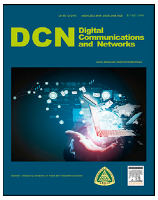
Digital Communications and Networks 3 (2017) 250–259

Contents lists available at ScienceDirect

Digital Communications and Networks

journal homepage: www.elsevier.com/locate/dcan



Visualization of big data security: a case study on the KDD99 cup data set

Zichan Ruan [*], Yuantian Miao, Lei Pan, Nicholas Patterson, Jun Zhang

School of Information Technology, Deakin University, Geelong, VIC 3220, Australia

ARTICLE INFO

Keywords:
Big data visualization
Sampling method
MDS
PCA

ABSTRACT

Cyber security has been thrust into the limelight in the modern technological era because of an array of attacks often bypassing untrained intrusion detection systems (IDSs). Therefore, greater attention has been directed on being able deciphering better methods for identifying attack types to train IDSs more effectively. Keycyber-attack insights exist in big data; however, an efficient approach is required to determine strong attack types to train IDSs to become more effective in key areas. Despite the rising growth in IDS research, there is a lack of studies involving big data visualization, which is key. The KDD99 data set has served as a strong benchmark since 1999; therefore, we utilized this data set in our experiment. In this study, we utilized hash algorithm, a weight table, and sampling method to deal with the inherent problems caused by analyzing big data; volume, variety, and velocity. By utilizing a visualization algorithm, we were able to gain insights into the KDD99 data set with a clear identification of “normal” clusters and described distinct clusters of effective attacks.

1. Introduction

Visualization is a critical part in analyzing data after its acquisition and preprocessing, because it intuitively and creatively helps represent data and comprehend the underlying relationship with the complex data [1–3]. Big data has three key characteristics, known as the three Vs: volume, variety, and velocity. These three characteristics can cause difficulties when operating big data, especially during its visualization. With huge amount of data having heterogeneous and diverse dimensionalities, conventional visualization methods show poor performances in functionalities, scalability, and response time when applied to big data [2,4–6].

Existing research efforts emphasize the visualization of complex data sets, that is, data sets with many dimensions and records. The existing solutions often assume that complete data are ready and visualization jobs often utilize a considerable amount of time to generate results. In fact, these assumptions generate serious problems for real-world big data issues. For example, security applications often require very quick “snapshots” to be provided to human experts to make timely decisions, in which significant delays in visualizing data is unacceptable. Therefore, a robust solution is desired to visualize big data.

This paper proposes a novel sampling method along with a visualization algorithm introduced in Ref. [7] to ultimately provide a new approach to big data visualization in the KDD99 data set. By exploiting a hash algorithm, we were able to preserve the redundancy appearing in the visualization data set. In addition, we proposed a weight table to address and moderate the bias problem caused by severely imbalanced classes. Furthermore, by collecting data sets for visualization from a partial data set instead of a whole set, we remarkably reduced time consumption when sampling. Consequently, redundancy caused by volume and bias problems can be traced back to the issue of variety and the velocity requirements, which were solved by our sampling method when applied in the KDD99 data set. Our approach provides a clear insight into types of cyber-attacks exhibited in the KDD99 data set; these have proven to be specifically successful according to the visualization results, allowing a clearer pathway on which Intrusion Detection Systems(IDSs) can be trained to be more effective in combating these problematic attacks.

Vast amount of research has been conducted by using the KDD99 data set with respect to intrusion detection [8–10]; however, not many studies have focused on the visualization of the KDD99 data set. Big data sets always suffer from redundancy and bias problems; using employing our sampling method, redundancy could be avoided while bias problems can be moderated. It is also worth mentioning that the results of this study show satisfactory class-condition clusters with inner-class compactness and inter-class distinctness. The idea of reducing data points through visualization is a creative way of representing data, and allows the human eye to intuitively receive information about a data set. The

[*] Corresponding author.
E-mail addresses: zichanr@deakin.edu.au (Z. Ruan), myuanti@deakin.edu.au (Y. Miao), l.pan@deakin.edu.au (L. Pan), nickp@deakin.edu.au (N. Patterson), jun.zhang@deakin.edu.au (J. Zhang).

https://doi.org/10.1016/j.dcan.2017.07.004
Received 25 February 2017; Received in revised form 8 May 2017; Accepted 19 July 2017
Available online 12 August 2017

"Outline" could provide an abstract of sufficient information from the big picture, while "Detail" could provide specific information of classes with small distributions.

