assigned which is ReLu and b represents the bias assigned.

Adam optimization algorithm that is an additional aspect that has been implemented while building this model which helps in determining how to use the difference between the results obtained and the adjustments required to be performed on the weights on the nodes so that the model obtains an optimal solution (Arya & Sastry G., 2020). Particularly as the policing dataset is quite huge with approximately 23 csv (comma-separated values) files and 39 million rows and Adam optimizer works quite efficiently with larger dataset, has smart learning rate, momentum behaviours and above all it requires low memory (Zhang C. , 2018). All these attributes of Adam optimization would eventually help execute the code efficiently and consume less memory which would ultimately accelerate the execution rate of the model and prediction, so hence it has been considered.

3.2 Support Vector Machine

Support Vector Machine is a supervised machine learning technique that classifies the entire dataset into two specific group of classification problem (Stecanella, 2017). The objective of SVM considering policing dataset is to find an optimal hyperplane that distinctly classifies the data effectively into two groups i.e., one with fraudulent identities and the other with genuine identities where the data points are not extremely far or close to the hyperplane (Singh, Gupta, Rastogi, Chandel, & Riyaz, 2012). So, to tune the parameters accurately training and testing set should be split accordingly and as SVM is an effective algorithm in high dimensional scenarios as it uses a subset of training dataset in the decision function it is also referred as support vectors because it's memory efficient (Ray, 2017). The principle behind dividing the dataset into two separate parts with the hyperplane in SVM is given by the following equation (Equation 2).

$$h^M(a) + b = 0 \qquad (2)$$

Equation 2: Support Vector Machine, where h represents weight vector of hyperplane, m represents maximize, a represents input vector & b represents intercept and bias term of the hyperplane.

SVM is a powerful machine learning technique that is in the sector of classification, functions estimation problems and even distribution estimation (Almasi & Khooban, 2017). SVM is often seen applicable in the field of fraud detection because of its requirement for the training dataset to look for global minimum, their elementary geometric interpretation and the optimum hyperplane (Radhika, 2020). This algorithm comparatively takes a bit longer time to execute as many aspects are considered while building the model and also to get the optimal hyperplane amongst many considered possible hyperplanes to classify the data.

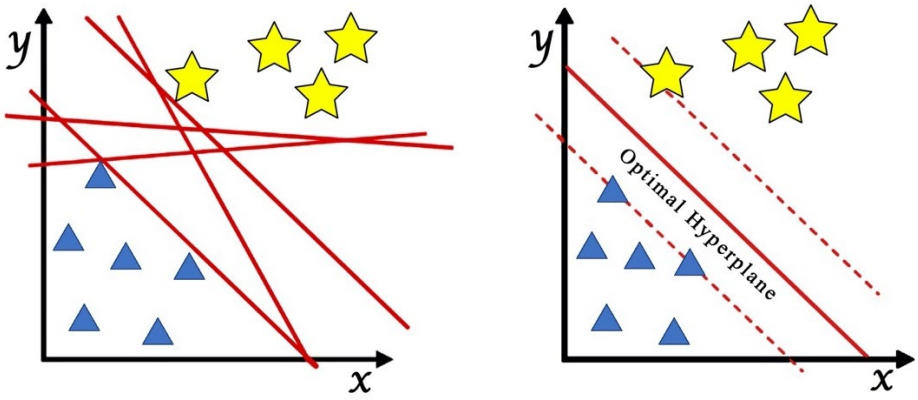


Figure 5: Possible and optimal hyperplanes

3.3 Naïve Bayes

Amongst many types of models on Naïve Bayes, Bernoulli Naïve Bayes classifier is commonly implemented on text documents specifically for text classification where feature vector is binary column which in the policing dataset is the "match" column with values such as 0s and 1s (Singh, Bhatt, Bedi, & Mishra, 2020). Bayes theorem is relatively faster while executing even in larger dataset as even the entire 39 million rows data was processed in about an hour. As this algorithm is particularly effective to work with larger dataset and also works on calculating a new attribute which is a posterior probability given by the equation shown below makes it stand out distinctly from other models (Equation 3).

$$P(F|G) = P(G|F)\ P(F) \ / \ P(G) \qquad (3)$$

Equation 3: Bayes theorem for probability calculation, where ***F*** represents Fraudulent data, ***G*** represents Genuine data and ***P*** represents probability.

The Naïve Bayes theorem considers each feature independent to one another which can also lead to complexity as this changes the model's initial features where attributes are correlated to each other. So, basically the algorithm performs a conditional probability of each class for the input dataset and then these are changed with the assigned class label which is the "match" column (the match column in the policing dataset is a column with binary values where 1 represents fraudulent and 0 represents genuine data) that determines if each row of record is fraudulent or not. And then finally, the class with the largest probability is selected as classification referring to MAP (Maximum a posteriori) estimation decision rule to implement Naïve Bayes algorithm to predict the final outcome.

When glancing the algorithm superficially it seems that it requires only to process input, build a model using Naïve Bayes algorithm for probabilistic distribution and predict the output. But there is more to do within the Naïve Bayes theorem to get an optimum result for our policing dataset. Amongst many important features of implementing Naïve Bayes algorithm, a vital aspect is that it minimizes the effect of outliers in the dataset as it emphasizes on the probabilistic distribution and neglects the skew otherwise done by outliers on a dataset. Additionally, when working with Naïve Bayes algorithm burden of collecting large set of data is also not required as this model is efficient to produce accurate outcomes even when working on smaller amount of training dataset. So, this model is equally effective in scenarios when less dataset is assigned as training set but having a relatively larger set ensures better prediction.

