Cyber security incidents analysis and classification in a case study of Korean enterprises

Alaa Mohasseb 1 · Benjamin Aziz 1 · Jeyong Jung 2 · Julak Lee 3

Received: 13 March 2019 / Revised: 13 March 2019 / Accepted: 15 February 2020
© The Author(s) 2020

Abstract
The increasing amount and complexity of Cyber security attacks in recent years have made text analysis and data mining techniques an important factor in discovering features of such attacks and detecting future security threats. In this paper, we report on the results of a recent case study that involved the analysis of a community data set collected from five small and medium companies in Korea. The data set represents Cyber security incidents and response actions. We investigated in the study the kind of problems concerned with the prediction of response actions to future incidents from features of past incidents. Our analysis is based on text mining methods, such as n-gram and bag-of-words, as well as on machine learning algorithms for the classification of incidents and their response actions. Based on the results of the study, we also suggest an experience-sharing model, which we use to demonstrate how companies may share their trained classifiers without the sharing of their individual data sets in a collaborative environment.

Keywords Cyber security · Security data sets · Data mining · Machine learning · Text analysis

1 Introduction
The increasing amount and complexity of Cyber security attacks in recent years have brought data mining techniques into the attention of researchers and experts as an important technique...
in detecting such attacks through the analysis of data and side effects left by malware and spyware programs and incidents of network and host intrusions. In particular, text analysis and mining is widely used in many Cyber security areas, such as malware detection and classification (e.g. [14–16,18,22,24,27,29,33]) and malicious code detection (e.g. [7,30,31]). The popularity of social media has opened up the doors for text mining and analysis as important techniques for increasing the knowledge about users’ context, e.g. their location and time, and combining that knowledge with other attributes related to important events, topics, emotions and interests [21]. Text mining and analysis has also been used for predicting links [8] and detecting leaks of confidential data [28], the unintentional sharing of private health information [17], as well as in classical areas such as digital forensics [12], electronic communications analysis [36] and Web text analysis [19].

Despite all this popularity, the Cyber security community has remained somehow reluctant in adopting an open approach to security-related data due to many factors. These include political factors because of the continuing hesitance on the part of organisations to share data and technical factors related to the consistency, quality and the lack of consensus on the nature of variables that should be monitored or the metrics that should be used to quantify the security data themselves [20]. Nonetheless, in recent times, this trend has started to turn with the arrival of large and open security data sets and data-sharing platforms backed by the reliability and reputation of well-established organisations in the area of Cyber security. Examples of these include VCDB [34], CERT’s Knowledgebase at Carnegie Mellon University [10], SecRepo [26], CAIDA [9] and others. As a result, we are starting to witness an increasing trend in the usage of data sets in gaining insight into how Cyber incidents occur and how they should be defended against. A slightly different approach to the sharing of knowledge is based on what is known as community data sets, where a pre-set up federation allows organisations to share data, knowledge and experience. A good example of such federations is the European Cyber Security Organisation (ECSO) [1] and the European Network and Information Security Agency (ENISA) [2]. Other, more national-level hubs of Cyber security have also been set up all over the world for the same purposes.

The main goal of this study is to demonstrate that experience can be shared in a community of organisations based on a model of text analysis, where text classifiers can be used across organisations to predict future aspects of Cyber incidents. We consider, as a case study, one community data set that represents Cyber security incidents and response actions collected from five Korean companies. The data set was obtained from the KAITS Industrial Technology Security Hub [3] in Korea. In that regard, the paper has the following contributions:

- First, we demonstrate how Cyber security-related questions can be answered based from analysing a data set. In our analysis, we focus on questions related to the prediction of future responses to Cyber incidents from the type of malware or the name of the malicious code encountered in previous incidents. We demonstrate that this has good accuracy using standard text classification algorithms.
- We also construct an initial model of how trained instances of the text classifiers can then be shared across the companies without revealing their local data sets, therefore lifting the contribution of this paper from a pure technical level (text analysis) to a higher organisational level (experience sharing).

Our paper is an expanded version of a recent work presented in [27]. We expand on [27] by providing a more comprehensive related work survey and discussion, as well as providing full details of the motivating scenario, the data set used in the case study, the analysis model and the results of the classification process obtained. We give in this paper an initial experience-sharing model as an approach for future research based on the obtained results.
The rest of the paper is structured as follows. In Sect. 2, we discuss other works in the literature related to this paper. In Sect. 3, we describe the scenario motivating our work and the underlying data set sample used. In Sect. 4, we outline the research questions and the text mining model used in predicting the answers to these questions. In Sect. 5, we present the results of the experiments, and in Sect. 6, we discuss some aspects of these results. In Sect. 7, we outline a model of how companies in our scenario can share experience through the sharing of their classifier instances. Finally, in Sect. 8, we conclude the paper and outline some directions for future research.

2 Related work

The use of text analysis and data mining in detecting vulnerabilities and Cyber security threats is a research area that has been active for a number of years now, particularly with the massive proliferation of social media and user content-based applications. Most of this effort has been focused on the extraction of text in order to build information related to incidents. We review here a few examples of works that have used text and data analysis for the detection of malware and malicious code based on a variety of techniques.

Text mining has been used extensively in the area of intrusion detection, e.g. in [5,6,22,23,31,33,37]. In [5], the authors presented a method based on byte n-gram analysis to detect malicious code using common n-gram (CNG) analysis, which relies on profiles for class representation. The results showed that the applied method achieved 100% accuracy on training data and 98% accuracy in threefold cross-validation. Similarly, the authors in [31] presented a method to analyse suspicious files using OpCode n-gram patterns, which are extracted from the files data after dis-assembly for detecting unknown malicious code. The OpCode n-gram patterns can then be included in anti-virus programs as signatures. The evaluation of the methodology was performed using a test collection comprising of more than 30,000 files, in which various settings of OpCode n-gram patterns of various size representations and eight types of classifiers were evaluated. Evaluation results indicated that the proposed method achieved levels of accuracy higher than 96%.