The remainder of this paper is organized as follows: Section 2 reviews big data visualization and previous works about the KDD99 data set. A brief introduction to the visualization algorithm and detailed information of the sampling method are then presented in Section 3, followed by Section 4 comprising specific information about the KDD99 data set, experimental setup, and the results with comprehensive discussion. Finally, Section 5 provides the closing arguments of the paper along with the future research direction.

2. Literature review

2.1. Big data visualization

Although there is no widely accepted definition of big data, the three are recognized as the main and critical characteristics of big data [4,5,11,12]. Volume indicates that the size of the data set is large, variety implies high complexity and diversity of the types and resources of the data generated, and velocity suggests that the speed of data managed and analyzed should be fast. However, these features make it impossible for the conventional hardware or data-processing approaches to acquire, store, manage, and analyze big data within an acceptable time frame [3,11–13]. In addition, they bring about challenges regarding further insights, developments, and exploits of big data in finance, marketing, and networking fields.

Undoubtedly, research into big data could bring attractive opportunities and an incalculable values. However, we still face numerous challenges when attempting to take advantage of big data because of its inherited features, which add up to the difficulties when experts are trying to obtain, store, analyze, and visualize big data. Computer architecture face challenges regarding conventional hardware, for example, CPU-heavy but I/O-poor has been an obstacle to the progression of further research upon big data for decades [4,5]. Nevertheless, we focused on the data processing approaches disregarding the difficulties on the hardware and attempted to make a breakthrough by improving the mechanism when analyzing the data set. Big data analysis features heterogeneity in type, structure, scale, accessibility, and granularity of the data sets [13–15]. Data representation should be the primary focus of concern as this is the prior step to analyzing a data set by gaining an understandable and meaningful data structure as well as semantics for computer analysis and interpretation. Thus, a proper representation with integrated and effective data processing technologies and a structured mechanism is needed to help analysts see the insights of different datasets, while sufficiently revealing data, thus enhancing the significant features of the data sets and ignoring the noises contained within. Commonly, there are always a large amounts of redundancy in real-world data sets [14,15]. This downfall requires the use of redundancy reduction technologies in this field. Moreover, the inconsistency and incompleteness of a data set could result in difficulties in data processing. That is, although raw data should be preprocessed before further operation, a set of integrated and effective data processing techniques to retain the main characteristics of data without misunderstanding can be difficult to achieve. In this study, we focus on the data analysis phase, especially the visualization technology as data analysis is a method of transforming the acquired raw material and storing it into valuable information; however, there is limited research based on the visualization of big data.

Visualization serves as an effective and intuitive solution to represent and convey data information in an aesthetic and understandable manner by using different visualization methods such as pie charts, line charts, and venn diagrams [4]. As stated by Tam and Song [11], within an intuition visualization, interesting and significant information or patterns could be easily discovered by observers in spite of their complexity to discern in a statistical manner. Regarding complex and large-scale data sets, sophistic visualization can provide valuable information hidden in tables. The difficulties within big data visualization could be traced back to the natural characteristics of big data: volume, variety, and velocity. These characteristics determine a proper data representation, which is a critical step in the progress of big data visualization [16]. To our knowledge, no existing visualization study has focused on the visualization of KDD99 data set; two visualization methods for presenting big data sets are analyzed in the following paragraphs. However, owing to the difficulty of visualizing the KDD99 data set and limitation of the two methods, we propose our own visualization method to address the KDD99 visualization problem.

Glatz et al. [17] were inspired by the NetFlow system and tried to visualize the communication logs. After acquiring and handling the communication log from large traces, they applied Frequent Itemset Mining (FIM) to retrieve sets of interesting traffic patterns. Based on the continuation of the visualization, the extracted patterns into hypergraphs, which are capable of presenting connections among multiple features were transformed. Fig. 1 is an application of their scheme. The top 5 most frequent itemsets were extracted from 15,000 traffic flows to illustrate the underlying information contained within those traffic flows. The blue circle shown in the figure presents the itemsets with their sizes presenting the volume of traffic flows captured. The red squares indicate the attributes of the set of traffic flows, and the lines with an arrow indicate the connection. Although the proposed scheme was found to be less time consuming and useful in profiling traffic patterns, this kind of visualization is statistic-based and lacks intuition. As there are no communication logs in the KDD99 data set, their visualization method is not suitable for KDD99.