3.4 K-Nearest Neighbour

KNN or the K-nearest neighbour algorithm is a simple application of machine learning algorithm that can be implemented on classification as well as regression case scenarios (Harrison, 2018). Similar to any other supervised learning model KNN also learns from the assigned labels in training set and predicts the unlabelled test data but a unique feature it possesses is assigning a particular number which is known as the number of neighbours for the value of K. According to the number assigned for the value of K, the dataset is grouped and the distance between two points is calculated using Euclidean Distance formula to check how far or close any point is (Equation 4). Euclidean distance calculates the shortest distance between any two points without concerning in which dimensions they are placed (Pandey, 2021). In the equation the C_i and M_f helps in classifying the observation into either fraudulent or genuine data group and the condition $I\ (Ci\ =\ a)$ the value is 1 else 0. (Statistics Libre Texts , 2020).

$$P\ (C\ =\ a\ |\ M_f\ =\ m_0)\ =\ 1/k\ \Sigma\ I\ (C_i\ =\ a) \qquad (4)$$

Equation 4: An approach to get optimal result from K-Nearest neighbour, $\boldsymbol{P}$ represents probability, $\boldsymbol{C_i}$ is a class label, $\boldsymbol{a}$ represents any class, $\boldsymbol{M_f}$ is matrix of features, $\boldsymbol{m_o}$ is test observation, $\boldsymbol{k}$ is the value for KNN which is a positive number mostly odd numbers, $\boldsymbol{\Sigma}$ represents the sum where $\boldsymbol{i \in S_k}$ (i belongs to S_k) where $\boldsymbol{S_k}$ is set of K-nearest observation which is also the member of class $\boldsymbol{a}$.

There is no specific rule to assign the value of K, therefore it mainly depends on the scenario or quantity of the dataset, but it requires a rigorous testing of random values of K in the model to select the best result (Gokte, 2020). Rather than just assigning any number to K it should be an odd number as the value of K so as to avoid equal split between two classified data, so values such as 3, 5, 7 and other numbers are assigned as the value of K (Figure 6).

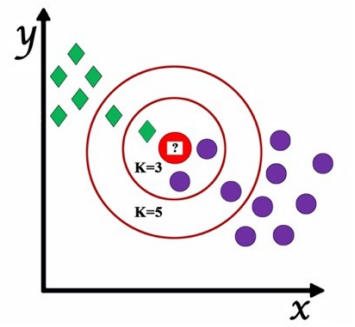


Figure 6: KNN neighbours' diagrammatic explanation

4. Results and Discussion

This section includes detailed information on the workflow of the models and the results from each model in order to compare and evaluate the outcomes.

4.1 Workflow of various Machine Learning approaches

The working of TensorFlow model comprises of various important steps which begins with implementing the cleaned training dataset to build a model. The cleaning phase includes removable of unnecessary attributes, clustering into groups, removing nan values and more. Building the model includes implementation of various features such as assigning specific number of hidden layers, early stopping and finally setting specific optimization and activation functions. Assigning hidden layers is not only a crucial aspect of TensorFlow model but also a very tedious task as there is no specific rule to assign the layers so multiple values and combinations should be tried to get the best outcome. Then the prediction model is executed along with evaluation to check the authenticity of the model before implementing the test data. The result is then visually represented with confusion matrix and ROC curves to provide in-depth insight to the data (Figure 7).

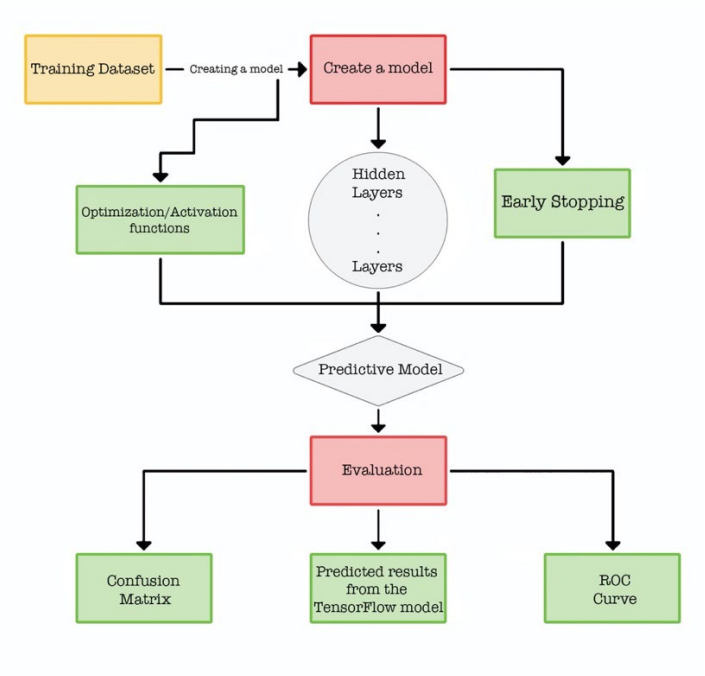


Figure 7: Architecture of workflow implementing the TensorFlow model

Along with TensorFlow model other three models which are SVM, KNN and Naive bayes were also considered to compare the results with TensorFlow. The other three models have similar steps to train and build the model which initially begin with cleaning the training dataset, followed by segmentation and labelling of data (Figure 8). Then the crucial phase of training the model takes place which first requires selection of appropriate classification model which is either SVM, KNN or Naïve Bayes. Under SVM model, SVM classifier is selected, under Naïve Bayes the binomial model of Bernoulli Naïve Bayes theorem is selected and finally for KNN classification the values of K are selected generally odd numbers. Then the testing phase includes testing of dataset separately for each model and then finally visualizing the predicted result.