In [6], text categorisation was used to learn the characteristics of normal and malicious user behaviour from data stored in the log files of Web servers, more specifically in the context of tele-medicine applications where malware programs can undermine user privacy. In [23], a framework was designed for the detection of intrusions based on system calls that uses text processing and data mining techniques. Once suspicious, system calls are textually analysed and then clustered based on the K-means method [25] to determine whether they fall within the malicious group of calls. The authors in [33] used data mining and text classification methods to detect security threats by extracting relevant information from various unstructured log messages, while in [22], the authors presented a text mining-based anomaly detection (TMAD) model to detect HTTP attacks in network traffic. TMAD is an anomaly detector that uses n-gram text categorisation and term frequency-inverse document frequency (TF-IDF) methods.

In addition, the authors in [37] proposed a method to automatically detect malicious code using the n-gram analysis. The proposed method used selected features based on information gain. Probabilistic neural networks were used in the process of building and testing the proposed multi-classifier system. The individual classifiers were used to produce classification evidences, where these evidences were then combined by the Dempster–Shafer combination rules [13,32] to form the final classification results for new malicious code. Experimental results showed that the detection engine improved compared to the classification results pro-
duced by the individual classifiers. Other works have used more general (i.e. non-textual) data mining methods in the context of malware detection and analysis. For example, in [15], the authors utilised hooking techniques to trace the dynamic signatures that malware programs try to hide, in which behaviour records are used for classification training and building a description model. Machine learning algorithms, such as naïve Bayesian, J48 (decision tree) and support vector machine, were then used for the classification process. The authors in [18] proposed a graph-mining method to detect variants of malware using static analysis, while covering the existing defects. The proposed approach combines static analysis and graph-mining techniques. In addition, a novel algorithm was proposed, called minimal contrast frequent subgraph miner (MCFSM) algorithm, which was used for extracting minimal discriminative and widely employed malicious behaviour patterns. The proposed method showed high detection rates and low false-positive rates. Additionally, in [14], an API-based association mining method was proposed for detecting malware. The criteria for API selection and the criteria for association rule selection were proposed to reduce the number of rules and to improve the quality of the rules. A classification method based on multiple association rules was adopted. The experiments showed that the proposed strategies can significantly improve the running speed of object-oriented association mining, in which the time cost for data mining was reduced by 32% and the time cost for classification was reduced by 50%.

Another area that has benefited largely from the application of data mining techniques is spyware detection and analysis. For example, in [7] a method was proposed for spyware detection using data mining techniques. The framework used the breadth-first search approach, which is known to work well for detecting viruses and similar software. Experimental results showed that the method had an accuracy of 90.5% for detecting spyware.

The authors in [35] proposed an integrated architecture to defend against surveillance spyware and used features extracted from both static and dynamic analysis. These features were ranked according to their information gains. In addition, a support vector machine classifier was created for each client. In order to keep the classifier up to date, a server collects reports from all clients as well as retrain and redistribute the new classifier instance to each client. The proposed spyware detection system achieved an overall accuracy rate of 97.9% for known surveillance spyware programs and 96.4% for unknown programs.

Our approach in this paper differs from all of the above works in that it not only demonstrates the use of text and data mining techniques in the analysis of Cyber security data sets, but also defines an initial model of how experience derived from analysing the data and training the classifiers can be shared in a community of organisations and used in future predictions of Cyber incidents across these organisations. In this respect, our paper contributes to the technical as well as the organisational dimensions of the problem. We are also more focused on predicting features of incidents rather than detecting an incident as a whole.

3 Motivating scenario and data set

We describe in this section the scenario motivating our work and the sample data set collected and analysed in the paper.

3.1 Description of the scenario

The main scenario motivating our work was based on the current operational model of the KAITS Industrial Technology Security Hub [3] in Korea. The hub is a public-private
Fig. 1 Experience-sharing scenario across *n* number of organisations

partnership supported by Korean governmental agencies in order to promote the sharing of Cyber security knowledge, experience and expertise across small and medium enterprises (SMEs) and increase their ability in responding to future incidents. In our case, this ability is interpreted as the ability in predicting attributes about future Cyber security incidents based on features of existing ones. The scenario is depicted in Fig. 1.

Underlying this scenario is the assumption that each instance of a classifier represents the experience of the company that owns it. This assumption is based on the fact that the classifier is trained using the company’s past data, which represent the company’s past experience with Cyber security incidents. Therefore, as depicted in the figure, the sharing of experience is simply translated in our case as the sharing of trained classifiers. This sharing can be regulated through some experience or classifier sharing agreement, which would rely on the predictive qualities of each instance of the classifier being shared. The scenario brings with it a number of operational constraints; however, one in particular has an impact on the approach we follow for the rest of the paper.

**Constraint [Each company’s data set is secret to that company].** Companies share their trained classifier instances but not their internal data sets. Therefore, we cannot assume in this scenario the existence of an overall hub-level data set consisting of the composition of the individual internal company data sets, and thus create only a single instance of a classifier. As a result, the scenario requires that the sample data sets collected be kept isolated from one another, therefore leading to the creation of different instances of the classification algorithms. In practical terms, this meant that we had to keep the individual company files separate rather than combining them in one single master file.

### 3.2 Description of the data set

The data set represents a sample of Cyber security intrusion events detected in five Korean SMEs over a period of ten months and collected by the KAITS industrial hub [3]. These companies are known by the pseudonyms DF, MT, SE, EP and MS. The data for each company are stored in a separate file, kept separate from the other files. There are 4643 incidents overall in the data set, and these are distributed as shown in Table 1.
Each entry in a file (expressed as a row) has the following metadata:

- **Date and Time of Occurrence**: this is a date/time value representing the date and time of the incident’s occurrence.
- **End Device**: this is a textual (string) value representing the name of the end device affected in the incident.
- **Malicious Code**: this is a textual (string) value representing the name of the malicious code detected in the incident.
- **Response**: this is a textual (string) value representing the response action that was applied to the malicious code.
- **Type of Malware**: this is a textual (string) value representing the type of the malware (malicious code) detected in the incident.
- **Detail**: this is a textual (string) value to freely describe any other detail about the incident (example the location in the computer where the malware resided).