Van der Maaten [1] explored the t-distributed stochastic neighbor embedding (t-SNE), which is an embedding technique used for the visualization of the heterogeneous data with diverse dimensions by using scatter plots. In the current study, the t-SNE was implemented along with two tree-based algorithms, that is, the Barns-Hut and dual-tree algorithms, paper, and several examples are provided [1,18]. The results showed that the performance of t-SNE could remarkably be promoted by transforming these two algorithms and approximating the gradient for learning t-SNE embeddings in $O(NlogN)$. Moreover, the Barnes-Hut variant of t-SNE outperformed the dual-tree variant in the result [1].

Fig. 2 shows the Barnes-Hut t-SNE used in the visualization of the TIMIT dataset containing 3696 spoken utterances by speakers from two genders; a total of 1,105,455 speech frames were acquired. Every speech frame was labelled with corresponding phone numbers from 39 phones. Thirteen Mel-Frequency Cepstral Coefficients (MFCC) features along with 39-dimensional delta and delta-delta features representation were utilized to present the speech frames. MFCC features within 7 windows width were combined; thus, 273-dimensional feature vectors were used as input data. The scatter plot in Fig. 2 illustrates that it was feasible for Barnes-Hut t-SNE to embed datasets with over one million data records (TIMIT data set). In addition, affordable time that is less than 4 hours, was consumed for the embedding construction in the process. However, the results of the scatter plot do not show clear class-conditional densities. Parzen density was introduced to analyze the result; surprisingly, the Parzen density map at the right of Fig. 2 indicates that the density of the points was not uniform over the embedding space, despite the scatter plot suggesting it. After the inspection of the individual classes density estimate, the results suggested that most classes were modeled by small, dense clusters in the two-dimensional embedding [1]. Although complete information is presented in Fig. 2 and the Barnes-Hut t-SNE is proved to be effective in visualizing data sets with high dimension and large volume, the graph is unreadable for human beings, and is not useful for further analysis and understanding of the data set. Thus, this visualization method is not sufficient for visualizing the KDD99 data set.

2.2. Previous works upon KDD99

The KDD99 data set is widely recognized as a benchmark for the evaluation of IDSs in data mining and feature selection aspects, and as

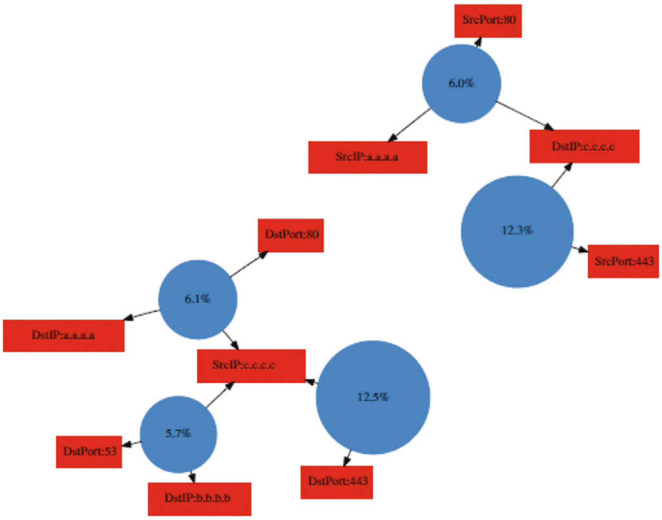


Fig. 1. Frequent itemset visualization of top 5 frequent item-sets extracted from 15,000 traffic flows based on the scheme proposed in Ref. [17].