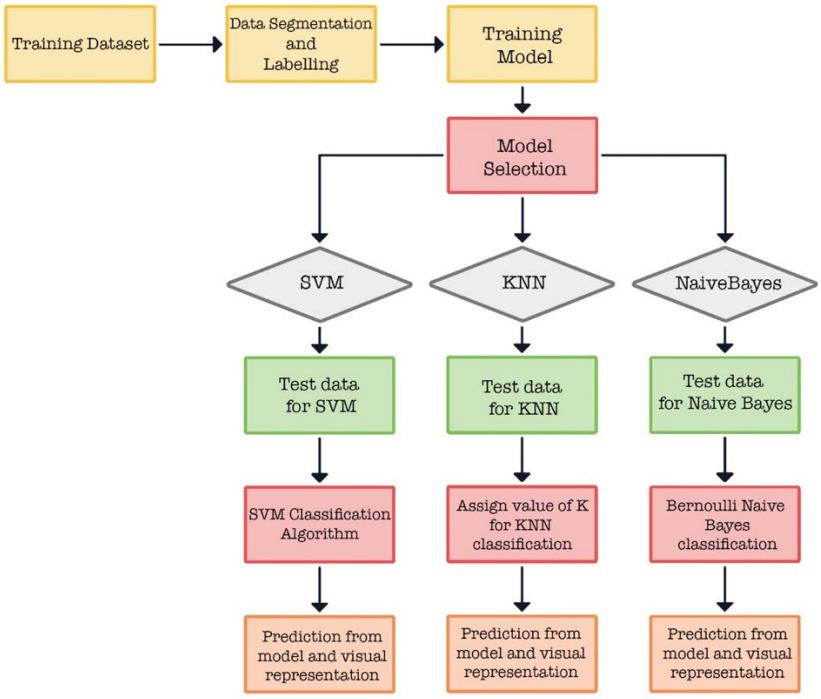


Figure 8: Architecture of workflow implementing various NN methodologies

4.2 Confusion Matrix

A confusion matrix describes performance of a classification model, so it was applied to predict outcomes of the policing dataset to analyse the total number of correct and incorrect predictions made by each model (Brownlee, 2016). Models having lesser number of False Positives and False Negative values are considered as the most efficient model for prediction, henceforth confusion matrix plays a pivotal role in reasoning the degree of accuracy or inaccuracy of any model and upon analysing predictions made from each model, TensorFlow has produced the optimal result amongst other three applied methodologies as it has the least number of False Negative and False Positive values (Data School, 2014).

TensorFlow is considered to be the best models amongst other considered models primarily on the basis of the predictions it made with just 36 False Negative and 107 False Positive values but additionally it has the highest degree of accuracy and precision amongst all the other considered models too (Figure 9). A higher number of True Positive and True Negative values substantiates the fact that the model will predict greater number of identities successfully and have higher accuracy while listing the fraudulent identities although working on an imbalanced dataset. Having a fewer number of False Negative or False positive data indicates that the model has a lesser number of inaccurate predictions than from other models. Another lucrative aspect of implementing TensorFlow is that these inaccuracies can be reduced by tunning the model through various ways such as adding in weights, changing optimization function, adjusting number of hidden layers and even changing number of epochs.

It was not possible to obtain the result at once as various parameters have to be applied, tested, changed and the entire process of building the model with the improvised training criteria has been continuously carried out. When the number of hidden layers were increased or decreased the model predicted inaccurately with higher rate of False

Negative and False Positive values which ended up resulting in predicting identities that are fraudulent in reality as genuine identities and genuine person listed to be the suspect of the crime. So, all the parameters were tested individually taking notes on each minute changes and outcomes it produced to analysing and making it relevant to finalise which parameters resulted in better prediction. Another important aspect that was considered was to get higher number of True Positive and True Negative values which illustrates that the model has predicted well. Prominently the greater these values of True Positive and True Negative the better the model has predicted as the True Positive values represents those data that are genuine and the model too predicts them as genuine. As its counterpart the True Negative values portrays those data that are fraudulent identities in real world and the model has successfully predicted it as fraudulent identities. So, for the different number of hidden layers with the exponential value beginning from 2 till 512 were tested individually and double or triple combination of layers of the same and/or different number of hidden layers were also tested to get the best combination while building the model.

As there is no specific rule on assigning the number of hidden layers, intrinsically random number of hidden layers were implemented on the basis of trial-and-error method which is trying all the possible number of hidden layers to get the best combination. To get the best combination of hidden layers individual prediction obtained by assigning a specific parameter was tracked and noted, which was later tabulated to get an appropriate result. As it was a tedious task which is equally prone to errors, so all the records were carefully jotted and after tabulating all the results, the best combinations were carefully skimmed out. Higher precision, higher accuracy and lower loss values were amongst some of other attributes that were considered.

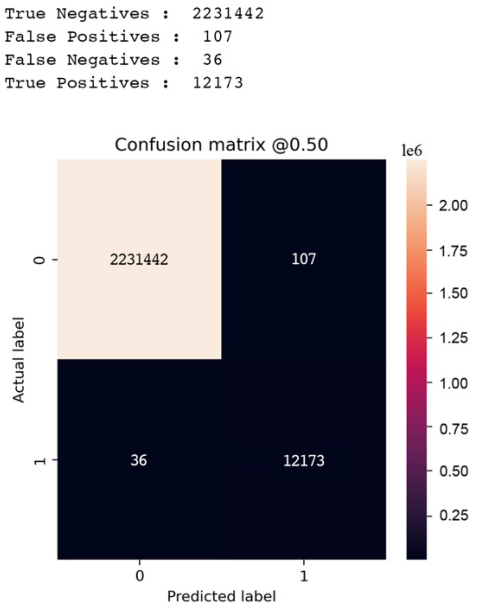


Figure 9: Confusion Matrix result from TensorFlow

As the results from TensorFlow was extremely accurate so to evaluate whether the results were reliable, the model was not overfitted and the accuracy would differ or not when various parameters were changed such as altering epoch number, adding weight and bias, over sampling and resampling the data were implemented. When the parameters were set in TensorFlow with combination of various hidden layers, optimization and activation functions, it produced 99.99% accuracy and although epoch was set to 100 and the model was early stopped when the training model has no new pattern to learn. Also weights and bias in the model are applied which are setting bias to zero and also careful bias is set for a very important purpose as of zero bias is added so that it randomises to prevent the neurons learning exact

same things and break the symmetry between neurons within a same layer. When bias weights are not applied it makes the model in the same gradient to be the xerox of the earlier outcome. Whereas with careful bias it loads the previously saved model which has the primarily initialised bias values.