The following is an example entry from one of the files in the data set:

(14/02/2017 11:58, rc0208-pc, Gen:Variant.Mikey.57034, deleted, virus, C:\Users\RC0208\AppData\Local\Temp\is-ANFS3.tmp\SetupG.exe)

## 4 Research questions and methodology

Our research in this paper is aimed at investigating one class of problems, where we take a text classification approach to solving them. This class of problems attempts to predict future aspects of Cyber security incidents. More specifically, we are concerned with questions related to how organisations can gain the ability to predict response actions to future Cyber security incidents involving malware. We consider the following two questions:

**Q1. How to predict a response action from the name of malicious code**

**Q2. How to predict a response action from the type of malware involved**

The significance of questions like these is that they can help in planning the level of resources that an organisation needs to allocate in response to future incidents in an efficient manner, as well as to take further actions to limit the spread of malware and to maintain the organisation’s business continuity. We note that the three metadata elements of the data set relevant to the above two questions that will be used in the classification process are Malicious Code, Response and Type of Malware. We will discuss later the classes we extracted for each one of these metadata.

Our proposed model consists of the following two main phases: (1) analysis and (2) classification. This model is illustrated in Fig. 2. The purpose of creating the model was to prepare the data for features selection and classification.

We next discuss the two phases of this model.

---

**Table 1** Incidents distribution

| Company name   | Total number of incidents |
|----------------|---------------------------|
| Company 1 (DF) | 932                       |
| Company 2 (MT) | 633                       |
| Company 3 (SE) | 923                       |
| Company 4 (EP) | 448                       |
| Company 5 (MS) | 1707                      |
The analysis phase is divided into two sub-phases: pre-processing and features extraction. These are defined as follows:

**Pre-processing:** This phase is mainly responsible for text pre-processing as some of the data entries in the data set consist of textual values (e.g. Detail, Malware Detection). The main objective of pre-processing is to clean the data from noise in order to improve the accuracy of the results by reducing the errors in the data. This is done by removing special characters and stop words such as “a” and “the”, punctuation marks such as question and exclamation marks, and numbers. In addition, all terms are converted to lowercase. The resulting terms are used to generate the n-gram features. For example, in the Response data entries, the following has been given as an entry “The file is a malware and has been blocked”, and after pre-processing, the following information is extracted “file malware blocked”.

**Features extraction:** In this phase, features are extracted from the three main classes in the data set (i.e. Malicious Code, Response and Type of Malware). The most commonly used features in text mining are n-gram and bag-of-words. The model makes use of “bigrams”, which are n-grams where n=2. For example, each two adjacent words create a bigram such as “malware detection”. In this phase, a bag-of-words is created, which contains all words (bigrams) and features are created from the three metadata elements above. This bag-of-words is filtered based on the minimum term frequency method, where terms that occur in less than the minimum frequency are filtered out and not used as features. For example, in the Response data entries, after the pre-processing phase, the following information is extracted, “file malware blocked”. However, since terms that occur in less than the minimum frequency are filtered out and not used as features, the following features are then extracted from the above information, “malware, blocked”.

The classification phase makes use of four machine learning algorithms, which are J48 decision tree, support vector machine (SVM), naïve Bayes (NB) and random forest (RF) algorithms, in order to build, test and compare the n-gram features predictive models. Each company’s data set is split into training and test sub-data sets. The training data set is used for building the model, and the test data set is used to evaluate the performance of the model. Hence, the features that have been extracted are used in the classification process.

To assess the performance of the machine learning classifiers, KNIME [4] was used. The experiments were set up using the typical tenfold cross-validation, i.e. the data set is split into tenfold, and each fold is used, in turn, for testing, while the other 9 are used for training. The output of the training process is a model, which is then used for the classification in the test fold. The labels produced by the model are matched to the true labels and typical performance indicators, such as accuracy, precision, recall and the F-measure [11].
5 Results

In this section, we present the results of the application of each of the four machine learning algorithms to each of the five companies data sets, in order to predict the outcome of the questions we posed in the previous section. For brevity, we show only the accuracy results for each case presented as percentage values. More detailed results can be found in Appendix at the end of the paper.

5.1 Q1 Results

Table 2 presents the classification performance results of the J48, NB, RF and SVM classifiers for the five companies for the question “How to predict a response action from the name of malicious code”. Full details can be found in Tables 7, 8, 9, 10 and 11 in Appendix.

For Company 1 (DF), SVM demonstrated the highest classification accuracy with 87%, whereas RF was the lowest with 82% accuracy. J48 and NB classifiers had a low recall value for “deleted” responses. More specifically, looking at where the errors occurred, J48 and NB classified most of this category as “segregated”. Similar to “deleted” responses, J48, NB and RF had relatively low recall values for “not defined” responses and classified a small percentage of these as “deleted” and “segregated”. In addition, SVM and RF could not identify “blocked” responses and incorrectly classified these as “segregated”.

For Company 2 (MT), SVM and RF showed the highest accuracy with 87%, whereas accuracy was 85% for NB, which was the lowest value. For this company, all the classifiers had 0% precision, recall and F-measure values for the cases of “none” and “name changed” responses, which were misclassified as “deleted” and “segregated”. Furthermore, J48 had the lowest recall value for “blocked” responses and misclassified a percentage of these as “deleted”. SVM, RF and NB also had low recall values for the “deleted” response as small percentage of these were misclassified as “segregated”.

For Company 3 (SE), the classification accuracy for J48, SVM and RF was 89%, and for NB was 85%. Similar to Company 2 (MT), all the classifiers could not identify “name changed” responses and had 0% precision, recall and F-measure values. This category was misclassified as “none” and “segregated”. In addition, NB had low recall values for “deleted” and the rest of the classifiers had 0% precision, recall and F-measure values. Furthermore, J48, SVM and RF misclassified a percentage of “none” responses as “segregated”.