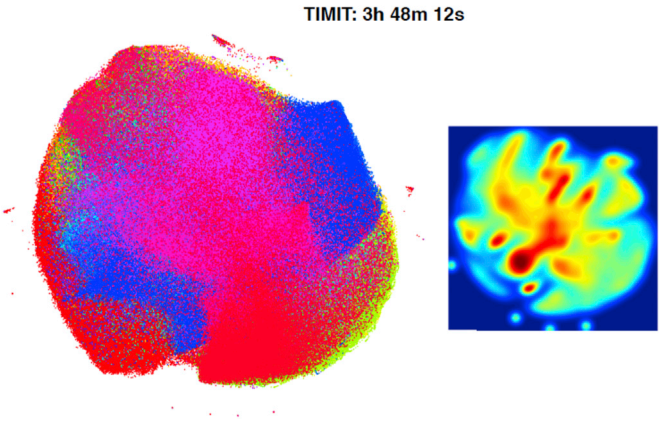


Fig. 2. Barnes-Hut t-SNE visualization results plotted using the TIMIT speech frames data set. The left figure shows a scatter plot in which the colors of the points indicate the classes of the corresponding objects. The right figure shows a Parzen density estimate of the two-dimensional embedding. The title of the figure indicates the computation time that was used to construct the corresponding embeddings: 3 h 48 min and 12 s [1].

Table 1
Comparison of previous works upon the KDD99 data set.

Work	Data Set	Dimensionality Reduction Technique	Results	Visualization
(George, Annie 2012) [8]	KDD99	PCA	28 PCA improved the performance of SVM	Bar chart
(Hashem, Soukaena Hassan 2013) [9]	HybD	PCA, GR	SVM outperformed NB in accuracy, while PCA showed further positive effects than GR in SVM classifier	Not applied
(Aladesote, O Isaiah et al., 2016) [10]	KDD99	PCA, GR	13 features were considered significant	Line chart and bar chart
Our work in this paper	KDD99	PCA, MDS	Novel sampling method and proper visualization technique provide big data visualization	Pie chart and scatter plot

Table 2
An example of a weight table.

Class	Proportion(%)	Weight(%)
Class 1	50	13.79
Class 2	10	68.97
Class 3	40	17.24

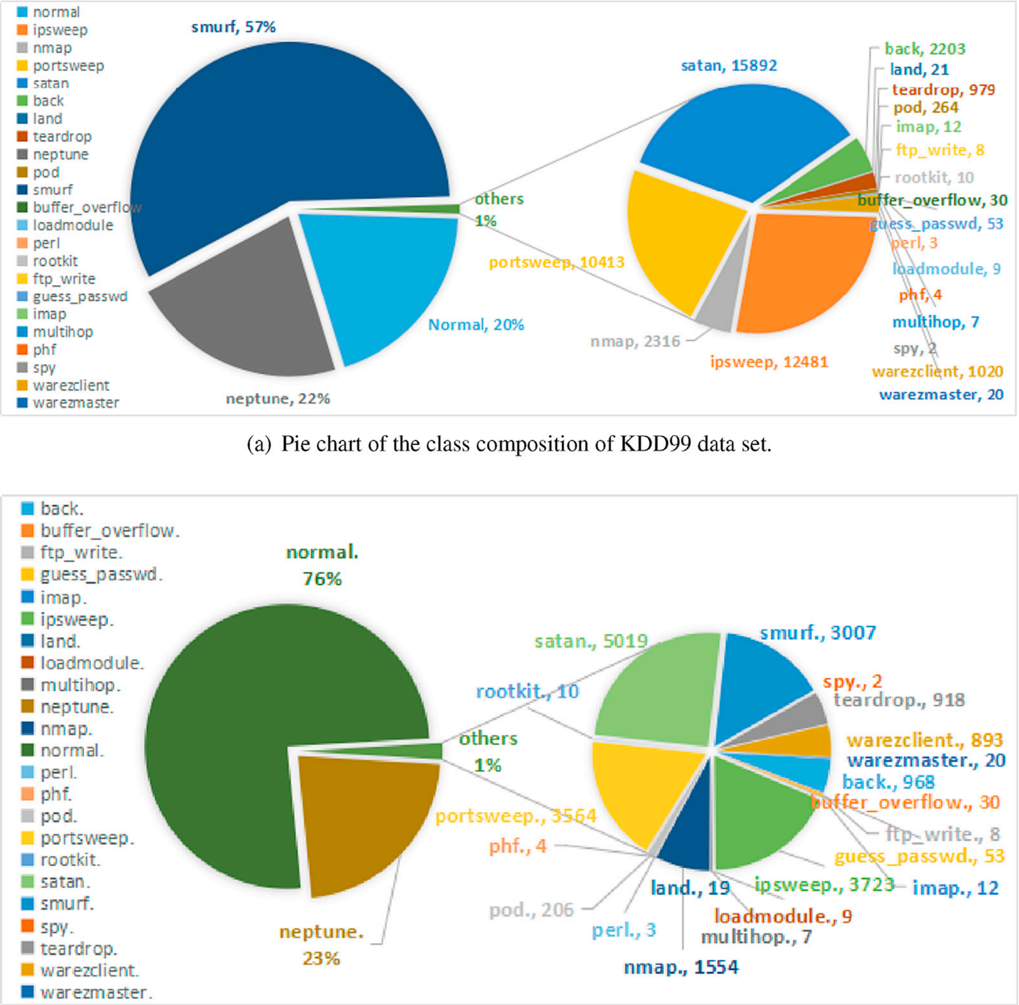
this study utilizes it as an example of big data visualization, it is necessary to review previous literature regarding the KDD99 dataset. This data set contains approximately 4,900,000 41-features data records with a label for each record showing the normality of corresponding data records. The detailed information of the KDD99 data set are provided in Section 4.