As the dataset is imbalanced dataset with very few positive samples for the training model to learn from, hence weights are added to the dataset, so a heavier weight is added for the minimal data that is available to improve the prediction. Another approach that is considered for model improvement is oversampling a dataset, in this approach the minority class amongst the two are randomly duplicated, which in case of policing dataset is for the match column having values as 1, that are the fraudulent cases when two strings are compared, please refer to Figure 10. For oversampling of dataset multiple methods are applied for testing, which is applying early stopping, applying only specific number of epochs, applying 10 times more epoch that normally applied value or even limiting epochs in some cases.

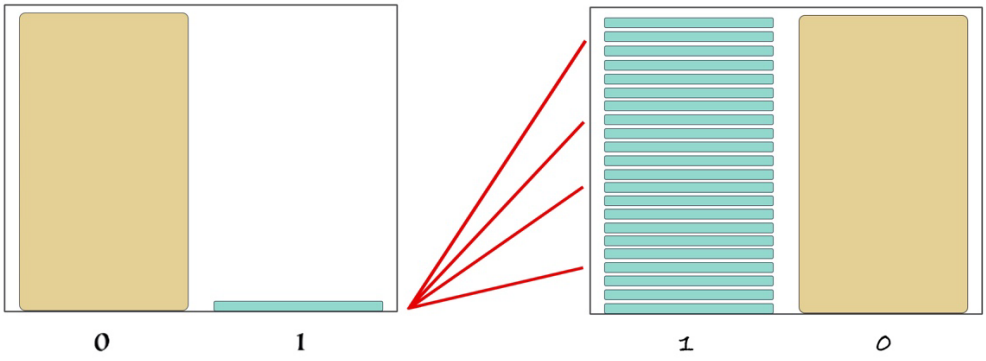


Figure 10: Oversampling on imbalanced policing dataset

Amongst all applied parameters when epoch numbers are limited to only a certain value suddenly the accuracy decreases as the model is not able to learn and improve itself and the training phase is abruptly stopped. Hence another parameter to set a higher epoch number with early stopping is assigned which also has a relatively less accuracy. But in contrary the best combination of all parameters for oversampling is when zero bias with 100 epochs and early stopping is set which gives the best result with 99.98% accuracy. Although TensorFlow at a glance might seems to be a model with higher accuracy but it continues to produce good results even when various approaches or alteration are done to the parameters while building the model so it is the best model with highest percentage of accuracy amongst all the other applied models which is 99.99% with the model that has careful bias and a custom number set for hidden layer's combination finalised after multiple hit-and-trial method as shown in Table 1.

Table 1: Various parameters applied on TensorFlow to obtain the best result

Model type	Percentage	Number of epochs
Standard model with careful bias	99.99%	100 epochs early stop at 77
Zero Bias added to model	99.92%	100 epochs early stop at 20
Class Weights added	99.91%	100 epochs early stop at 29
Oversampling data (with zero bias)	99.96%	100 epochs early stop at 44

Oversampling data (with zero bias)	99.93%	Epochs limited to 5 epochs
Oversampling data (with zero bias)	99.62%	Epochs increased to 1000 epochs

Second model considered for prediction was KNN model which has a crucial task of assigning a value of K which is the core of this methodology that is assigning the K-nearest value. The number of K is considered as the main aspect of assigning class of that specific value of neighbours. So, initially K value was assigned as 3 so this algorithm first groups three data value from the training set in random and then takes one value from the testing set so depending on the highest frequency of the training set either as fraudulent or genuine it then assigns the testing set to be fraudulent or genuine. In this KNN model initially when K was assigned as 3, it resulted with approximately double the number of False Negatives value which is 58 and 174 False Positives values. This subsequently decreases the True Negative and True Positive values as compared to prediction from TensorFlow although it does have better result than from SVM and Naïve Bayes as shown in Figure 11.

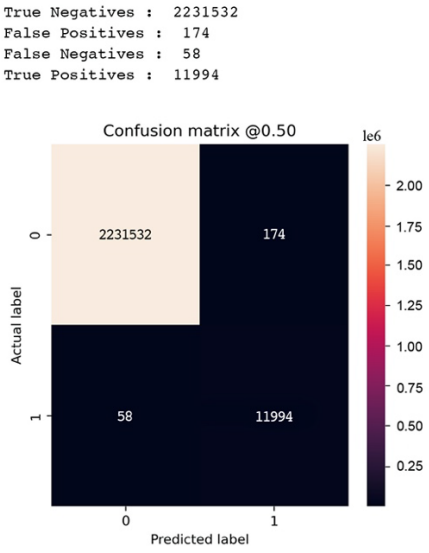


Figure 11: Confusion Matrix result from KNN model when K=3

Since KNN also produced good predictions and its accuracy was quite similar to TensorFlow's result so some parameters were altered to the KNN model to investigate if it would improve or worsen the prediction. For this process only odd numbers are assigned as the value of K to avoid even split of data amongst the groups for which initially the K value was set to 3, followed by assigning other numbers such as 5, 7, 9, 11, 13, 15. After applying some K-values in the sequences of odd numbers beginning from 3 up to 15 there were just some minor changes in terms of accuracy percentage and False Negative and False Positive values so a larger random leaps of K-values were then considered which included K-values as 39, 59, 69, 79 and 99. About twelve different K values were used to make the predictions but with each increment in the value of K, not only the False Positive and False Negative values were escalated but the crucial component which is the accuracy was slightly decreased every time. Although there is no drastic change in the predictions when K-values assigned were close odd numbers but still the initial assignment of K-value as 3 has the highest accuracy rate amongst any other K-value and at the same time it has the lowest number of False Negative and False Positives. This ultimately indicates that the predictions when K-value is set as 3 has the most accurate predictions.