For Company 4 (EP), SVM had the highest classification accuracy with 91% and RF had the lowest accuracy with 82%. J48 and SVM had 0% precision, recall and F-measure values for “none” responses, and these were misclassified as “segregated”. In addition, similar to Companies 2 (MT) and 3 (SE), all classifiers could not identify “name changed” responses and these were misclassified as “segregated”. Furthermore, SVM and NB had a lower recall value for “deleted” responses and J48 and RF had 0% precision, recall and F-measure values.

Table 2  Performance of the classifiers for identifying the types of response based on malicious code—best results are highlighted in bold

| Company name | J48 (%) | SVM (%) | RF (%) | NB (%) |
|--------------|---------|---------|--------|--------|
| Company 1 (DF) | 83      | 87      | 82     | 84     |
| Company 2 (MT) | 86      | 87      | 87     | 85     |
| Company 3 (SE) | 89      | 89      | 89     | 85     |
| Company 4 (EP) | 86      | 91      | 84     | 87     |
| Company 5 (MS) | 93      | 93      | 93     | 89     |
Table 3 Performance of the classifiers for identifying the types of response based on malware—best results are highlighted in bold

| Company name   | J48 (%) | SVM (%) | RF (%) | NB (%) |
|----------------|---------|---------|--------|--------|
| Company 1 (DF) | 66      | 52      | 66     | 66     |
| Company 2 (MT) | 70      | 70      | 70     | 70     |
| Company 3 (SE) | 66      | 66      | 66     | 66     |
| Company 4 (EP) | 74      | 74      | 74     | 74     |
| Company 5 (MS) | 98      | 98      | 98     | 98     |

for “name changed” responses, which were misclassified as “segregated” and “blocked”. J48 misclassified “recovered” responses as “blocked”.

Finally, for Company 5 (MS), J48, SVM and RF had the highest accuracy value with 93% and NB had the lowest accuracy value with 89%. All the classifiers had 0% precision, recall and F-measure for “none” and “name changed” responses and misclassified these as “deleted” and “segregated”. Moreover, all the classifiers had relatively low recall values for “segregated” responses and misclassified a percentage of these as “deleted”.

5.2 Q2 Results

Table 3 presents the classification performance details of the J48, NB, RF and SVM classifiers for the five companies for the question “How to predict a response action from the type of malware involved”. Full results are shown in Tables 12, 13, 14, 15 and 16 in Appendix.

For Company 1 (DF), J48, RF and NB had similar accuracy results of 66%, while SVM had the lowest accuracy of 52%. All classifiers had a 100% precision, recall and F-measure values for type of response “not defined”. For “deleted” responses, J48, RF and NB had a high precision, recall and F-measure values, whereas SVM had a recall value of 100% but low precision and F-measure values. In addition, for “blocked” responses, J48, RF and NB had high recall values of 100%, while SVM could not identify this type and had 0% precision, recall and F-measure values. SVM misclassified this type as “deleted”. Furthermore, all the classifiers failed to identify or classify other types of responses such as “none”, “recovered”, “segregated” and “name changed” based on the malware code and had 0% precision, recall and F-measure values as a result. These types were all misclassified as “deleted”.

For Company 2 (MT), all classifiers had similar accuracy of 70%. The classifiers were able to classify only two types of responses, which were “not defined” and “segregated” and had 100% recall values for both. However, they failed to identify or classify other types of response such as “deleted”, “none”, “blocked” and “name changed” and consequently had 0% precision, recall and F-measure values for these responses. These were then wrongly classified as “segregated”.

For Company 3 (SE), all classifiers had similar accuracy of 66% and recall values of 100% for two responses: “not defined” and “blocked”. In addition, all classifiers had similar precision, recall and F-measure values for the response type “deleted”, although some percentage of cases of this type were misclassified as “blocked”. Moreover, all the classifiers failed to identify or classify other types of responses such as “none”, “segregated” and “name changed” and therefore had 0% precision, recall and F-measure values for these types. These types were then wrongly classified as “blocked”.

For Company 4 (EP), all classifiers had similar accuracy values of 75% and recall values of 100% for two responses: “not defined” and “blocked”. In addition, all classifiers had similar recall values of 80% for “deleted”. However, 20% of “deleted” were misclassified as
Table 4  Type of response distribution for the five companies showing data imbalance

| Type of responses | DF  | MT | SE  | EP  | MS |
|-------------------|-----|----|-----|-----|----|
| Blocked           | 166 | 89 | 411 | 231 | 5  |
| Deleted           | 39  | 98 | 172 | 91  | 1153|
| Name Changed      | 2   | 2  | 10  | 4   | 2  |
| None              | 65  | 3  | 43  | 9   | 4  |
| Segregated        | 153 | 288| 206 | 61  | 201|
| Not defined       | 0   | 153| 81  | 28  | 326|
| Recovered         | 42  | 0  | 0   | 24  | 10 |

Table 5  Type of malware distribution for the five companies showing data imbalance

| Type of malware        | DF  | MT | SE  | EP  | MS |
|------------------------|-----|----|-----|-----|----|
| Email attachment       | 372 | 0  | 117 | 99  | 1140|
| Spyware                | 93  | 153| 81  | 27  | 326|
| Virus                  | 467 | 480| 725 | 322 | 87 |
| Downloaded file        | 0   | 0  | 0   | 0   | 149|
| Web contents           | 0   | 0  | 0   | 0   | 5  |

“blocked”. Furthermore, all the classifiers failed to identify or classify other types of responses such as “none”, “recovered”, “segregated” and “name changed” and had 0% precision, recall and F-measure values. These were misclassified as “blocked” and “deleted”.

Company 5 (MS) had the best classification performance in which all classifiers had an accuracy value of 98%. However, similar to the previous four companies, all the classifiers failed to identify or classify other types of responses such as “none”, “recovered” and “name changed” and had 0% precision, recall and F-measure values. These types were classified as “deleted” and “segregated”. In addition, J48, SVM, RF and NB had recall values of 100% for two types of responses, “not defined” and “segregated”, and a recall value of 98% for response type “deleted”.

6 Implications

In this case study, many factors affected the identification and classification process outlined in Section 4 using machine learning. We discuss here the results of the previous section in the light of these factors.

