Predicting Likelihood of Legitimate Data Loss in Email DLP

Mohamed Falah Faiz[a], Junaid Arshad[a], Mamoun Alazab[b], Andrii Shalaginov[c]

[a]*School of Computing and Engineering, University of West London, London, UK*
[b]*College of Engineering, IT Environment, Charles Darwin University, Australia*
[c]*Department of Information Security and Communication Technology, Norwegian University of Science and Technology, Norway*

Abstract

The volume and variety of data collected for modern organizations has increased significantly over the last decade necessitating the detection and prevention of disclosure of sensitive data. Data loss prevention is an embedded process used to protect against disclosure of sensitive data to external uncontrolled environments. A typical Data Loss Prevention (DLP) system uses custom policies to identify and prevent accidental and malicious data leakage producing large number of security alerts including significant volume of false positives. Consequently, identifying legitimate data loss can be very challenging as each incident comprises of different characteristics often requiring extensive intervention by a domain expert to review alerts individually. This limits the ability to detect data loss alerts in real-time making organisations vulnerable to financial and reputational damages. The aim of this research is to strengthen data loss detection capabilities of a DLP system by implementing a machine learning model to predict the likelihood of legitimate data loss. We conducted extensive experimentation using Decision Tree and Random Forest algorithms with historical email incident data collected by a globally established telecommunication enterprise. The final model produced with Random Forest algorithm was identified as the most effective as it was successfully able to predict approximately 95% data loss incidents accurately with an average true positive value of 90%. Furthermore, the proposed solution successfully enables identification of legitimate data loss in email DLP whilst facilitating prioritisation of real data loss through human-understandable explanation of the decision thereby improving the efficiency of the process.

Keywords: Data Loss Prevention, Email DLP, Insider Threats, Threat Prediction, Machine Learning

1. Introduction

The volume and variety of data collected for modern organizations has increased significantly over the last decade. The advancements in and pervasive use of digital technologies have a profound role in it [1] resulting in continuous extraordinary growth in the volume of data reaching up to 175 ZByte by 2025 as predicted by International Data Corporation (IDC) in [2]. Furthermore, the risk to information security in the current age has raised manifold with significant increase in incidents of data breach. A recent study by Accenture [3] has revealed that the data breaches have increased by nearly two-fold in the five years since 2012, increasing from 68 in 2012 to 130 in 2017. Typically, such incidents involving large organizations often result in leaking customer personal and financial data. For instance, recent data breach incident reported by British Airways resulted in leakage of 380,000 customer personal and financial data [4]. Furthermore, the volume and variety of attacks on organizations have become increasingly sophisticated as reported by European Agency for Network and Information Security (ENISA) in [5] making protection against them a significant challenge. Moreover, recent introduction of General Data Protection Regulation (GDPR) [6] by the European Union has increased emphasis on protection of security and privacy in general and for the personal data in particular especially due to significant financial penalties for non-compliance.

Within this context, Data Loss Prevention (DLP) has a profound role in achieving defence in depth where it can be considered as a layer of defence system to achieve security of data at-rest, data in-motion, and data-in-use. Typically, a DLP system represents a set of tools and embedded processes used to ensure that sensitive data is not lost, misused, or accessed by unauthorized users [7]. A DLP system achieves this by providing an in-depth insight into data usage (access) and transportation (sharing) within an organization. With the significant increase in the cyber-attacks targeting data theft, use of systems such as DLP has also increased with a recent study [8] indicating 62% of organizations adopting such solutions as part of their organizational security architecture.

Although a DLP system is effective in providing an in-depth insight into how a monitored system is being used, it can also increase the complexity of the threat detection task. For instance, a DLP system can produce large number of periodic alerts based on an organization's security policies which makes prioritising DLP incidents extremely challenging task. Furthermore, in its default implementation, DLP serves as an auditing system which is aimed at collecting user interactions with data in an offline manner.

The extraordinary increase in the volume and variety of attacks as well as monitoring capabilities of DLP systems introduce novel opportunities to achieve real-time, intelligent analytics to aid protection against data theft. However, achieving an effective solution to these challenges requires intelligent data processing abilities to address the scale and complexity of the alert data. In this context, machine learning has been historically used to address complex challenges across different domains including intrusion detection and response [9, 10, 11, 12, 13], healthcare [14] and business intelligence [15]. The focus of our research is to utilize machine learning techniques to enhance the capability of a typical DLP system to facilitate proactive defence against insider threats. In particular, we aim to assess the feasibility of our proposed approach to predict likelihood of data loss based on historical DLP data. We conducted rigorous experimentation and analysis using real-life data provided by a large UK telecommunication provider with the view to assess use of our approach in real world systems. Additionally, we envisage evaluating whether historic alert information such as (when? how? and what was leaked) can be used to identify legitimate data loss in advance as well as help assess real data loss incidents.

Therefore, we make the following contributions:

- We present a novel machine learning-based approach to use the incident data produced by a DLP system to enhance protection against data theft. We specifically focus at the Email DLP with data collected from a live installation of DLP system within a large telecommunication provider.
- We conducted an in-depth evaluation of using machine learning techniques to achieve effective protection against data theft and insider threats. In particular, we use Decision Trees and Random Forest algorithms with varied experimental settings to achieve a rigorous analysis of their feasibility to address the challenge of data loss prediction.
- Based on the outcomes of our experimentation and analysis of the DLP data under study, we develop an intelligent machine learning model that is able to predict legitimate data Loss incidents in Email DLP with approximately 95% accuracy and an average true positive rate of 90%.

Rest of the paper is organized as follows: Section 2 presents a background knowledge about DLP systems, their different types as well as examples of existing DLP systems. Section 3 presents a critical overview of existing work related to this research identifying gap which is addressed by this paper. This is followed by a detailed description of the data used in this research in section 4 along with consideration with respect to feature extraction and data quality. Section 5 includes a detailed account of experimentation methodology as well as different experiments performed with the chosen machine learning techniques followed by an in-depth analysis of the results in section 6. Section 7 concludes the paper.

2. Data Loss Prevention

Due to the extraordinary influx in the data generation capabilities of modern computing systems, data has become one of the most important assets for all types of organisations including Small and Medium-sized Enterprises (SME) to large organisations. Safeguarding this data can be extremely challenging as organisations are moving towards a digital culture and therefore issues such as accessibility and policy enforcement become non-trivial. However, leakage of sensitive data can cause enormous damage to an organisation, in particular, it could cause financial damage in terms of major fines as governed by the GDPR as well as reputational damage or damage to organisational growth. In this context, Data Loss Prevention or Data Leakage Prevention (DLP) is a system with well-constructed processes which is aimed at protecting against disclosure of sensitive data to outside the organisation. A DLP system can be visualised as a set of tools with embedded processes used to ensure sensitive data is not lost, misused, or accessed by unauthorized users [7]. Such systems are often setup with range of complex information security policies to meet an organisation's baseline security requirements. A DLP system preforms a deep content/context analysis aiming to identify keywords (classified attributes) such as financial data (e.g. credit card number), intellectual property (IP), legal data, source code, personally identifiable information (PII) and much more [16, 17].

2.1. Data loss detection methods

The primary goal of a DLP solution is to detect accidental and/or malicious data leakage of corporate data and notify security administrator immediately in order to manage the data loss risk in a timely manner. Based on the knowledge extracted from the DLP incident alerts, an organization can implement preventative measures by refining DLP security policies to avoid future occurrence of similar incidents. As the scope of data loss prevention is broad, a number of detection methods are commonly used by DLP systems such as context-based, content-based, and content tagging [18, 16]. Context-based data inspection relies on analyzing contextual information such as source, destination, size, or recipient of data item in question, while content-based data inspection uses techniques such as pattern or regular expression matching, and text analysis. Content tagging assigns tags to sensitive data items that form basis for this detection method. In this respect, there are two major approaches in developing model for the underlying DLP solution as outlined in [18, 19] i.e. specification and learning based approaches. Both these approaches are independent of the data endpoint on which they are applied as well as the detection method.

Understanding the scope of DLP is important as this can help understand the impact of research presented in this paper. In this context, Salem et al. [20] presented a commonly adapted taxonomy for DLP system which categorizes them based on the scope of protection or type of data as explained below as well as presented in Fig 1.

- *Data in use* refers to data that user presently interacts via network endpoint device (laptop, desktop computer, and iPad/tablet) in the form of text, documents and applications. A DLP system operating at *data in use* level, monitors and alerts unwanted activities which violate organisation information security policies. Such activities can include copy/paste/transfer, external uploads, print and screen-capture operations.

- *Data in motion or transit* refers to data travelling through a computer network from one node to another node. Data can travel either through internal (private) network or external (public) network such as a workstation residing outside controlled environment. There are three common types of transfer of data i.e. Public networks, Private networks and Local devices. A typical example can be where an email with a sensitive information sent to an unintended recipient is considered as affirmative data loss and most of the lost data cannot be recovered.