Table 2: Different K-values and predictions made

K-Value	False Negative	False Positive
K = 3	58	174
K = 5	63	221
K = 7	71	258
K = 9	72	286
K = 11	81	320
K = 13	91	366
K = 15	100	387
K = 39	190	568
K = 59	243	641
K = 69	246	664
K = 79	273	670
K = 99	275	711

Naïve Bayes model, although executes the codes in relatively high speed than any other model but the prediction from this model is not quite efficient as it's agility and therefore is ranked third out of the four models tested. Upon analysis the False Negative value has increased to 206 (whereas TensorFlow had just 36 and KNN had 58) and in addition to this, the False Positive value has a massive difference than the first two models where in TensorFlow model it is just about 107 and in KNN has 174 but in Naïve Bayes it is 9,196 records (Figure 12). This result clearly illustrates that a greater number of genuine people will be predicted and considered as suspect and would be interrogated if this model is implemented in real world investigation of fraudulent identities. This is the greatest drawback of this model when implemented on crime related dataset where the inefficiency of model prediction is clearly depicted with an elevated result of False Positive and False Negative values.

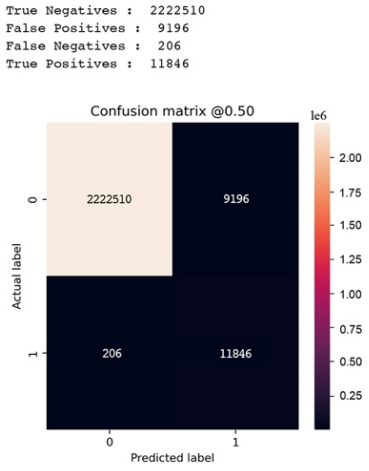


Figure 12: Confusion Matrix result from Naïve Bayes

One of the models that demonstrates the least efficiency in predicting result on the policing dataset was the algorithm Support Vector Machine. In this model not only, the False Positive value have massive figures like in Naïve Bayes model but an inflation in the False Negative values as well. Which directly indicates that about 12,058 records were predicted as genuine when they were fraudulent identities and 12,230 have be detected as fraudulent identities whereas in reality, they aren't fraudulent IDs (Figure 13). Upon application of this model in real world, it would greatly

impact and have adverse effect on investigation involving criminal data and identifying their identities as the model is quite insignificant with prediction of fraudulent and genuine identities.

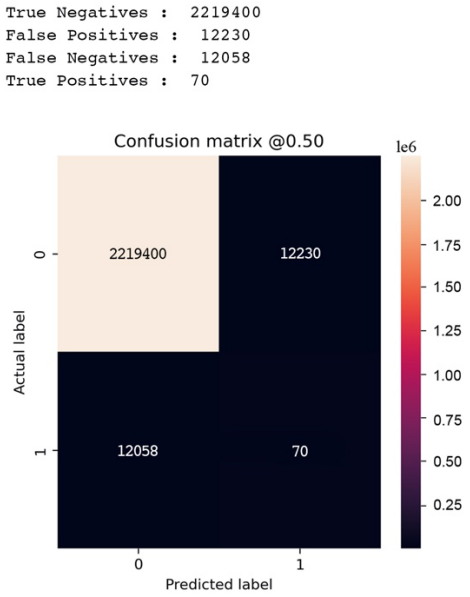


Figure 13: Confusion Matrix result from SVM

After an in-depth consideration of all the four model and their outcomes represented by confusion matrix, it is clearly evident that TensorFlow is the uttermost optimal algorithms to be implemented on policing dataset and identifying identities whereas in contrary Support Vector Machine is the least competent model to be implemented. Not only the results of confusion matrix for these models determine their efficiency but the possibility of decreasing errors and reducing inefficient predictions makes models preferable than the other. TensorFlow not only has one of the best results for confusion matrix with lowest number of False Negative and False Positive values but it's accreditation of allowing to tune the model by adjusting parameters which includes selection of appropriate optimization and activation functions, adding weights, adding bias, oversampling and resampling the data and custom-defined hidden number of layers makes TensorFlow one of the best models for application on criminal fraud detection.

Additionally, when only the False positive and False Negative values are skimmed out from the confusion matrix to better understand how each model has predicted, the result clearly demonstrates that TensorFlow has the least number of False Negative value with just 36 records whereas SVM predicted 12058 records. Similarly, for the False Positive values too TensorFlow just predicted about 107 records but SVM on the other hand predicted 12230 records. The difference between these model's predictions is quite significant and since working on criminal records requires very high accuracy and precision as a minor inaccuracy in prediction would end up leading genuine person to be interrogated and criminals to be set free, so TensorFlow and KNN are only considered suitable to be applied on similar scenarios to policing dataset.

Table 3: Comparing False Negative and positive values of all models

Model's Name	**FN**	**FP**
TensorFlow	36	107
KNN	58	174

Naïve Bayes	206	9196
SVM	12058	12230

4.3 ROC curve and Precision recall curve

ROC curves are another efficient way to test the model and their efficiency to obtain a fair result and conclude which model is best amongst all implemented models. The ROC curve clearly indicates that TensorFlow again has predicted best outcome amongst all the four models implemented as its closer to 1 or has 100% accuracy and is further from the centre classifying threshold. After TensorFlow is KNN with 99.94% and Naïve Bayes with 90.61% accuracy. SVM predicted worst outcome amongst all models for confusion matrix and for the ROC curves too it has the worst prediction with just 85.69% and is the closest to the threshold or the cut-off value set at the centre. Since the results that are closer to the threshold is considered to be a bad model for prediction so SVM clears is not a model to be considered for similar dataset. Also, it demonstrates how well the TensorFlow model predicts the positive classes which are the fraudulent identities in case of policing dataset when the actual outcome is also positive or fraudulent ID. Another method to evaluate a ROC curve is observing Area Under the Curve (AUC) and as with TensorFlow it has more area under the curve than any other model so it subsequently demonstrates that it has better prediction. Whereas for SVM model it has the lowest area under its curve indicating it has predicted the lowest number of positive classes (Figure 14).