The overall results for the identification of different types of responses based on the given malicious code indicated that SVM was the best classifier for all the five companies in terms of performance and accuracy. In addition, most classifiers could not identify response categories such as “none”, “blocked” and “deleted”. The results for the identification of the different types of responses based on the given malware showed that all classifiers were suitable for some types of responses, but they were still unable to identify or classify every response types. The poor performance from the classifiers was due to the fact that some types of malware were assigned by companies to multiple response types (e.g “segregated” and “name changed” were assigned to malware type “virus”) and the high and low frequency of some types affected the classification due to the imbalance of the categories. This data imbalance problem is clearly visible from the data distributions of Tables 4 and 5.
Table 6  Average performance of the four classifiers used for both questions

| Research questions | J48 (%) | SVM (%) | RF   | NB   |
|--------------------|---------|---------|------|------|
| Question 1         | 87.4    | 89.4    | 87 % | 86 % |
| Question 2         | 74.8    | 72      | 74.8 | 74.8 |

We can summarise the above observations by the following points:

1. The classification accuracy was affected by the imbalance of the data set categories as shown in Tables 4 and 5.
2. The performance of the classifiers was affected by the multi-labelling of some of the data categories.
3. The inconsistency of the categories used across the five companies (e.g. types of responses and malware), as shown in Tables 4 and 5, made the evaluation task difficult.
4. Our results showed that SVM is more suitable for the detection of types of response using malicious code, whereas J48, RF and NB are more suitable for the detection of types of response using malware.

Table 6 shows the average performance of the classifiers against the two questions considered in the study. It is clear from this table that Q2 was the more difficult question to predict the answer for, hence the low performance of the classifiers.

7 An experience-sharing model

In this section, we define an experience-sharing model that builds on the scenario of Sect. 3 and uses the results we obtained in Sect. 5 to demonstrate, in principle, how experience sharing can be implemented across organisations based on data prediction. The main motivation behind such experience sharing is to improve the ability of an organisation to predict features related to future incidents, e.g. in our case, improve the ability to answer the questions of Sect. 4.

We define experience sharing as the sharing of an instance of a classification algorithm that has been trained using data belonging to some company. Sharing implies sending a classifier, which we term the *internal* classifier, and possibly receiving either the other company’s classifier, termed the *external* classifier, or receiving some form of reward in exchange for sending the internal one. We do not consider here decisions related to the sending of a classifier instance, since this is determined by the quality of what is received. Therefore, we only focus on decisions related to the receiving of the other company’s classifier, where such external classifiers are compared in their quality to the internal ones to determine whether they are acceptable from the receiving company’s point of view. Since the assumption is that we compare the like-for-like in terms of the algorithms used, the difference will purely reflect the quality of the *experience* of either company as expressed by its local data set. For simplicity, we leave out of the scope of this model other administrative factors, such as the policy or contract governing the sharing process, where we consider these in future (richer) versions of this model.

Assuming $C$ is the set of all classification algorithms being considered across organisations, where in our case $C = \{J48, SVM, RF, NB\}$, and $Q$ is the set of questions an organisation aims to answer, here $Q = \{Q_1, Q_2\}$, then we define a classifier-receiving decision function, $\zeta : C \times C \times Q \rightarrow B$, as a predicate:

$$\zeta(c_{\text{int}}, c_{\text{ext}}, q) = \begin{cases} F & \text{if } (\text{accuracy}(c_{\text{ext}}, q) - \text{accuracy}(c_{\text{int}}, q)) < \ell \\ T & \text{otherwise} \end{cases}$$
where $c_{\text{int}} \in C$ is the internally trained classifier, $c_{\text{ext}} \in C$ is the externally trained classifier, which the organisation is considering to import, and $q \in Q$ is the question the company is attempting to predict the answer for. The purpose of this decision function is to determine whether a classifier should be received (True–T) or not (False–F). In this case, we have restricted the decision function to the comparison of the classifier accuracy only. The function would return false (F), if the difference in the accuracy between the externally and the internally trained classifiers was less than some predefined threshold percentage value, $\ell \%$. This means that the decision is not to receive the external classifier, as it will be of no use to the company since its accuracy in predicting the answer to the question is lower than the accuracy of the internally trained classifier. Note again here that we are assuming that the algorithm (e.g. J48, SVM, etc.) underlying both classifier instances is the same. Otherwise, if the external classifier exceeds $\ell \%$ percentage difference with the internal classifier, the decision will be true (T), i.e. to receive the external classifier. As noted earlier, it is straightforward to change the function such that it compares based on any other metric.

Figure 3 shows how the process of experience sharing would work between two companies using the decision function, $\zeta$.

*Example*. Let us consider an example of how the above decision function can be used. Assume Companies 1 (DF) and 5 (MS) would like to share their knowledge of answering $Q_1$. In this case, both would select SVM as their best classifier. The sharing process from both companies’ points of view would result in these two calculation of the $\zeta$ function, assuming that $\ell_{\text{DF}} = 5\%$ and $\ell_{\text{MS}} = 2\%$:

$$
\zeta_{\text{DF}}(\text{SVM}_{\text{DF}}, \text{SVM}_{\text{MS}}, Q_1) =
\begin{cases}
F & \text{if } (\text{accuracy} (\text{SVM}_{\text{MS}}, Q_1) - \text{accuracy} (\text{SVM}_{\text{DF}}, Q_1)) < 5\% \\
T & \text{otherwise}
\end{cases}
$$

$$
\zeta_{\text{MS}}(\text{SVM}_{\text{MS}}, \text{SVM}_{\text{DF}}, Q_1) =
\begin{cases}
F & \text{if } (\text{accuracy} (\text{SVM}_{\text{DF}}, Q_1) - \text{accuracy} (\text{SVM}_{\text{MS}}, Q_1)) < 2\% \\
T & \text{otherwise}
\end{cases}
$$

Fig. 3 The process of experience sharing between two companies
In the above example, we see that company DF has an incentive to accept company MS’s classifier as it improves on its own classification accuracy. On the other hand, company MS has no incentive to accept company DF’s classifier as the latter does not improve upon MS’s classification accuracy. As in other cases, the value of $\ell$ will determine what the outcome of the decision function is. Setting $\ell$, in a real-world case, would require an internal study by the organisation. We consider this as one possible future research work.