- *Data at rest* refers to data that resides within persistent storage units and other repositories such as hard drives, databases, cloud storage, SharePoint, file servers and local network drives. A DLP system can monitor and prevent unwanted behaviour in a proactive fashion while this state has been considered as more secure than data-in-motion as data does not always leave the controlled environment.

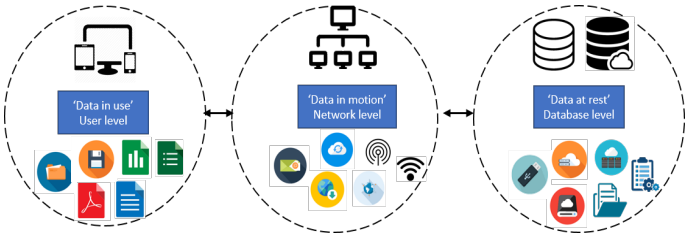


Figure 1: Different scopes of data loss prevention systems

With regards to the different types of DLP presented above, each protection scope can be implemented individually as well as being complementary to each other. However, a DLP system with all three protection scopes will enable the organisation to achieve a layered defence-in-depth. Furthermore, the research presented in this paper is focused at *Data in motion or transit* with particular emphasis on the data loss through sent emails.

3. Related Works

The research presented in this paper is related to data loss prevention, threat detection and intelligent classification. We present a summary of related research from these domains below.

Machine Learning (ML) and Advanced Data Analytics have been historically used to successfully address variety of cyber security challenges and have proven ability to train from real-world data, handle impression and solve problems such as classification, clustering and prediction. Effectiveness of ML techniques in this domain has led to the development of specialist systems such as DLP. For instance, Symantec has been looking into DLP through ML-assisted processing already since [21]. It was described that the conventional ways of looking for such data traces are either *describing* through matching known keywords or *fingerprinting* through matching parts of files. However, those technologies are incapable of handling challenges related to Big Data paradigm such as data veracity and variety [22]. Therefore, the idea is to utilize training and testing phases in machine learning to be able to detect new data loss events through intelligent models based on historical data. Another product available to end users is DLP solution by Forcepoint that offers ability to classify previously unseen data as well as train model through flagging documents and files [23]. Finally, Jaiswal et al. [24] patented a detection solution based on the machine learning through feature extraction and binary classification model training based on the corresponding documents from both classes.

Wu et al. [25] proposed a DLP solution based on user keystroke profiling to address an existing problem (file format issue) within DLP systems. In particular, a typical DLP system scans various files with different formats. However, any unsupported file types will be left unscanned leading to undesired risk. Authors used Support Vector Machine (SVM) to build keystroke profiles measuring the cost of character types switching time and typing frequency of each user. In addition, the proposed solution identified creator of the content enabling incident reviewers to make decisions based on the file origin. Although the information about the content originator can be very useful when assessing an incident, the proposed solution did not support existing DLP systems therefore limiting its interoperability with existing systems.

Carvalho et al. [26] represents one of the earliest approaches to data loss and proposed a DLP solution using K-Nearest Neighbours (KNN) algorithm and textual content method known as Cosine method. The proposed solution addressed data loss caused through corporate email aiming to predict accidental data loss based on recipient email addresses This study is particularly interesting as the focus is similar to our research. The study used an assumption considering unknown email recipient address as an anomaly. The study has used semi-synthetic data along with social network data which is about 1,100 sim-

ulated incidents from 20 users. They produced scoring for each labelled data where the recipient with the lowest predicted score considered as a data leak (outlier). Although the proposed model achieved high accuracy, i.e. 82%, however the process was reactive as it attempted to identify unknown emails recipients based on pre-defined heuristics rather than preventing a data loss. A major drawback of this approach is that it does not take into account emails sent to a receiver without any previous interactions leading to increase in false positive rate.

Costante et al. [19] proposed a DLP solution based on hybrid approach which attempts to detect potential data leakage by spotting anomalies in database transactions. The authors used a white-box anomaly-based approach (minimizing time required to map a rule with a new alert) to detect unseen transactions while a rule-based engine is used to prevent any activities that are previously identified as malicious. This work was later extended in [27] where the authors used histogram based profiling of database usage enhanced with aggregated feature variables and transaction flow analysis facilitating detection of complex threats. The study used real-world dataset containing 12,040,910 records reporting high detection rate with a 0.003% false positive rate.

Kyrre W. et al. [28] used Bayesian networks (BN) to implement a scoring based approach to indicate potential data leakage through insider threat. The threat scores were based on number of known insider threat characteristics and aggregated data generated by existing DLP systems. The threat score is then compared with classification set by the DLP system and the indication of insider threat is derived for incidents where a misclassification or no classification is identified. The study attempted to identify potential manipulation attacks by feature vectors of the score summary statistics in order to allow each user to be compared with past behaviour as well as with the behaviour of team. The approach produced a high false positives rate when detecting anomalies, especially when the number of feature variables is high.

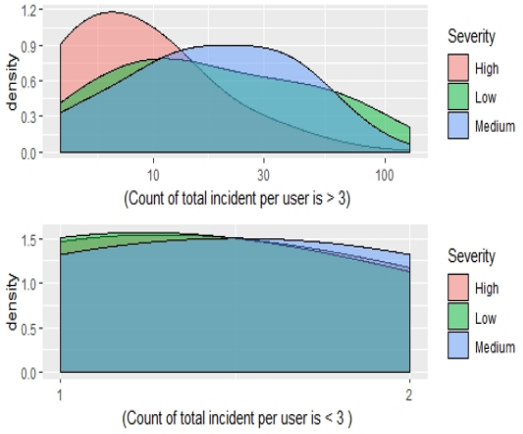
Vukovic et al. [18] presented another approach for data loss prevention based on a rule engine and threat estimation. The authors presented the pioneer effort focused at supporting an existing DLP system that is already in production environment. The proposed approach uses risk assessment via metrics such as destination of the file, count of time since last destination was evaluated, and number of detected suspicious messages for the given recipient, etc. The study used a real-world case-study to demonstrate how proposed approach may work for different breach scenarios however the approach can be improved by implementing it with a machine learning algorithm.

In addition to the above, a number of efforts have been made to complement DLP systems in achieving detection of threats in general and insider threats in particular. For instance, Kim et al. [29] proposed an anomaly-based system to identify targeted cyber attack known as Advanced Persistent Threat (APT). Although the approach also overlaps the domain of intrusion detection, it is considered relevant due to its use of behavioural data from sources such as network, host, security equipment and devices. The study used Hadoop framework and Map Reduce to implement the solution where it counted the behaviour features of each process and returned the feature description as a results. Similarly, Kandias et al. [30] presented a study focused on insider threats where the main aim is to rank users based on various factors and then used the score to predict vulnerable users who may potentially be dangerous to the organisation as well as the IT systems. The authors used data from various Information Systems such as real-time data from an Intrusion Detection System (IDS), User taxonomy, Psychological Profiling (applied social engineering theory) and data gathered from honeypot technique.

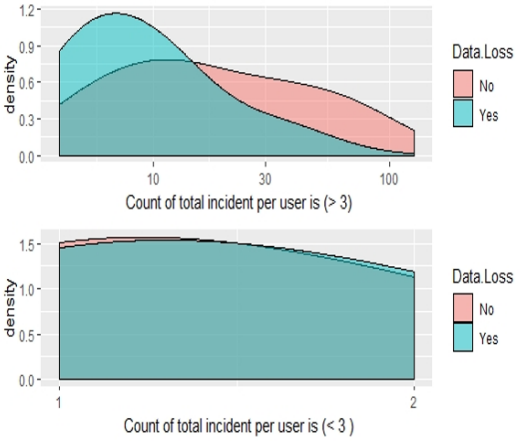
In summary, current research and methodological approaches are mostly focused at identification or prevention of data loss whereas limited efforts have been made to supplement existing DLP systems to facilitate intelligent use of visibility (data loss alerts) provided by such systems. This is significant because a DLP system often triggers high number of alerts based on complex policies making it non-trivial for an incident reviewer to analyze each alert individually for data loss. Existing solutions do not provide support in identifying and prioritising events that comprises real data loss as identified by [**?** 18]. Therefore there exists a gap in literature to supplement a DLP framework to achieve intelligent threat estimation to mitigate potential threats in a timely manner. Within this context, this research aims to address this gap by proposing an intelligent mechanism to predict and prioritise genuine data loss enabling an organisation to (i) be alert of particular users groups who are vulnerable (ii) implement preventative measures in advance such as content or context blocking, and (iii) achieve proactive security communications ahead of threat occurrence.