In 2012, George [8] ran a set of experiments utilizing the support virtual machine (SVM) as a classification algorithm and applying the Principal Component Analysis (PCA) as a dimensionality reduction technique. After the results were acquired and analyzed, the author stated that PCA could improve the performance of SVM in the KDD99 data set. The results showed that the execution time declined from 0.293 to 0.009 s; moreover, the precision and recall of the classifiers with 28 of 41 principal components employed provided evidences that PCA promoted the accuracy of classification in SVM from 0.7708 to 0.9375 and from 0.8125 to 0.875. From the results, George further concluded that unsupervised dimensionality reduction could have positive influences in a supervised classification algorithm [8].

Hashem [9] introduced a new data set (hybrid dataset; HybD) composed of 10% KDD99 data set with suggested host-based features to undergo experiments. HybD contains 41 features from KDD99, 3 from suggested host-based features, and 1 label indicating the current data record is normal or was an attack. These newly added features were highly connected to the host, and thus could give explicit information toward the host itself when the proposed IDS was detecting attacks. As in Ref. [8], Hashem selected SVM and PCA to operate the anomaly detection; however, the author added Naive Bayesian (NB) classification algorithm and Gain Ratio (GR) dimensionality reduction method to obtain more complete results through comprehensive comparisons. The results demonstrated that the SVM generally outperform NB in terms of accuracy rate, even though the accuracy rate of NB could reach to 98.6%. According to the result analysis provided by the author, PCA and GR had similar non-positive effects in NB, while PCA and GR always boosted the accuracy of the SVM classifier in HybD and PCA helped SVM perform better than GR [9].

Unlike the previous works, Aladestote et al. [10] proposed a novel array of extraction methods to help conduct further research on the KDD99 data set. PCA and GR were exploited in their work to determine the most significant features out of all 41 attributes. The GR technique was employed, and it aimed for a 7 discrete attribute selection, while PCA was utilized for testing 34 continuous attributes. According to the threshold value set through equations, 13 features out of 41 features were considered to be relatively significant: *Protocol*, *Service*, *Flag*, *Land*, *Logged*$_i$*n*, *Duration*, *Dst_bytes*, *Wrong_fragment*, *Num_compromised*, *Serror_rate*, *Srv_rerror_rate*, *Src_bytes*, and *Dst_host_srv_diff_host_rate* [10]. Aladestote et al. believed that further studies on intrusion detection should retrieve these 13 relatively significant features to decrease execution time and convey more effective and efficient IDSs.

Table 1 summarizes and compares the information from the above-mentioned literature. All three studies selected PCA as one of dimensionality reduction techniques, and it is clear that PCA had positive influence in the intrusion detection; this motivated us to apply PCA in our study. Those works are impressive; however, none of them emphasized visualization. Limited charts were attached to explain the performance of classifiers or feature selections. We believe that proper visualization of the data set could reveal underlying information that would normally remain undiscovered.