8 Conclusion

In this paper, we presented the results of a case study on analysing a Cyber security incidents community data set for five Korean SMEs. Our analysis framework was designed and implemented to answer two kinds of problems: (1) the prediction of response actions to future Cyber security incidents involving malware or malicious code, and (2) the utilisation of the knowledge of the response actions in guiding analyses to determine the type of malware or the name of the malicious code. Furthermore, text mining methods such as n-gram and bag-of-words were used to extract relevant features from the data set, and then we applied machine learning algorithms for the classification process. The experimental results showed good performance for the classifiers in predicing different types of response and malware. An initial model of how experience can be shared among companies was also defined based on the given scenario.

As future work, we aim to investigate whether other questions may be answerable within the scope of the features offered by the data set. We also plan to expand the analysis to other data sets in order to investigate the impact of handling the problem of class imbalance on the classification performance and accuracy. Finally, we aim to formalise a more comprehensive model for the sharing of text and data classifiers as a means of sharing experience across organisations. Such a model would allow organisations to “trade” their trained classifiers using, for example, a smart contract specifically designed for this purpose. Using the blockchain technology, for example, one could envisage a scenario where successful predictions are rewarded, whereas unsuccessful ones are penalised, therefore leading to a fairer trade model.

Acknowledgements The research work reported in this paper was supported by the University of Portsmouth Themes Research and Innovation Fund (TRIF) 2018—Grant Number 41611. The authors would also like to thank the KAITS Industrial Technology Security Hub for providing the sample data set and for recently awarding the research effort carried out in this paper a bronze prize in the annual competition on best research and innovation projects in Cyber security.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

Appendix: detailed performance results

The following tables present the detailed performance results for each case. P represents precision, R represents recall and F represents the F-measure. For simplicity, we keep these
metrics as ratios, whereas we present accuracy as a percentage in-line with the summary results presented in Tables 2 and 3.

See Tables 7, 8, 9, 10, 11, 12, 13, 14, 15 and 16.

### Table 7 Performance of the classifiers in identifying the different types of response based on the malicious code for Company 1 (DF)

| Class         | J48 | SVM | RF  | NB  |
|---------------|-----|-----|-----|-----|
| Accuracy      | 83% | 87% | 82% | 84% |
| None          | 1.00| 0.94| 0.97| 1.00| 0.94| 0.97| 1.00| 0.94| 0.97|
| Recovered     | 1.00| 1.00| 1.00| 1.00| 1.00| 1.00| 1.00| 1.00| 1.00|
| Segregated    | 0.73| 0.98| 0.84| 0.75| 0.98| 0.85| 0.64| 1.00| 0.78|
| Deleted       | 0.24| 0.14| 0.18| 1.00| 1.00| 1.00| 1.00| 1.00| 1.00|
| Not defined   | 1.00| 0.69| 0.82| 0.89| 1.00| 0.94| 1.00| 0.69| 0.82|
| Blocked       | 1.00| 1.00| 1.00| 0.00| 0.00| 0.00| 0.00| 0.00| 1.00|

### Table 8 Performance of the classifiers in identifying the different types of response based on the malicious code for Company 2 (MT)

| Class         | J48 | SVM | RF  | NB  |
|---------------|-----|-----|-----|-----|
| Accuracy      | 86% | 87% | 87% | 85% |
| None          | 0.00| 0.00| 0.00| 0.00| 0.00| 0.00| 0.00| 0.00| 0.00|
| Name changed  | 0.00| 0.00| 0.00| 0.00| 0.00| 0.00| 0.00| 0.00| 0.00|
| Segregated    | 0.89| 0.89| 0.89| 0.84| 0.91| 0.88| 0.84| 0.91| 0.88|
| Deleted       | 0.62| 0.92| 0.79| 0.70| 0.71| 0.70| 0.70| 0.71| 0.70|
| Not defined   | 1.00| 0.84| 0.92| 1.00| 0.84| 0.92| 1.00| 0.84| 0.92|
| Blocked       | 1.00| 0.61| 0.79| 1.00| 1.00| 1.00| 1.00| 1.00| 1.00|

### Table 9 Performance of the classifiers in identifying the different types of response based on the malicious code for Company 3 (SE)

| Class         | J48 | SVM | RF  | NB  |
|---------------|-----|-----|-----|-----|
| Accuracy      | 89% | 89% | 89% | 85% |
| None          | 0.69| 0.61| 0.65| 0.69| 0.61| 0.65| 0.69| 0.61| 0.65|
| Name changed  | 0.00| 0.00| 0.00| 0.00| 0.00| 0.00| 0.00| 0.00| 0.00|
| Segregated    | 0.81| 1.00| 0.89| 0.81| 1.00| 0.89| 0.81| 1.00| 0.89|
| Deleted       | 0.00| 0.00| 0.00| 0.00| 0.00| 0.00| 0.00| 0.00| 0.00|
| Not defined   | 1.00| 0.94| 0.97| 1.00| 0.94| 0.97| 1.00| 0.94| 0.97|
| Blocked       | 1.00| 1.00| 1.00| 1.00| 1.00| 1.00| 1.00| 1.00| 1.00|
Table 10  Performance of the classifiers in identifying the different types of response based on the malicious code for Company 4 (EP)

| Class          | J48  | SVM  | RF   | NB   |
|----------------|------|------|------|------|
|                | Accuracy: |      |      |      |
| None           | 86%  | 91%  | 84%  | 87%  |
| Name changed   | 0.00 | 0.00 | 0.00 | 0.00 |
| Segregated     | 0.78 | 1.00 | 0.88 | 0.73 |
| Deleted        | 0.00 | 1.00 | 0.32 | 0.00 |
| Not defined    | 1.00 | 0.94 | 0.87 | 1.00 |
| Blocked        | 0.87 | 1.00 | 0.95 | 0.91 |
| Recovered      | 0.00 | 0.00 | 0.78 | 0.78 |