4. Data Collection and Preparation

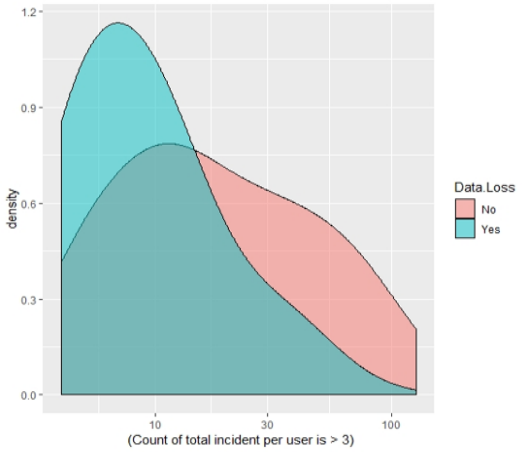
The focus of this research is to achieve an intelligent analytics of alerts generated by a DLP system to support likelihood prediction of the legitimate data loss through email. In order to achieve applicability and effectiveness of the outcomes of this research, we used data from a live DLP system installation containing historical DLP email incidents data from a leading UK telecommunication company. The dataset was generated over a period of three months from September 2018 and contained 8,117 records for 1,419 unique users. In order to comply with organisational policies and UK/EU regulations such as the GDPR and the UK Data Protection Act 2018 [31] set out to protect individual's privacy, the dataset was fully-anonymised. In addition, the dataset was semi-synthesised in order to achieve confidentiality of the results produced in this project. A detailed description of the different features within the data set is presented in the Table 1.



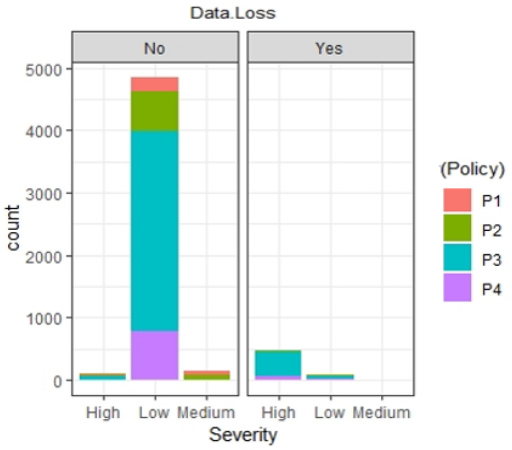
(a) alert distribution w.r.t. severity



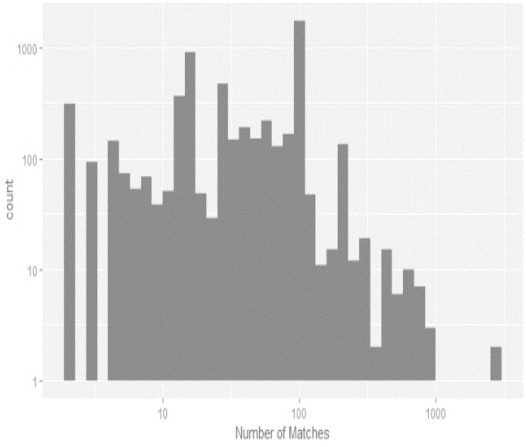
(b) alert distribution w.r.t. users



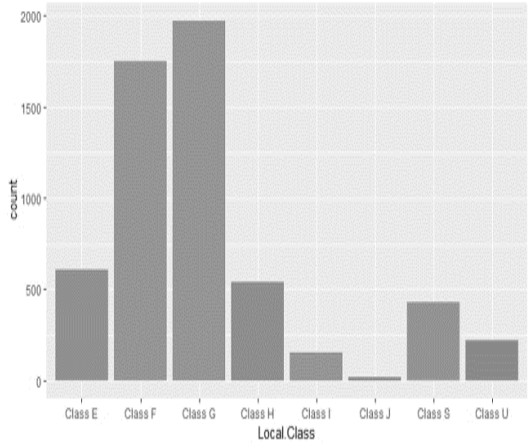
(c) Distribution of data loss occurrences



(d) Distribution of alerts by severity



(e) Number of keyword matches for each incident



(f) Distribution of data loss alerts by employee class

Figure 2: Characteristics of data used in the experimentation

Furthermore, an analysis of different characteristics of the data such as alert distribution with respect to incident severity, users, employee class (organizational roles) and number of incidents per user is presented in Fig 2. The data used for these experimentation contains alerts which represent different user profiles as well as an even distribution of different severity levels. For instance, as illustrated by Fig 2.b, the alerts captured by the data present an even coverage of users within the system with significant majority of incidents representing users where number of incidents per user is less than 3. Furthermore, as presented in Fig 2.a, the severity of incidents captured in the data

No.	Feature Name	Data type	Incident life-cycle	Characteristic type	Usability	Description
1	DummyID	integer	pre-investigation	User	No	Unique user
2	Type	Char	pre-investigation	Incidnet	Yes	Incident type; email
3	Severity	Char	Pre-investigation	Incident	Yes	Incident Severity e.g. (High, Medium and Low)
4	Sent	Date	Pre-investigation	Incident	Yes	Incident occurred date
5	Policy	Char	Pre-investigation	Incident	Yes	DLP policy indicates type of data e.g. (personal data policy /Intellectual property policy)
6	Matches	Integer	Pre-investigation	Incident	Yes	Keyword count in the email
7	Status	Char	Both	Incident	No	State of the incident (open/closed)
8	Has Attachment	Boolean	Pre-investigation	Incident	Yes	if email contained an attachment
9	Subject	Char	Pre-investigation	Incident	Yes	Subject of the email (RE/FW/New)
10	Recipient(s)	Char	Pre-investigation	Incident	Yes	number of recipients. (internal/external)
11	Department	Char	Pre-investigation	User	Yes	User department
12	ID	Integer	Pre-investigation	Incident	No	Incident ID generated by the system
13	Message status	Char	Pre-investigation	Incident	No	Contains single value for all the incidents
14	User Contacted	Char	Post-Investigation	Incident	No	if user have been contacted before (Yes/No)
15	Data Loss	Boolean	Post-Investigation	Incident	Yes	whether the incident is identified as data loss or not (Yes/No)
16	Classification	Char	Post-Investigation	Incident	No	post investigation classification of the file (S1,S2,S3,S4)
17	Tuning Required	Char	Post-Investigation	Incident	No	if the incident needs tuning due to Falsely captured by the system
18	Outcome	Char	Post-Investigation	Incident	No	incident outcome; training/malicious/non-breach

Table 1: DLP email feature descriptions

are distributed across different levels and therefore data is not skewed towards incidents with specific threat level and facilitates good coverage.

Within the data collected for this research, every incident has a life cycle and can have a status such as *new, triaged, investigation, false positive* and *closed*. Each status can have number of incidents, however, in order to select relevant data for this research, different variables are grouped into two classes such as *pre-investigation* (an incident not assessed by a security reviewer) and *post-investigation* (fully assessed incidents) as also highlighted in the Table 1. Any incident that is partially assessed would have the status set to *triaged* or *investigation*.

4.1. Feature Extraction and Engineering

As part of data preparation process, a number of concerns were identified with potentially significant impact to the overall experimentation and consequently the outcomes. These are detailed below.

- *Post-investigation* variables are not usable due to their life cycle status, and therefore, only pre-investigation variables can be used to train the model.
- The *Data Loss* feature represents the outcome of investigation for an incident with values *Yes* or *No* and is populated through domain knowledge and risk assessment by an experienced administrator. Therefore, any incident with the status *New, False Positive* and *Triaged* are not used to train the model as they do not have value for the *Data Loss* field.
- Some features have inconsistent or unstructured values. For instance, *Subject* is classified as usable feature in the Table 1 however, each subject line is unique and have different values. Omitting these feature makes no data for prediction. In addition, a variable with high number of unique values may not be handled by certain algorithm. For instance, Random Forest algorithm does not handle more than 53 unique categories. This can affect the performance of the overall model and has been addressed as part of feature engineering process in the next section.
- Majority of the *pre-investigation* variables are categorical therefore replacing missing values with a mathematical function such as mean value of the column does not apply to this problem. Some values were unrecoverable, therefore carefully omitted while other problems were addressed later in the experiments by adapting different methods such as transforming features, scaling, factorising and aggregating new features as discussed in section 4.1.

In view of the above and to achieve quality and rigor of the experimentation process, a feature engineering process was conducted. Feature engineering task is a vital initial

data pre-processing task that can help avoid bias and lack of generalization of the model. In order to understand the properties of the collected data, an Exploratory Data Analytics (EDA) [32] exercise was performed. It was discovered that although features such as *subject, department, sent date, recipients* could be significant to the detection model, data for these features in the original dataset are not compatible with the machine learning algorithms used and therefore required transformation. In this regard, the following four features are transformed using existing data to align these to the requirements of the machine learning algorithms used.

Feature 01 - Subject: The value of the feature *Subject* in the original dataset contains textual information. Although the complete subject line can be used as part of an interesting text analysis approach for future, however, for this research, we classify the subject into three categories i.e. *Forward, Reply* and *New* as follows:

- *Forward:*: Incidents with subject line containing word *FW:* is considered as a forwarded message, therefore all the characters are replaced with the word âĂIJForwardâĂİ.
- *Reply*: Incident with subject line containing word *RE:* is considered as a replied message, therefore all the text is replaced with the word âĂIJReplyâĂİ.
- *New*: Any incident with none of the above characters is considered as a new email composed by an internal user.