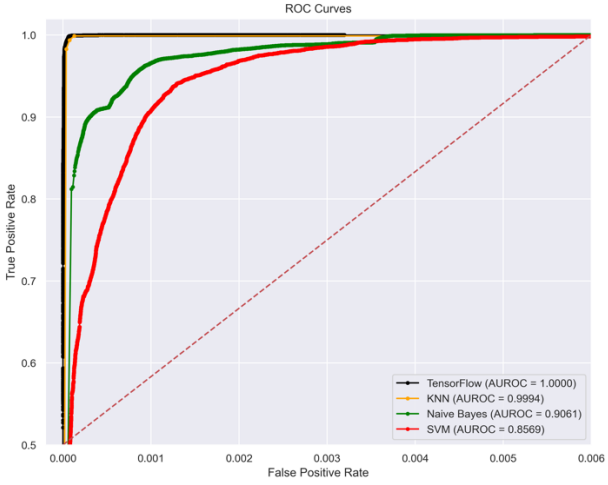


Figure 14: ROC curve of all models

The precision-recall curve demonstrates the trade-off between True Positive and False positive predictive value for all the four predictive models implemented (Ekelund, 2017). As the policing dataset has more genuine IDs and comparative fewer numbers of fraudulent identities so particular on such data precision-recall curves are very good way to analyse the results. A precision-recall curve demonstrates much clear view to analyse how each model has predicted in which the curve again proves that TensorFlow has predicted most efficiently as the curve is closer to 1. So, all those model that hasn't predicted well produces curves further away from 1 which are curves created by SVM and Naïve Bayes.

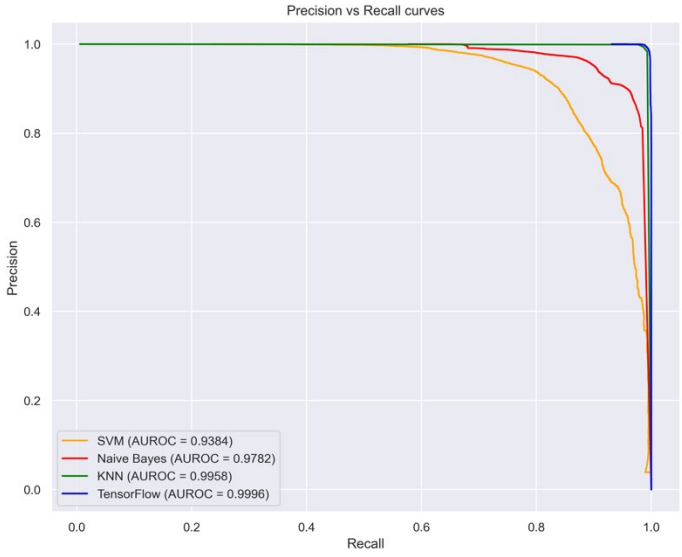


Figure 15: Precision Curve of all models

4.4 Accuracy Table

To get a better understanding and analysis on outcomes from each model and ultimately to select the most optimum model an accuracy table is created by accumulating results from ROC curve. TensorFlow model has predicted outcome with 99.99% which is one of the most optimal outcomes not only in terms of the rate of accuracy, but it has equally effective result from confusion matrix, ROC and Precision recall curve as well. To make this result evident many other conditions and parameters were changed but TensorFlow had higher accuracy than any other model every single time. Similarly, after TensorFlow the second-best prediction is by KNN model. But the results from Naïve Bayes and SVM model are not that effective, and the predictions include numerous incorrect outcomes where fraudulent identities are considered genuine and vice-versa. Although Naïve Bayes model takes the least execution time compared to any other applied models, but the results obtained are not sufficient to make strong conclusions as the accuracy of prediction from the model is comparatively very low. The other two models, namely KNN and SVM have higher execution time of over 6 hours but neither of the models have good prediction score. The execution time is not the primary requirement to create a good prediction. The suitable model is the one with the highest accuracy and the least inaccurate predictions which is the prerequisite of the model; in this case for the policing dataset the TensorFlow and Keras were the best fit.

The TensorFlow model take approximately 1 hour 40 minutes to build and deliver the results with relatively higher precision and better result than the other 3 models applied and has comparatively less execution time too.

Table 4: Accuracy of different models based on ROC curve results

Model	Accuracy	Execution Time
TensorFlow	99.99 %	1 hr 40 mins
KNN	99.94 %	6 hr 48 mins

Naïve Bayes	90.61 %	5 mins
SVM	85.69 %	6 hrs 32 mins

Numerous computations have been carried out to analyse the outcomes from the model, and the accuracy of the TensorFlow model has the highest accuracy compared to other three applied ML models. In final prediction from the TensorFlow model, it has successfully been predicted that all 5 suspects in the dataset out of 39 million records have manipulated their identities, highlighted with a rectangular box in Figure 16. This demonstrates that the TensorFlow model can be implemented to resolve identity issues with reasonable accurate predictions. In contrast the predictions from other three ML models are not as accurate as TensorFlow, a few or none of the suspects have been predicted accurately.

Unnamed: 0	New_ID	id1	id2	nom_ref1	nom_ref2	name1	name2	dob1	dob2	name_dob1	...	home_dist	crime_dist	offence	role	year	birthday	match	Prediction
61	8570040	18982	7081	51361681U	51189179681U			26-05-92	26-05-92		...	0.975	0.975	1	1	1.0	1.0	1	1
235	304	21738	25704	4510110630E	4570903910630E			26-11-94	26-11-94		...	0.942	0.942	0	1	1.0	1.0	1	1
358724	6960104	454987	240572	16589105T	16915871105T			16-03-90	16-03-90		...	0.978	1.000	1	1	1.0	1.0	1	1
32400	10033117	330565	331276	279192762	2711007412762			23-11-99	23-11-99		...	0.936	0.936	1	1	1.0	1.0	1	1
32509	10033394	330565	345821	279192762	2711007412762			23-11-99	23-11-99		...	0.911	0.911	1	1	1.0	1.0	1	1

5 rows × 24 columns

Figure 16: Accurate prediction of 5 suspects from TensorFlow model

Although the TensorFlow model is the best model in this research, it is not perfect. There are some inaccurate predictions, as they are highlighted with a rectangular box in Figure 17.