Table 11  Performance of the classifiers in identifying the different types of response based on the malicious code for Company 5 (MS)

| Class          | J48  | SVM  | RF   | NB   |
|----------------|------|------|------|------|
|                | Accuracy: |      |      |      |
| None           | 93%  | 93%  | 93%  | 89%  |
| Name changed   | 0.00 | 0.00 | 0.00 | 0.00 |
| Segregated     | 0.99 | 0.76 | 0.86 | 0.99 |
| Deleted        | 0.88 | 1.00 | 0.93 | 0.88 |
| Not defined    | 1.00 | 0.98 | 0.99 | 1.00 |

Table 12  Performance of the classifiers in identifying the different types of response based on malware for Company 1 (DF)

| Class          | J48  | SVM  | RF   | NB   |
|----------------|------|------|------|------|
|                | Accuracy: |      |      |      |
| Deleted        | 66%  | 52%  | 66%  | 66%  |
| Not defined    | 0.96 | 0.90 | 0.47 | 0.96 |
| Blocked        | 1.00 | 1.00 | 1.00 | 1.00 |
| None           | 0.35 | 1.00 | 0.52 | 0.35 |
| Recovered      | 0.00 | 0.00 | 0.00 | 0.00 |
| Segregated     | 0.00 | 0.00 | 0.00 | 0.00 |
| Name changed   | 0.00 | 0.00 | 0.00 | 0.00 |
Table 13 Performance of the classifiers in identifying the different types of response based on malware for Company 2 (MT)

| Class            | J48 | SVM | RF | NB |
|------------------|-----|-----|----|----|
| Deleted          | 0.00| 0.00| 0.00| 0.00|
| Not defined      | 1.00| 1.00| 1.00| 1.00|
| Blocked          | 0.00| 0.00| 0.00| 0.00|
| None             | 0.00| 0.00| 0.00| 0.00|
| Segregated       | 0.60| 1.00| 1.00| 1.00|
| Name changed     | 0.00| 0.00| 0.00| 0.00|

Table 14 Performance of the classifiers in identifying the different types of response based on malware for Company 3 (SE)

| Class            | J48 | SVM | RF | NB |
|------------------|-----|-----|----|----|
| Deleted          | 1.00| 0.68| 0.81| 1.00|
| Not defined      | 1.00| 1.00| 1.00| 1.00|
| Blocked          | 0.57| 1.00| 0.72| 0.57|
| None             | 0.00| 0.00| 0.00| 0.00|
| Segregated       | 0.00| 0.00| 0.00| 0.00|
| Name changed     | 0.00| 0.00| 0.00| 0.00|

Table 15 Performance of the classifiers in identifying the different types of response based on malware for Company 4 (EP)

| Class            | J48 | SVM | RF | NB |
|------------------|-----|-----|----|----|
| Deleted          | 0.75| 0.80| 0.78| 0.75|
| Not defined      | 1.00| 1.00| 1.00| 1.00|
| Blocked          | 0.71| 1.00| 0.83| 0.71|
| None             | 0.00| 0.00| 0.00| 0.00|
| Recovered        | 0.00| 0.00| 0.00| 0.00|
| Segregated       | 0.00| 0.00| 0.00| 0.00|
| Name changed     | 0.00| 0.00| 0.00| 0.00|
Table 16 Performance of the classifiers in identifying the different types of response based on malware for Company 5 (MS)

| Class          | J48   | SVM   | RF    | NB    |
|----------------|-------|-------|-------|-------|
| Deleted        | 0.99  | 0.98  | 0.99  | 0.99  |
| Not defined    | 1.00  | 1.00  | 1.00  | 1.00  |
| None           | 0.00  | 0.00  | 0.00  | 0.00  |
| Recovered      | 0.00  | 0.00  | 0.00  | 0.00  |
| Segregated     | 0.87  | 1.00  | 0.93  | 0.87  |
| Name changed   | 0.00  | 0.00  | 0.00  | 0.00  |