Feature 02 - Department Similar to the above, a scoring-based approach was developed for the *Department* variable resulting in 3 qualitative variables as explained below.

- *Avg_ inc_ per_ dpt_ total*: number of users involved in a data loss incident for each department divided by the total number of incidents
- *By_ dpt_ Median*: number of incidents per user in each department including non-data loss incidents and divided by the median of the department
- *Avg_ of_ dpt_ total*: incidents per users in each department divided by the average of the total incidents of all departments

The above variables provide an in-depth insight into the linkage between data loss incidents and users from specific organizational units of an organization.

Feature 03 - Recipients: Currently this variable contains list of recipients such as internal, trusted external or personal/non-corporate. Therefore, this variable is transformed into an acceptable recipient score based on number of internal recipients in an email as explained below.

- Total number of internal recipients in an incident (x) / total number of recipients (y)
- Acceptable Recipients Score (ARS) = number of internal recipients (x) / (y) total number of recipients

Feature 04 - Sent: The *Sent* feature describes when the incident has occurred in a date format. Although this is a useful feature, however, this research does not intend to perform any time series functions. Therefore, the time/data variable is transformed into qualitative variable. For example, 04/07/2018 transformed into [7, 18].

In addition to the mentioned above, features 3 *(DuringOfficeHrs)* and 5 *(HasInternalReci)* in the Table 2 have been transformed in the similar way from the DLP Email dataset while feature 1 *(Localclass)*, feature 2 *(Employment.status)* and feature 4 *(AppAccess)* are generated by a dedicated IT system available in the organisation.

5. Experimental Design

Since the aim of this research is to predict data loss for DLP incidents, this problem can be envisaged as a classification challenge. Within this context, we conducted rigorous experimentation using two different approaches where the first set of experiments were conducted using decision tree Classification and Regression (CART) [33] algorithm and the second experiments were based on ensemble tree (Random Forest) algorithm [34]. The choice of machine learning techniques is motivated by the ability of chosen techniques to produce not only the classification outcome but also reasoning for each classification facilitating deeper analysis and review by a domain expert. Furthermore, these machine learning techniques have been used successfully to address classification challenges within security domain such as intrusion detection [10, 9] and spam call identification [35] as well as in wider application domains such as healthcare [14] and business intelligence [15]. The experimentation has been conducted using SQL language format to fetch data from different databases and warehouses, while Python libraries are used to perform data wrangling and feature engineering tasks. The machine learning models were implemented using the popular data analytics framework R [36] on a standard computing machine with a processing power of 2.4GHz Intel i5 processor with a 8 GB memory running on Microsoft windows 10 operating system.

Each experiment consisted of number of iterations simulating different scenarios to identify the performance and efficiency of the classification algorithms. The output is produced in a form of confusion matrices which were used to evaluate the model at the end of each iteration. The optimal model is achieved when the classifier predicts data loss in terms of true positive rate above 85% while false positive rate should be less than 20% to be accepted. In this context, high accuracy with *p-value* less than 0.05 means that the results of the model is of high significance

No.	Feature Name	Data type	Incident lifecycle	Characteristic type	Usability	Description
1	Local.class	Char	Pre/Post-Investigation	User	Yes	class of the particular user Class (E, F, G, H, S & U)
2	Employment.status	Char	Pre/Post-Investigation	User	Yes	user's employment status
3	DuringOfficeHrs	Boolean	Pre/Post-Investigation	Incident	Yes	whether the incident occurred between 8am & 6pm (Yes/No)
4	AppAccess	Char	Pre/Post-Investigation	User	Yes	type of access to the network (Basic/Full)
5	Has_Internal_Reci	Boolean	Pre- Investigation	Incident	Yes	if incident has internal recipients

Table 2: Additional features from organizational systems

and the null hypothesis can be rejected. On the contrary, for any model with *p-value* greater than 0.05, the null hypothesis will be accepted as this would indicate weak evidence against the null hypothesis.

5.1. Experiments using Decision Trees

The experiments with the decision tree were conducted using a dataset size of 8,117 incidents with 21 variables of which 7,265 of the incidents are non-data loss while 852 incidents containing data loss incidents. The general characteristics of this data is presented in Figure 2 and details of different variables have been presented in Table 1 and 2.

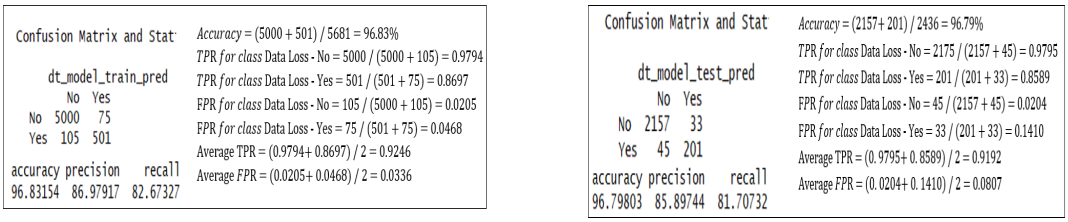
For the first set of experiments, decision tree algorithm was used to conduct analysis across different settings based on varying distribution of training and test data i.e [7:3], [6:4] and [6:4] with pruning where [x:y] represents (x% training and y% test data). We present sample results from these experiments with detailed discussion in this section later on. The experiments were conducted using the Recursive Partitioning And Regression Trees (RPART) library which employs the Classification and Regression (CART) algorithm. The model executed with standard parameter settings on both training and testing data set. The output has been presented in the confusion matrix in the Figure 3. In particular, the Figure 3.a shows the confusion matrix for the training data where the model had an accuracy of 96.83% while a recall of 86.97%. The number of actual positives labelled correctly and the precision of 82.67% demonstrates accuracy of the model from those predicted positive. Additionally, the p-value is 0.03065 which demonstrates significance of the model. Furthermore, the Figure 3.b presents the confusion matrix for experiments using previously unseen randomly selected test data. The model produced an accuracy of 96.79% which reflects high accuracy, generalization and effectiveness of the algorithm in analysing previously unseen data. Furthermore, the recall and precision rate also decreased for these experiments strengthening the overall efficiency of the approach. However, it is evident that the hypothesis we are testing can be accepted as the p-value has decreased to 0.0212 which is less than the threshold.

The Figure 4 shows the tree diagram produced from the above experiment using CART decision tree algorithm. The output seems quite interesting as the branches used in the tree indicate high relevance to the problem. The branches seem very informative as the algorithm identified number of nodes that upon manual analysis by domain experts are found to be relevant. The primary nodes associated with a data loss used in the above tree along with brief analysis of the output is presented below.

1. When *Severity* of an incident is Low or Medium, the probability of an incident being a non-data loss is 89% which inversely means there is 11% likelihood of data loss based on 100% observation cover.
2. When the *Severity* is High and the Policy is either P1 or P2 (representing information security policies), the probability of data loss is 81% based 11% actual observation cover. However, there is 19% likelihood of the incident being a non-data loss incident.
3. When the *Severity* of an incident High & the Policy is either P3 or P4, the probability of data loss is 88% based on 9% actual observation cover. However, there is 12% likelihood of the incident being a non-data loss incident.
4. When the *Severity* of an incident High & the Policy is either P1 or P2 and the count of total incidents per user is greater than or equal to 10, the probability of data loss is 37%. In addition, if the count of number keyword Matches is greater than 191, the probability of data loss increases to 55% based on 2% observation cover. The inversion of this rule is that there can be 45% likelihood of non-data loss incident.

The output is very promising as the association within selected variables makes logical sense and the selected features seem to have high significance to the hypothesis tested in this empirical study. This is a desirable result, however different sample size can affect the above result, therefore an attempt of resampling the dataset was made with the [6:4] ratio where 60% dataset was reserved for training set and 40% for testing set.