	New_ID	id1	id2	nom_ref1	nom_ref2	name1	name2	dob1	dob2	name_dob1	...	home_dist	crime_dist	offence	role	year	birthday	match	Prediction
2	32566965	213039	373241	4936442A	1112288923C			12-01-85	12-01-85		...	0.608	0.608	0	1	1.0	1.0	0	1
14	2005	21738	179545	4510110630E	109535242E			26-11-94	26-11-94		...	0.669	0.669	0	1	1.0	1.0	0	1
35	32535693	594473	66561	40664456L	401215198H			29-11-05	29-11-05		...	0.824	1.000	1	1	1.0	1.0	0	1

3 rows × 24 columns

Figure 17: Some inaccurate predictions from TensorFlow model

5. Conclusion

This section includes selection of the most optimum results and explanations on why a particular model is best over the other applied ML models. This section concludes the overall results of the paper.

This research includes implementation of TensorFlow model in detection of fraudulent identities as TensorFlow has never been applied to criminal fraud detection and this adds to the novelty of this paper. Along with TensorFlow model other three methodologies namely SVM, Naïve Bayes and KNN were also implemented to investigate the results from each of the model and have a comparative study to select the best model amongst all with the highest accuracy and least error in prediction of fraudulent identities. This methodology was applied on anonymized policing dataset used as a part of SPIRIT project funded by the European Union's Horizon 2020 initiative. The dataset was trained on various models and only TensorFlow and Keras integrated model successfully predicted the 5 main suspects from the large dataset, so the tedious task of the police officers to investigate the manipulated identities has been eased. This model ultimately helps investigating criminal records and provides more insight on what records, crimes, age group, gender, etc are often the most manipulated attributes. This model can further be applied to similar areas where manipulated identities are prevalent, as criminals disguise themselves as genuine citizens such as in credit fraud, travel record fraud, etc. The models were implemented with appropriate training criteria's, optimization and activation functions to improve prediction on either intentional or unintentional manipulation of data to detect false identities on the huge policing dataset. To obtain an optimal result with TensorFlow early stopping functionality was also implemented which stops the model early if no significant change is observed even after many epochs are also executed.

This articles also demonstrates how TensorFlow offers more functionalities and parameters to be altered for improvement of the model than just traditional method of simply training and testing of dataset in other machine learning approach. The functionalities include adjusting and setting new parameters for optimization, try various combination and/or arbitrary values for hidden layers, applying bias weight and oversampling of data to improve the prediction. All these adjustments make TensorFlow the most optimal model amongst all the other applied methodologies as there are many places where tunning the training dataset is possible which eventually makes the prediction of test dataset perfect with a higher accuracy than any other applied model. Also, TensorFlow is the only model that successfully predicted 5 suspects out of 39 million records.

Given all the results, the identity resolution approach implementing TensorFlow on the dataset produced the best result in investigating false identities for police and law enforcement agencies when 80-20 sampling technique with various adjustments made during building the model. Additionally, KNN model too produces good results so various K-values were implemented to get the suitable value that produced the best prediction. Support Vector Machine is the most unreliable methodology amongst all the four implemented as it has greater number of False Positive and False Negative values which indicates that the SVM model predicts quite inaccurately by predicting false identities as genuine and genuine ones as false identities. The outcome from SVM is unrealistic and imprecise as it results in interrogating genuine people and setting criminals free. In such cases Naïve Bayes and SVM are absolutely unacceptable and insignificant in maintaining police records and analysing criminal history of any individual. There are some limitations of this research as the work only covers application of single model at a particular time on the dataset. To improve the accuracy and for a better prediction, cascading of ML models can be studied. Also, in this paper we have only used one policing dataset, transfer learning can be applied to other crime-related data too, and analysis on the efficiency of the model can be studied. Possible future work includes improving model performance by tunning various attributes, changing parameters and altering number of hidden layers previously set. Also, improvement on

currently applied methodology such as by changing parameter of K-value in KNN model to improve its efficiency. Additionally, study on other methodology that would be suitable to work with the policing dataset and provide accurate result can also be investigated.

Acknowledgements

This work was supported by the European Union Horizon SPIRIT project, Grant Number 786993. We would like to thank all our SPIRIT Project partners who provided us recommendation and feedbacks that greatly assisted in the improvement of this research.

Reference

Abadi, M. (2016). TensorFlow: learning functions at scale. *21st ACM SIGPLAN International Conference on Functional Programming* (pp. 1-5). New York: ACM.

Ahmed, S. R. (2020). Identity theft and Identity Fraud. In *Preventing Identity Crime: Identity Theft and Identity Fraud* (2nd Edition ed., pp. 4-20). Boston.

Ahmed, S., Gentili, M., Sierra-Sosa, D., & Elmaghraby, A. S. (2022). Multi-layer data integration technique for combining heterogeneous crime data. *Elsevier*, 2-15.

Almasi, O. N., & Khooban, M. H. (2017). A parsimonious SVM model selection criterion for classification of real-world data sets via an adaptive population-based algorithm. *SpringerLink*(30), 3421-3429.

Arya, M., & Sastry G., H. (2020). DEAL – 'Deep Ensemble ALgorithm' Framework for Credit Card Fraud Detection in Real-Time Data Stream with Google TensorFlow. *Taylor & Francis Online, 8*(2), 71-83.

Bartunov, S., Korshunov, A., Park, S.-T., Ryu, W., & Lee, H. (2012). Joint Link-Attribute User Identity Resolution in Online Social Networks. 25-30.

Bolton, R. J., & Hand, D. J. (2002). Statistical Fraud Detection: A Review. *Statistical Science, 17*(3), 235–255.

Brownlee, J. (2016). *Machine Learning Algortihm From Scratch with Python* . Machine Learning Mastery.

Chen, B. (2021, January 3). 7 popular activation functions you should know in Deep Learning and how to use them with Keras and TensorFlow 2. *Towards Data Science*.

Chen, H., Atabakhsh, H., Wang, A. G., Kaza, S., Tseng, L. C., Wang, Y., . . . Violette, C. (2006). A Crime Data Mining Approach to Developing Border Safe Research. *DL ACM*, 49-50.