References

1. European Cyber Security Organisation. https://www.ecs-org.eu/. Accessed 01 July 2018
2. European Union Agency for Network and Information Security. https://www.enisa.europa.eu/. Accessed 01 July 2018
3. KAITTS: Industrial Technology Security Hub. http://www.kaits.or.kr/index.do. Accessed 01 June 2018
4. Knime (2018) https://www.knome.com/. Accessed 07 June 2018
5. Abou-Assaleh T, Cercene N, Keselj V, Sweidan R (2004) N-gram-based detection of new malicious code. In: Proceedings of the 28th annual international computer software and applications conference, 2004. COMPSAC 2004, vol 2, pp 41–42. IEEE
6. Adeva JJG, Atxa JMP (2007) Intrusion detection in web applications using text mining. Eng Appl Artif Intell 20(4):555–566
7. Bahraminikoo P, Yeganeh M, Babu G (2012) Utilization data mining to detect spyware. IOSR J Comput Eng 4(3):01–04
8. Bartal A, Sasson E, Ravid G (2009) Predicting links in social networks using text mining and SNA. In: 2009 International conference on advances in social network analysis and mining, pp 131–136
9. Center for Applied Internet Data Analysis: CAIDA Data. http://www.caida.org/data/overview/. Accessed 05 Jan 2017
10. CERT Coordination Center: CERT Knowledgebase. http://www.cert.org/vulnerability-analysis/knowledgebase/index.cfm. Accessed 14 Jan 2017
11. Chinchor N (1992) Muc-4 evaluation metrics. In: Proceedings of the 4th conference on message understanding, MUC'92, pp 22–29. Association for Computational Linguistics, Stroudsburg, PA, USA
12. Decherchi S, Tacconi S, Redi J, Leoncini A, Sangiacomo F, Zunino R (2009) Text clustering for digital forensics analysis. In: Herrero Á, Gastaldo P, Zunino R, Corchado E (eds) Computational intelligence in security for information systems. Springer, Berlin, pp 29–36
13. Dempster AP (1967) Upper and lower probabilities induced by a multivalued mapping. Ann Math Stat 38(2):325–339. https://doi.org/10.1214/aoms/1177698950
14. Ding Y, Yuan X, Tang K, Xiao X, Zhang Y (2013) A fast malware detection algorithm based on objective-oriented association mining. Comput Secur 39:315–324
15. Fan CI, Hsiao HW, Chou CH, Tseng YF (2015) Malware detection systems based on API log data mining. In: IEEE 39th annual computer software and applications conference (COMPSAC), 2015, vol 3, pp 255–260. IEEE
16. Fan Y, Ye Y, Chen L (2016) Malicious sequential pattern mining for automatic malware detection. Expert Syst Appl 52:16–25
17. Ghazinour K, Sokolova M, Matwin S (2013) Detecting health-related privacy leaks in social networks using text mining tools. In: Zaiane OR, Zilles S (eds) Adv Artif Intell. Springer, Berlin Heidelberg, Berlin, Heidelberg, pp 25–39
18. Hellal A, Roudhane LB (2016) Minimal contrast frequent pattern mining for malware detection. Comput Secur 62:19–32
19. Hicks C, Beebe N, Haliscak B (2016) Extending web mining to digital forensics text mining. In: AMCIS 2016: Association for information systems surging the IT innovation wave—22nd Americas conference on information systems, pp 1–5.
20. Hoo KJS (2000) How much is enough? A risk-management approach to computer security.
21. Inkpen D (2016) Text mining in social media for security threats, chap. Situational awareness and threat assessment. Springer International Publishing, Cham, pp 491–517
22. Kakavand M, Mustapha N, Mustapha A, Abdullah MT (2015) A text mining-based anomaly detection model in network security. Glob J Comput Sci Technol
23. Kumar GR, Mangathayaru N, Narasimha G (2015) An approach for intrusion detection using text mining techniques. In: Proceedings of the the international conference on engineering and MIS 2015, ICEMIS ’15, pp 63.1–63.6. New York: ACM
24. Lu YB, Din SC, Zheng CF, Gao BJ (2010) Using multi-feature and classifier ensembles to improve malware detection. J CCIT 39(2):57–72
25. Macqueen J (1967) Some methods for classification and analysis of multivariate observations. In: Proceedings of the 5th Berkeley symposium on mathematical statistics and probability, pp 281–297
26. Mike Sconzo: SecRepo.com - Samples of Security Related Data. http://www.secrepo.com. Accessed 05 Jan 2017
27. Norouzi M, Souri A, Samad Zamini M (2016) A data mining classification approach for behavioral malware detection. J Comput Netw Commun 2016:1
28. Ojoawo AO, Fagbolu OO, Olaniyan AS, Sonubi TA (2014) Data leak protection using text mining and social network analysis. Int J Eng Res Dev 10(12):14–22
29. Rieck K, Trinius P, Willems C, Holz T (2011) Automatic analysis of malware behavior using machine learning. J Comput Secur 19(4):639–668
30. Schultz MG, Eskin E, Zadok F, Stolfo SJ (2001) Data mining methods for detection of new malicious executables. In: Proceedings 2001 IEEE symposium on security and privacy, S&P 2001, pp 38–49. IEEE
31. Shabtai A, Moskovitch R, Feher C, Dolev S, Elovici Y (2012) Detecting unknown malicious code by applying classification techniques on opcode patterns. Secur Inf 1(1):1
32. Shafer G (1976) A mathematical theory of evidence. Princeton University Press, Princeton
33. Suh-Lee C, Jo JY, Kim Y (2016) Text mining for security threat detection discovering hidden information in unstructured log messages. In: IEEE conference on communications and network security (CNS), 2016, pp 252–260. IEEE
34. VERIZON: VERIS Community Database. http://vcdh.org/. Accessed 21 Nov 2016
35. Wang TY, Horng SJ, Su MY, Wu CH, Wang PC, Su WZ (2006) A surveillance spyware detection system based on data mining methods. In: IEEE congress on evolutionary computation, 2006. CEC 2006, pp 3236–3241. IEEE
36. Xylogiannopoulos K, Karampelas P, Alhajj R (2017) Text mining in unclean, noisy or scrambled datasets for digital forensics analytics. In: 2017 European intelligence and security informatics conference (EISIC), pp 76–83
37. Zhang B, Yin J, Hao J, Zhang D, Wang S (2007) Malicious codes detection based on ensemble learning. In: International conference on communications and network security, pp 468–477. Springer

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Dr Alaa Mohasseb is a Lecturer in Computer Science at the School of Computing, University of Portsmouth. There, she is a member of the Computational Intelligent Research Group. She received her PhD in computing from the University of Portsmouth in 2018. Her current research interests are in the fields of text mining, natural language processing and machine learning and the implementation of methods in these fields to different applications such as question classification and cybersecurity threat prediction. She has won several awards including the bronze prize in the annual international competition on Best Research & Innovation Projects in Cybersecurity, run by the Korean Industrial Technology Security Hub.
Cyber security incidents analysis and classification...

Dr Benjamin Aziz is a Senior Lecturer at the School of Computing, University of Portsmouth. Benjamin holds PhD degree in formal verification of computer security from Dublin City University (2003) and has research interests and experience in the field of computer and information security, with over 130 publications related to areas such as security engineering of large-scale systems, IoT and SDN security, formal methods, requirements engineering and digital forensics. He is on board several program committees for international conferences and working groups, including ERCIM FMICS, STM, Cloud Security Alliance and IFIP WG11.3.

Dr Jeyong Jung is an Assistant Professor at the College of Social Sciences at University of Ulsan, South Korea. His research interests are cybercrime, darknet and cryptocurrency.

Dr Julak Lee is an Associate Professor at Chung-Ang University in Seoul, Korea. His research focuses on theoretical approaches for industrial security and industrial crimes. Currently, he is the president of the Korean Security Science Association and the vice-president of Korean Association for Industrial Security. He received a bachelors degree in Public Administration at Korea National Police University and a masters degree in Criminal Justice at Michigan State University. Also, he earned his PhD in Criminal Justice at the University of Portsmouth.