Following the results from the above experimentation, two further iterations of experiments were conducted using decision tree with different settings for training and test data sets. In particular, iterations I and II had standard parameter settings of the decision tree algorithm with two different sample sizes [7:3] and [6:4] for the training and testing data. However, the iteration III was improved based on outcomes of iteration II where the model produced an accuracy of 97.20%, precision of 87.65% and the recall of 84.77%, which demonstrates effectiveness of the



(a) Training (b) Test

Figure 3: Confusion matrix for A) training set and, B) test data

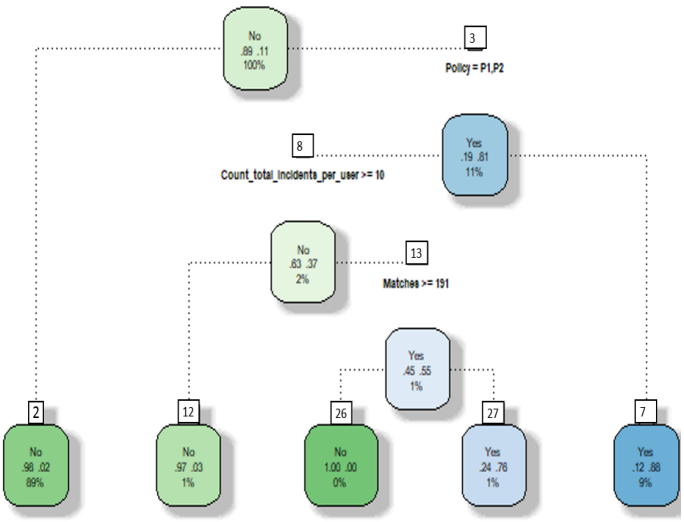


Figure 4: Sample tree for decision tree based experiments

approach. The Figure 5 shows the average True Positive Rate (TPR) and the average False Positive Rate (FPR) for all the three iterations conducted using decision trees. It appears the classifier from the iteration III produces equal True Positive rate as the classifier in iteration II however, the accuracy and the average False Positive rate of the final model iteration III is significantly higher than the other iterations. A detailed evaluation of these results is presented in the Section 6.

5.2. Experimentation with Random Forest

In order to assess the feasibility of using machine learning techniques to predict likelihood of insider threats, we also conducted experiments using Random Forest algorithm so as to achieve rigorous comparative analysis. As Random Forest is an ensemble algorithm, a common philosophy is that a prediction made by an ensemble model is far better than a stand-alone method such as single decision tree [37]. Ensemble model is typically envisaged to achieve better accuracy (mean value) to the actual by using sophisticated techniques such as bagging or hybrid intelligence [38]. Unlike decision tree algorithm, in random forest, it is difficult to trace through the tree node and understand the reason for the prediction therefore, the trees produced are not always methodical to explain in all aspects [39]. Within our experiments, random forest algorithm is used with the default parameter setting where problem type is classification, number of trees set to 500 and the number of variables tried at each split is set to 4.

5.2.1. Sensitivity Analysis

As discussed before, one of the primary reasons for lower prediction accuracy is that as there can be a proportion of errors on the training and the test data. Consequently, an attempt was made to identify features which are most significant to achieving prediction. This is achieved in two ways: a) using the *Importance()* function and b) using *Recursive Feature Elimination (RFE)* where RFE selects high relevance variables for the model removing the weakest features.

Furthermore, the Figure 6 shows the importance of each variable ordered by most - to least-importance. The Mean Decrease Accuracy (MDA) shows the variables that are most relevant to model accuracy where any variable with a large (MDA) is more important. Mean Decrees Gini (MDG) shows how each variable contributes to the homogeneity of the nodes in the model. It appears, the variable *Severity* has the most (MDA and MDG). This is very concerning as having this variable may bring skewness to the model. Therefore, in order to avoid skewness of the model, our experimentation data excluded the value of severity attribute so as to assess the significance of other attributes towards prediction generated by random forest algorithm.

Furthermore, variables including *Has.Attachment, AppAccess, during_OfficeHrs, Employment.Status* have the lowest (MDA and MDG), indicating their relative insignificance in predicting and splitting data. Therefore, these variables can be omitted from the model. Finally, *by_dpt_Median, avg_of_dpt_total* and *count_of_total_incidents_per_user* have similar mean decrease accuracy. Thus, the variable with low error rate i.e. *by_dpt_Median* can be omitted as this would improve the performance of the model. As stated in section 4.1, removing too many feature variables could result in improper data separation, while having too many variables will over fit the model. Prior to removing variables shortlisted above, a popular feature selection method called RFE (Recursive Feature Elimination) was used. RFE can help identify variables that are primarily required to build an accurate model. RFE uses an outer resampling method (Cross-validation) of 10 folds to evaluate

(a) True Positive

(b) Test

Figure 5: Average True Positive & False Positive rate of Decision Tree analysis

the performance of all the variables except the dependant variable (Data Loss in this case). Figure 7 presents results produced from the 10-fold cross validation accuracy produced as a part of RFE algorithm which demonstrates selected 10 variables to achieve comparable accuracy.

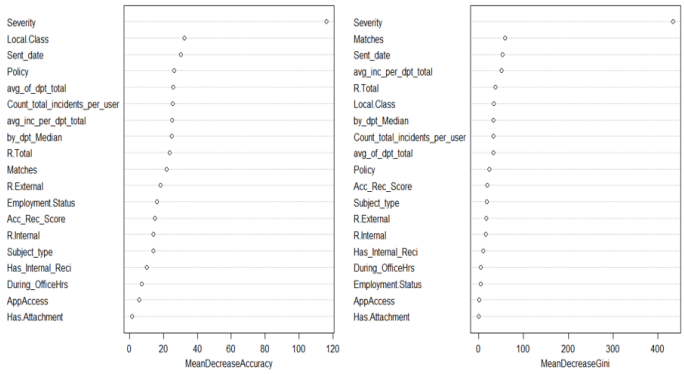


Figure 6: Variable importance by Importance() function

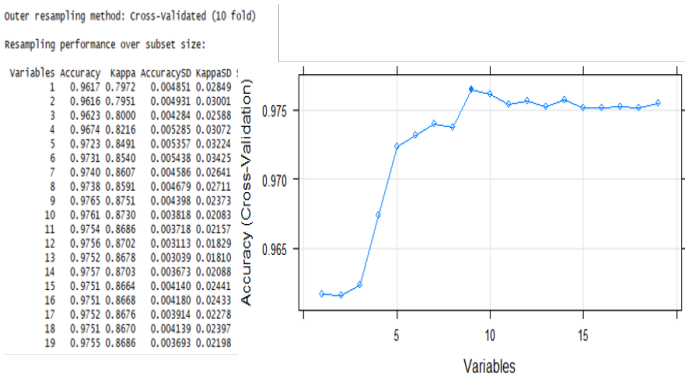


Figure 7: Results produced from the RFE algorithm and variable accuracy

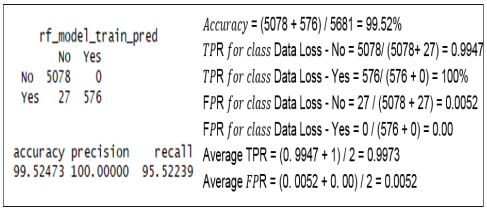
The RFE algorithm selected 9 features including *Severity, Local.Class, Sent_Date, Policy, avg_inc_per_dpt_total, Matches, Count_total_incidents_per_user, R. Total* and *Subject_type*. From this, it is evident that apart from the *Severity* variable, 5 other variables shortlisted as part of decision tree experiment were not selected by RFE.

The Figure 8.a shows the confusion matrix produced for the training set which indicates that the classifier is extremely effective in terms of accuracy, precision and recall. The classifier has an average TPR of 99.73% which is significantly high. Moreover, the model returned a p-value of (0.00000056), indicating that null hypothesis can be rejected.

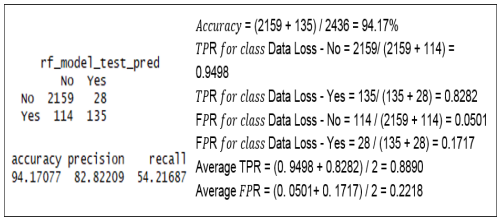
Furthermore, the Figure 8.b shows the confusion matrix for the test data set. The model produced an accuracy of 94.17%, precision of 82.82% and the recall 54.21% which indicates that the accuracy, precision and recall has reduced when compared to the training data. However, the model shows high significance as the p-value was (0.00000008917) for the test data.

The Figure 9 shows a sample tree for analysis using random forest algorithm. The tree is comprehensive when compared with the previous trees mainly due to exclusion of the Severity variable. The primary nodes used in the tree that show data loss occurrence are as follows.

1. When the incident contains recipients and *Sent Date* is either [January, June, July or October], there were about 10% data loss incidents identified. Meanwhile, if the *Sent Date* is [February, March, December], 5% data loss incidents were identified.
2. When the incident with no internal recipients and the *avg_of_dpt_total* is less than or equal to 3.6% and *Subject Type* is either [Forward or Reply], there were about 40% data loss incidents identified. Any incidents with *Subject Type* [New] represents 20% of data loss incidents.
3. Once again, incidents with no internal recipients and the *avg_of_dpt_total* is greater than 3.6% and the *avg_inc_per_dpt_total* greater than 64.3%, there were 100% data loss incidents identified.
4. Furthermore, incidents with no internal recipient and where the *avg_of_dpt_total* is greater than 3.6% and the *avg_inc_per_dpt_total* is less than or equal to 64.3%, and the *Subject Type* is [Forward], there were 20% data loss incidents identified.
5. Finally, when the incident does not contain any internal recipient and the *avg_of_dpt_total* is greater than 3.6% and the *avg_inc_per_dpt_total* is less than or equal to 64.3%, and the *Subject Type* is [New] or [Reply] and the *avg_dpt_total* is less than 16.4%, there were only 10% data loss incidents identified.