Cui, J., Wang, H., Xue, W., & Wang, B. (2021). Application of identity resolution and blockchain technology in the whole industrial chain management of electrical equipment. *International Conference on Advances in Physical Sciences and Data Processing* (pp. 1-7). Qingdao: IOP Science. Retrieved August 23, 2020, from https://www.netowl.com/aml-kyc-pep

Data School. (2014, March 25). Simple guide to confusion matrix terminology. *Data School*.

Ekelund, S. (2017, April). Precision-recall curves – what are they and how are they used? *Acute Care Testing org*, pp. 1-8.

Furtado, K. (2009). Identity Resolution: Technology Challenges and Strategic Imperatives for Insurance Industry Leaders. *Strategy meets action*.

Fusinska, B. (2018, February 14). Building deep learning neural networks using TensorFlow layers. *O'Reilly*.

Gokte, S. A. (2020). Most Popular Distance Metrics Used in KNN and When to Use Them. *KD Nuggets*.

Harrison, O. (2018, September 10). Machine Learning Basics with the K-Nearest Neighbors Algorithm. *Towards Data Science*.

IBM. (2006, May 23). IBM EAS. *IBM Entity Analytic Solutions*.

Informatica. (2013). *Identity Resolution in Law Enforcement.* Redwood City, CA: informatica.

Kou, G., Xu, Y., Peng, Y., Shen, F., Chen, Y., Chang, K., & Kou, S. (2020). (http://creativecommons.org/licenses/by-nc-nd/4.0/).Bankruptcy prediction for SMEs using transactional data and two-stage multiobjective feature selection. *Elsevier*, 1-10.

Li, J., & Wang, A. G. (2015). A framework of identity resolution: evaluating identity attributes and matching algorithms. *Security Informatics, 4*(16), 2-12.

Lillicrap, T. P., Santoro, A., Marris, L., Akerman, C. J., & Hinton, G. (2020). Backpropagation and the brain. (H. Wechsler, Ed.) *Nature Reviews Neuroscience volume, 21*, 335-346.

Makki, S., Assaghir, Z., Taher, Y. H., Hacid, M.-S., & Zeineddine, H. (2019). An Experimental Study With Imbalanced Classification Approaches for Credit Card Fraud Detection. *IEEE Access, 7*, 93010-93022.

Moir, C., & Dean, J. (2015). *A Machine Learning approach to Generic Entity Resolution in support of Cyber Situation Awareness.* Sydney: Proceedings of the 38th Australasian Computer Science Conference (ACSC 2015).

Pandey, A. (2021, January 7). The Math Behind KNN. *Exploring the metric functions used in K-Nearest Neighbour model*.

Phua, C., Smith-Miles, K., Lee, V., & Gayler, R. (2010). Resilient Identity Crime Detection. *IEEE Transactions on Knowledge and Data Engineering* , 533-546.

Radhika. (2020, October 23). The Mathematics Behind Support Vector Machine Algorithm (SVM). *Data Science Blogathon*.

Ray, S. (2017, September 13). Understanding Support Vector Machine(SVM) algorithm from examples (along with code). *Analytics Vidhya*.

Sakpere, A. B., & Kayem2, A. V. (2018). On Anonymizing Streaming Crime Data: A Solution Approach for Resource Constrained Environments. *IFIP International Federation for Information Processing 2018* (pp. 170–186). AG: Springer International Publishing.

School, I. P. (2021, August 16). Why Python is the preferred language for Machine Learning? *Weekly Data Science News*.

Singh, G., Gupta, R., Rastogi, A., Chandel, M. D., & Riyaz, A. (2012). A Machine Learning Approach for Detection of Fraud based on SVM. *International Journal of Scientific Engineering and Technology, 1*(3), 194-198.

Singh, M., Bhatt, M. W., Bedi, H. S., & Mishra, U. (2020, September 11). Performance of bernoulli's naive bayes classifier in the detection of fake news. *International Conference on Future Information Technology.* Panjab: Elsevier. Retrieved May 17, 2021, from https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/

Skalski, P. (2019, April 12). Gentle Dive into Math Behind Convolutional Neural Networks. *Towards Data Scienc*.

Statistics Libre Texts . (2020, August 17). K nearest neighbors. *RTG: Classification Methods*.

Stecanella, B. (2017, June 22). An Introduction to Support Vector Machines (SVM). *Monkey Learn Blog*.

TensorFlow. (2015). *TensorFlow Core*. Retrieved from Transfer learning & fine-tuning: https://www.tensorflow.org/guide/keras/transfer_learning, accessed 31st May 2023.

Unruh, A. (2017, November 9). What is the TensorFlow machine intelligence platform? *Red Hat & Open Source* .

Valkov, V. (2017, May 1). Building a Simple Neural Network — TensorFlow for Hackers (Part II) . *Adventures in Artificial Intelligence*.

Varmedja, D., Karanovic, M., Sladojevic, S., Arsenovic, M., & Anderla, A. (2019). Credit Card Fraud Detection - Machine Learning methods. *18th International Symposium INFOTEH-JAHORINA (INFOTEH).* East Sarajevo, Bosnia and Herzegovina: IEEE Xplore.

Wang, Y., Qin, J., & Wang, W. (2017). Efficient Approximate Entity Matching Using Jaro-Winkler Distance. *International Conference on Web Information Systems Engineering.* Springer, Cham.

Xu, J., Wang, G. A., & Li, J. &. (2007). Complex Problem Solving: Identity Matching Based on Social Contextual Information. *Journal of the Association for Information Systems, 8*(10), 525-545.

Yadav, S., Adwitiya, S., & Kumar, P. (2019). Multi-attribute identity resolution for online social network. *SN Applied Sciences , November*(21).

Zhang, C. (2018, September 9). Quick Notes on How to choose Optimizer In Keras. *A Tour to Machine Learning and Deep Learning*.

Zhang, L., Ma, K., Yuan, F., & Fang, W. (2022). A tabnet based Card Fraud detection Algorithm with Feature Engineering. Guangzhou: IEEE.