(a) Training



(b) Test

Figure 8: Confusion matrix for A) training set and, B) test data

It is evident that above model had good accuracy and precision rate though the recall is low. The current Out of Bag (OOB) estimate error rate is 5.86%. Further improvements can be made to the model in order to obtain better recall. This is achieved by tuning number of parameters e.g. *ntree* – which specifies number of trees to grow and *mtry* – number of variables randomly sampled as candidates at each split in a tree. The *tuneRF* function was used to identify the best *mtry* values. Setting these parameters would reduce the complexity of the overall model and also produces less OOB error. Fig 10 shows the OOB error graph where it is evident that the error rate stabilizes after about 1,400 trees. Therefore, the optimal number of *ntree* can be set to 1,400. Furthermore, to identify the optimal number of *mtry* via tuneRF function is presented in Fig 11 which identified that lowest OOB error rate is achieved when number of *mtry* is 10 and estimate OOB error is 5.11% which is reduced by 0. 26% when compared to the initial model.

Overall, our experimentation with Random Forest algorithm was conducted over four iterations where the iteration 1 and 2 were preformed with standard parameter settings of the Random Forest algorithm. Both these iterations were conducted with two different sample sizes i.e. [6:4] and [7:3] for the training and test data. Even though the first two iterations had a good result in terms of accuracy, precision and recall, both models showed less significance in validating the hypothesis where the p-value is greater than 0.05. However, this problem was addressed in the following iterations iteration 3 and 4 by feature reduction and parameter tuning. The Figure 12 shows the average True Positive Rate and the average False Positive Rate for experiments conducted with Random Forest using previously unseen test data. As is evident from Fig 12, iteration III had an average True Positive rate of 88.90% while it produced an average False Positive rate of 22.18% which indicated a good starting point. However, in the iteration IV, the model appears to have improved as the average True Positive rate raised to 90.79% as well as the average False Positive rate reduced to 18.98%.

6. Analysis and Discussion

The graphs in the Figure 13 shows the summary of accuracy, precision and recall produced for all three iterations using Decision Trees. Based on above figures, it appears the decision tree algorithm maintained a superior accuracy, precision and recall throughout all three iterations. The accuracy and the recall of the models seems to have improved in each iteration. However, the precision in the iteration III had decreased by a fraction. Additionally, as the average False Positive Rate of the final model in the Iteration III was small (7%), it demonstrates the effectiveness of improvements made in the final model.

Similarly, the Figure 14 presents the summary of the accuracy, precision and recall produced for all four iterations using Random Forest algorithm. The first two models seem to have outperformed in both test and training data as they produced the highest accuracy i.e. 97.38% and 97.66% respectively. However, both models were rendered insignificant as the p-value was greater than 0.05. However, improvements such as parameter tuning and feature reduction were made in the following iterations which improved the efficiency of the model. The model produced in iteration III had an accuracy of 94.17%, precision of 82.82% and the recall was 54.22%. The recall seems to have reduced but the model had an average False Positive rate of 22.18%. Furthermore, the model produced an F1 score of 65.53% which can be considered as a good model as this score indicates a balance between recall and precision. In the final model further improvements were made which produced an increased accuracy of 94.95%, as well as increasing precision to 85.20% and the recall of 63.98% which is more stable, as it has almost increased by 10% when compared with the previous version. The final model had an average False Positive rate of 18.98%.

The Figure 15 shows the F1 score (the balance performance measure between precision and the recall) produced for the Random Forest algorithm. It appears the model in iteration III had an F1 Score of 65.53 which seems to be an average score. Meanwhile, the final model in iteration IV had an increased F1 score of 73.09. This score can be considered as ideal score in comparison with the accuracy of the model.

6.1. Analysis of results

The primary aim of this research was to implement a machine learning model to predict likelihood of legitimate data loss occurrences based on historical DLP email incident data. Although, there were number of challenges encountered during each iteration for the chosen machine

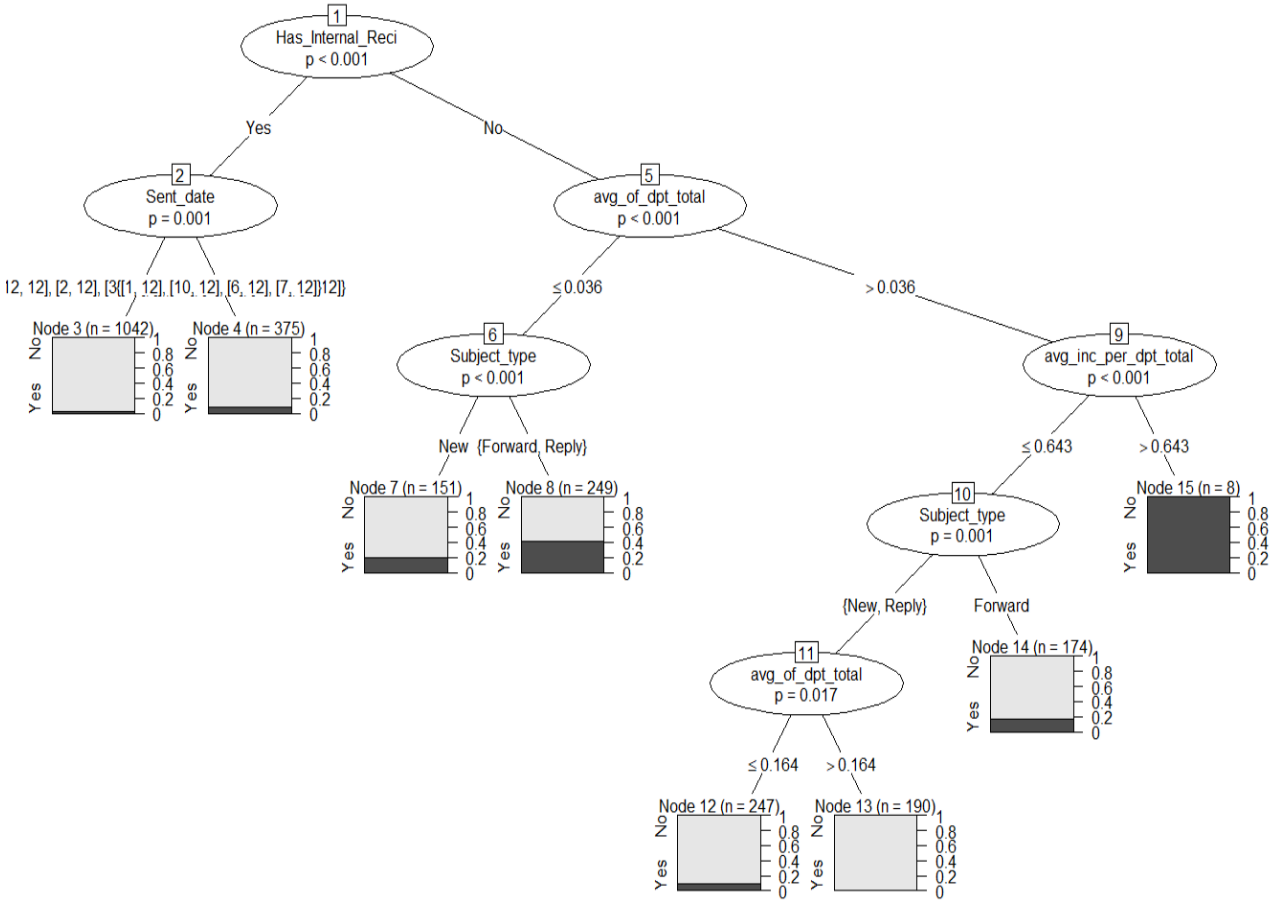


Figure 9: Sample tree for Random Forest experiments

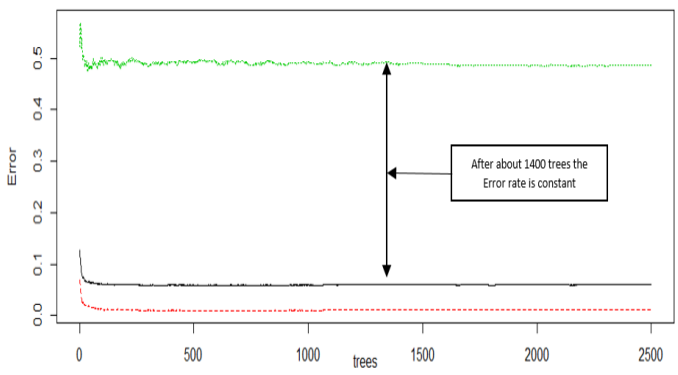


Figure 10: Finding best ntry value - OOB error rate

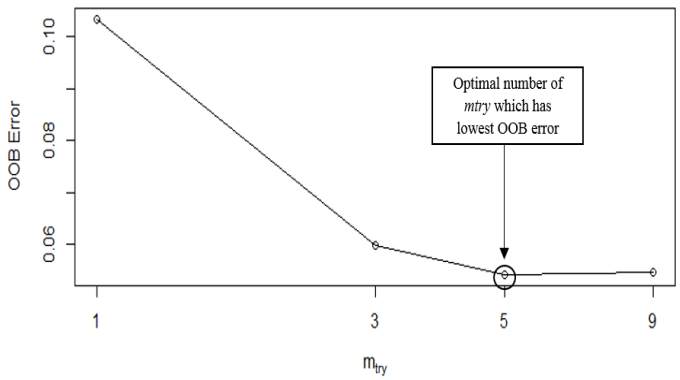
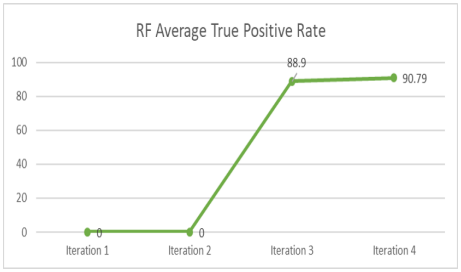


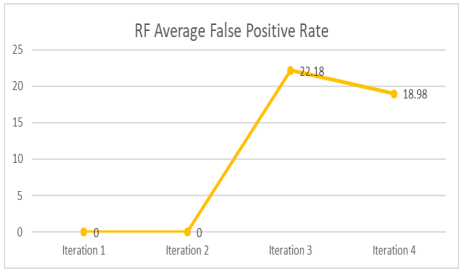
Figure 11: Minimum OOB error: mtry value

learning techniques, it is evident that experiments with both selected machine learning techniques produced positive results. However, the final models produced for each machine learning technique resulted different performance levels in terms of accuracy, precision and recall. A summary of results for both techniques along with different evaluation criteria is presented in the Table 3 with analysis presented below to identify the most effective model.

Due to the complexity of the analysis involved, we set the threshold for satisfactory outcome of experimentation to be an accuracy of minimum 85% with an average True Positive Rate of at least 80% and a F1 Score of at least 70%. Based on the results produced with Decision Tree algorithm presented in section 5.1, it is evident that all three iterations achieved more than 80% in terms of accuracy, precision and recall values. Therefore, a balance F1 performance measure is not required in all three occasions. However, the average F1 Score is 83.50% which meets all of our success factors. In addition, the pruned model produced in iteration III demonstrated much improved performance due to parameter tuning resulting in an economical False Positive rate of 7% which is improved by 1% when

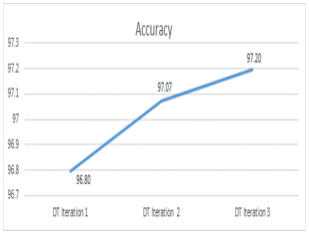


(a) True positive rate

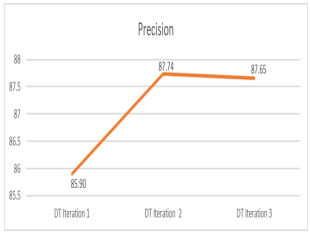


(b) False positive rate

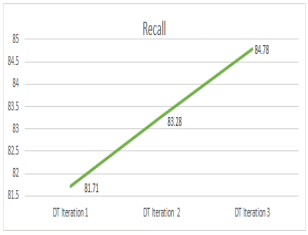
Figure 12: Summary of average true positive and false positive rate for Random Forest



(a) Accuracy



(b) Precision



(c) Recall

Figure 13: A) Accuracy, B) Precision, and C) Recall for Decision Tree experiments

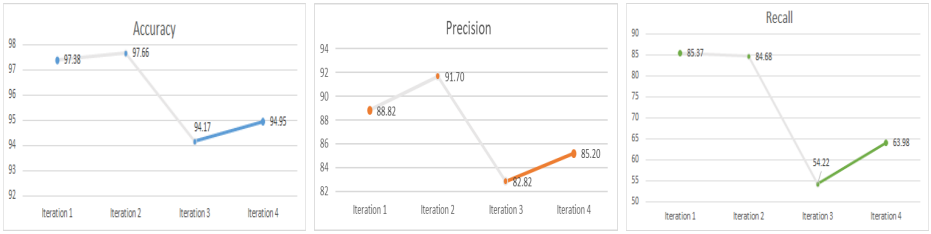
compared with results of iteration II. Finally, although the tree produced in the pruned model looks complete but the nodes and leaves are restricted due the use of *Severity* variable which was revealed as part of the experiments with Random Forest algorithm.

As part of experiments with Random Forest algorithm, the use of variable importance function and RFE algorithm revealed that *Severity* variable had potentially caused the insignificance for iteration I and II. Therefore, in order to address the potential skewness of the data, further experiments were conducted by excluding this variable from sample data which resulted in improved performance by the Random Forest algorithm towards validating the hypothesis and producing good accuracy and precision in the following iterations. However, the F1 score for iteration III was 65.53% which is below the success criteria we defined and therefore this model is not regarded as the optimal.

The improvements made in the iteration IV were effective, especially identifying appropriate number of *mtry* which produced minimal OOB error, along with removing additional variables that were least relevance had made significant improvements in terms of performance to the final model. For instance, for the training data, the final model produced an accuracy of 99.92% followed by precision of 100% and the recall of 99.32% with an average False Positive rate reaching at 0.07%. The overall F1 Score was 99.66% which indicated the model has an outstanding balanced performance and the parameter tuning were optimal. These improvements affected when predicting on the test data in a positive way. The prediction on the test data seems little decreased but still maintained the accuracy almost 95% followed by the improved precision of 85.20%. The recall value was the main concern as it was not meeting the minimum requirement of this project. Having made the changes, the recall had increased to 63.98% which is almost 10% increase. Moreover, the model had an average false positive rate of 18.98%, which has decreased by 2%. When considering the actual false positive rate for a Data Loss occurrence is about 14.79% while the F1 Score had hiked to 73.09%. This tells the final model produced in iteration IV is exceptionally good while meeting all the 3 criteriaâĂŹs to be considered as a successful model.

Finally, the tree produced in iteration IV, shows the utilisation of custom variables with meaningful nodes and leaves appears to be more logical and realistic. The influence of the tree is remarkable as the tree shows unique characteristics of incident that contributes larger proportion to a legitimate Data Loss. The characteristics are (incidents triggered during the office hours, subject type is New and the policy is P3, when email contains only 1 external recipient and the count of incidents per employee for department average is $> 6\%$).

Through a detailed inspection of the results across all iterations, it was identified that decision tree models utilised all the feature variables provided and demonstrated better performance as compared to Random Forest models. However, a thorough examination of the results it was discovered that the decision tree model failed to indicate potential limitations of the features provided which may require further adjustments. We conclude this to indicate reduced reliability for the results generated and therefore model cannot be fully trusted. On the contrary, Random Forest algorithm indicated that the model is insignificance although the accuracy was demonstrated as high. This enabled to perform further adjustments to the model and the



(a) Accuracy (b) Precision (c) Recall

Figure 14: A) Accuracy, B) Precision, and C) Recall for Random Forest experiments

Evaluation measure	Decision Tree (CART)			Random Forest		
Classifier	Test data	Test data	Validation data	Test data	Test data	Validation data
Severity feature	not included	included	included	not included	included	included
Accuracy	97.20%	92.80%	83.41%	97.66%	94.95%	90.03%
Recall	84.78%	55.17%	39.09%	84.67%	63.98%	75.60%
Precision	87.65%	72.72%	72.25%	91.70%	85.20%	79.88%
F-measure	86.19%	62.74%	54.00%	88.04%	73.09%	77.68%
Average TPR	92.95%	79.40%	78.58%	95.00%	90.80%	86.34%
Average FPR	7.04%	25.60%	21.41%	5.00%	18.98%	13.65%

Table 3: Summary of experimentation results and analysis

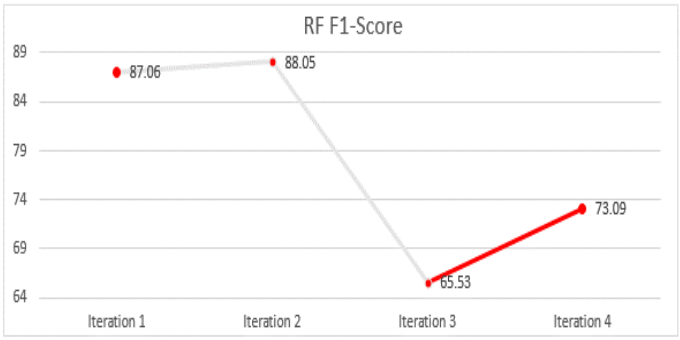


Figure 15: F1 performance measures for Random Forest

feature variables and therefore further experiments using reduced feature set were performed as part of iteration IV with Random Forest model. The model had an accuracy of 92.98% with a precision of 72% followed by recall of 55.17%. In addition, the F1 Score was 62.74. In view of all the aspects discussed so far, this identifies final model implemented in iteration IV using Random Forest algorithm to be the optimal and most effective in comparison with other iterations with both Decision Tree and Random Forest algorithms.

7. Conclusion and Future Work

A DLP system provides defence in depth for enterprise security systems by enabling detection and prevention of accidental and malicious data loss. It therefore enables protection against potential financial and reputational damages as well as compliance with emerging data protection legislation such as GDPR. However, a typical DLP system produces a large volume of alerts which also include significant proportion of false positives. Consequently, identifying legitimate data loss can be very challenging as each incident comprises of different characteristics often requiring extensive intervention by a domain expert to review alerts individually. This limits the ability to detect data loss alerts in real-time making organisations vulnerable to financial and reputational damages. This paper has presented a novel method using machine learning techniques to strengthen data loss detection capabilities of a DLP system. We conducted extensive experimentation using single decision tree and Random Forest algorithms with historical email incident data collected by a globally established telecommunication enterprise. The final model produced with Random Forest algorithm was identified as the most effective as it was successfully able to predict approximately 95% data loss incidents accurately with an average true positive value of 90%. Furthermore, the proposed solution successfully enables identification of legitimate data loss in email DLP whilst facilitating prioritisation of real data loss through human-understandable explanation of the decision thereby improving the efficiency of the process.

References

[1] V. M.-S. Fred H. Cate, Peter Cullen, "Data protection principles for the 21st century - revising the 1980 oecd guidelines," Organisation for Economic Co-operation and Development, Tech. Rep., 2014.

[2] J. R. David Reinsel, John Gantz, "Data age 2025 - the digitization of the world from edge to core," International Data Corporation (IDC), Tech. Rep., 2019.

[3] Accenture, "Cost of cybercrime study: Insights on the security investments that make a difference," Accenture - Ponemon, Tech. Rep., 2017.
[4] O. Gill and O. Rudgard, "British airways hacked as 380,000 sets of payment details stolen," 2018.
[5] E. U. A. for Network and I. S. (ENISA), "Enisa threat landscape report 2018 - 15 top cyberthreats and trends," Tech. Rep., 2019.
[6] "Gdpr portal: General data protection regulation," 2018. [Online]. Available: https://www.eugdpr.org/
[7] S. Alneyadi, E. Sithirasenan, and V. Muthukkumarasamy, "A survey on data leakage prevention systems," *Journal of Network and Computer Applications*, vol. 62, pp. 137 – 152, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804516000102
[8] E. Bickerstaffe, "Data leakage prevention," Information Security Forum, Tech. Rep., 2018.
[9] J. Arshad, M. A. Azad, M. M. Abdellatif, M. H. U. Rehman, and K. Salah, "Colide: a collaborative intrusion detection framework for internet of things," *IET Networks*, vol. 8, no. 1, pp. 3–14, 2018.
[10] J. Arshad, P. Townend, and J. Xu, "A novel intrusion severity analysis approach for clouds," *Future Generation Computer Systems*, vol. 29, no. 1, pp. 416–428, 2013.
[11] K. S. P. P. A. A.-N. S. V. R Vinayakumar, Mamoun Alazab, "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, pp. 41 525–41 550, 2019.
[12] R. V. Sitalakshmi Venkatraman, Mamoun Alazab, "A hybrid deep learning image-based analysis for effective malware detection," *Journal of Information Security and Applications*, vol. 47, pp. 377–389, 2019.
[13] M. Alazab, "Profiling and classifying the behavior of malicious codes," *Journal of Systems and Software*, vol. 100, p. 91âĂŞ102, 2015.
[14] F. Riaz, M. A. Azad, J. Arshad, M. Imran, A. Hassan, and S. Rehman, "Pervasive blood pressure monitoring using photoplethysmogram (ppg) sensor," *Future Generation Computer Systems*, 2019.
[15] K. Coussement and K. W. D. Bock, "Customer churn prediction in the online gambling industry: The beneficial effect of ensemble learning," *Journal of Business Research*, vol. 66, no. 9, pp. 1629 – 1636, 2013, advancing Research Methods in Marketing.
[16] A. Shabtai, Y. Elovici, and L. Rokach, *A survey of data leakage detection and prevention solutions*. Springer Science & Business Media, 2012.
[17] N. Miloslavskaya, V. Morozov, A. Tolstoy, and D. Khassan, "Dlp as an integral part of network security intelligence center," in *2017 IEEE 5th International Conference on Future Internet of Things and Cloud (FiCloud)*, Aug 2017, pp. 297–304.
[18] M. Vukovic, D. Katusic, R. Soic, and M. Weber, "Rule-based system for data leak threat estimation," in *2017 25th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, Sep. 2017, pp. 1–5.
[19] E. Costante, D. Fauri, S. Etalle, J. Den Hartog, and N. Zannone, "A hybrid framework for data loss prevention and detection," in *Proceedings - 2016 IEEE Symposium on Security and Privacy Workshops, SPW 2016, 23-25 May 2016, San Jose, California*. United States: Institute of Electrical and Electronics Engineers (IEEE), 8 2016, pp. 324–333.
[20] M. B. Salem, S. Hershkop, and S. J. Stolfo, *A Survey of Insider Attack Detection Research*. Boston, MA: Springer US, 2008, pp. 69–90. [Online]. Available: https://doi.org/10.1007/978-0-387-77322-3_5
[21] Symantec, "Machine learning sets new standard for data loss prevention: Describe, fingerprint, learn," Symantec, Tech. Rep., 2010.
[22] Y. L. MingJian Tang, Mamoun Alazab, "Big data for cybersecurity: Vulnerability disclosure trends and dependencies," *IEEE Transactions on Big Data*, vol. 5, no. 3, pp. 317 – 329, 2019.
[23] Forcepoint, "Its time for human-centric cybersecurity." Forecepoint, Tech. Rep., 2019.
[24] S. Jaiswal, A. Aggarwal, P. DiCorpo, S. S. Sawant, S. Kauffman, and A. D. Galindez, "Incremental machine learning for data loss prevention," Oct. 14 2014, uS Patent 8,862,522.
[25] J.-S. Wu, Y.-J. Lee, S.-K. Chong, C.-T. Lin, and J.-L. Hsu, "Key stroke profiling for data loss prevention," *2013 Conference on Technologies and Applications of Artificial Intelligence*, pp. 7–12, 2013.
[26] V. R. Carvalho and W. W. Cohen, "Preventing information leaks in email," in *SDM*, 2007.
[27] E. Costante, J. den Hartog, M. Petković, S. Etalle, and M. Pechenizkiy, "A white-box anomaly-based framework for database leakage detection," *J. Inf. Secur. Appl.*, vol. 32, no. C, pp. 27–46, Feb. 2017. [Online]. Available: https://doi.org/10.1016/j.jisa.2016.10.001
[28] K. W. Kongsgård, N. A. Nordbotten, F. Mancini, and P. E. Engelstad, "Data loss prevention based on text classification in controlled environments," in *International Conference on Information Systems Security*. Springer, 2016, pp. 131–150.
[29] H. Kim, J. Kim, I. Kim, and T.-m. Chung, "Behavior-based anomaly detection on big data," 2015.
[30] M. Kandias, A. Mylonas, N. Virvilis, M. Theoharidou, and D. Gritzalis, "An insider threat prediction model," in *Proceedings of the 7th International Conference on Trust, Privacy and Security in Digital Business*, ser. TrustBus'10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 26–37. [Online]. Available: http://dl.acm.org/citation.cfm?id=1894888.1894893
[31] "Uk data protection act 2018," 2018. [Online]. Available: https://www.gov.uk/government/collections/data-protection-act-2018
[32] C. H. Yu, "Exploratory data analysis," *Methods*, vol. 2, pp. 131–160, 1977.
[33] W.-Y. Loh, "Classification and regression trees," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1, no. 1, pp. 14–23, 2011. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/widm.8
[34] R. E. Banfield, L. O. Hall, K. W. Bowyer, and W. P. Kegelmeyer, "A comparison of decision tree ensemble creation techniques," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 1, pp. 173–180, 2006.
[35] M. A. Azad, R. Morla, J. Arshad, and K. Salah, "Clustering voip caller for spit identification," *Security and Communication Networks*, vol. 9, no. 18, pp. 4827–4838, 2016. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/sec.1656
[36] "The r project for statistical computing," 2019. [Online]. Available: https://www.r-project.org/
[37] G. Seni and J. F. Elder, "Ensemble methods in data mining: improving accuracy through combining predictions," *Synthesis lectures on data mining and knowledge discovery*, vol. 2, no. 1, pp. 1–126, 2010.
[38] A. Shalaginov and K. Franke, "Big data analytics by automated generation of fuzzy rules for network forensics readiness," *Applied Soft Computing*, vol. 52, pp. 359–375, 2017.
[39] R. Turner, "A model explanation system," in *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, Sep. 2016, pp. 1–6.