(a) Pie chart of the class composition of KDD99 data set.

(b) Pie chart of the class composition of the data set after redundancy reduction.

Fig. 3. Pie charts of the KDD99 data set. Legends in the left are the color presentation of each attack type and "normal". Attack types occupy less than 1% of the whole data set and are displayed in the right pie chart where, detailed numbers of data records within corresponding attack types are revealed instead of proportions. Fig. 3(b) shows the data set after redundancy reduction of the original KDD99 data set presented in Fig. 3(a). It is clear that there are many redundancies within the original KDD99 data set; thus, we propose a novel sampling method to address this problem.

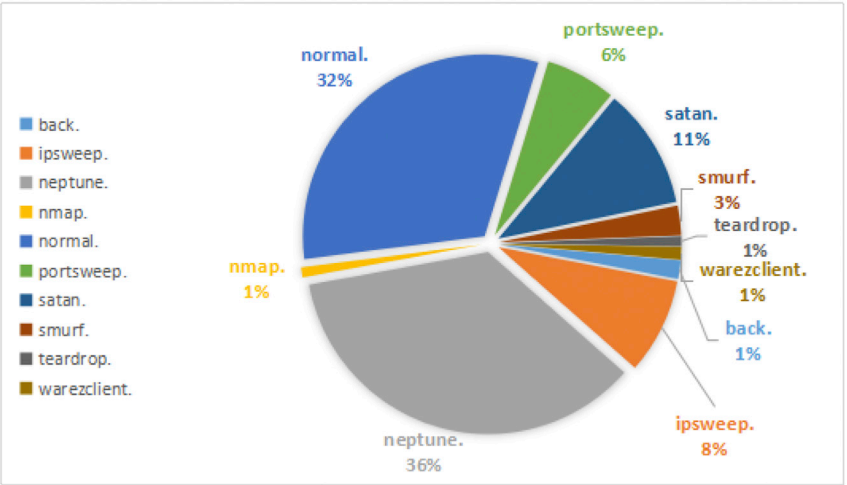


Fig. 4. Pie chart of the class composition of 800 data records sampled from the original data set.

Table 3
Comparisons between the classes distribution between the filtered data set and 800 samples from the KDD99 data set. The proportion of some attack types are 0, because the figures are too small and many digits are present instead of an empty set. The result indicates increased occupancies in the small classes within the Outline data set compared with those in the filtered data set. Thus, the sampling method helps reduce the influence of the bias problem and prevents the duplicated data points in the figures.

Distribution (%)	back	buffer_ overflow	ftp_ write	guess_ passwd	imap	ipsweep	land	load_ module	multi_ hop	neptune	nmap	normal	perl	phf	pod	port_ sweep	rootkit	satan	smurf	spy	teardrop	warez_ client	warez_ master
Filtered data set	0.09	0	0	0	0	0.35	0	0	0	22.53	0.14	75.61	0	0	0.02	0.33	0	0.47	0.28	0	0.09	0.08	0
Outline	1.63	0	0	0	0	8.50	0	0	0	35.63	0.88	31.75	0	0	0	6.25	0	10.75	2.63	0	0.88	1.13	0
Difference	1.53	0	0	0	0	8.15	0	0	0	13.10	0.73	−43.86	0	0	−0.02	5.92	0	10.28	2.35	0	0.79	1.04	0

3. System design

3.1. Visualization algorithm

According to the big data visualization algorithm introduced by Ruan et al., in 2017 [7], deliver aesthetic and intuitive visualization of big data, the natural features of data sets along with human factors should be considered when plotting figures. This specification determines the layout of figures and the limitations of human vision ability, which in turn decide an initial data point number. After comparisons and optimization, the number of data points within the figures should be 800 to be distinct for human beings to analyze. Regarding the color space, we utilized the online tool introduced by Harrower and Brewer [19] for the visualization experiments. The authors generated a new group of colormaps designed in the Hue, Value, and Chroma (HVC) color space; these could be easily converted into RGB or CMYK color spaces, which are employed in digital devices such as computers and printers.

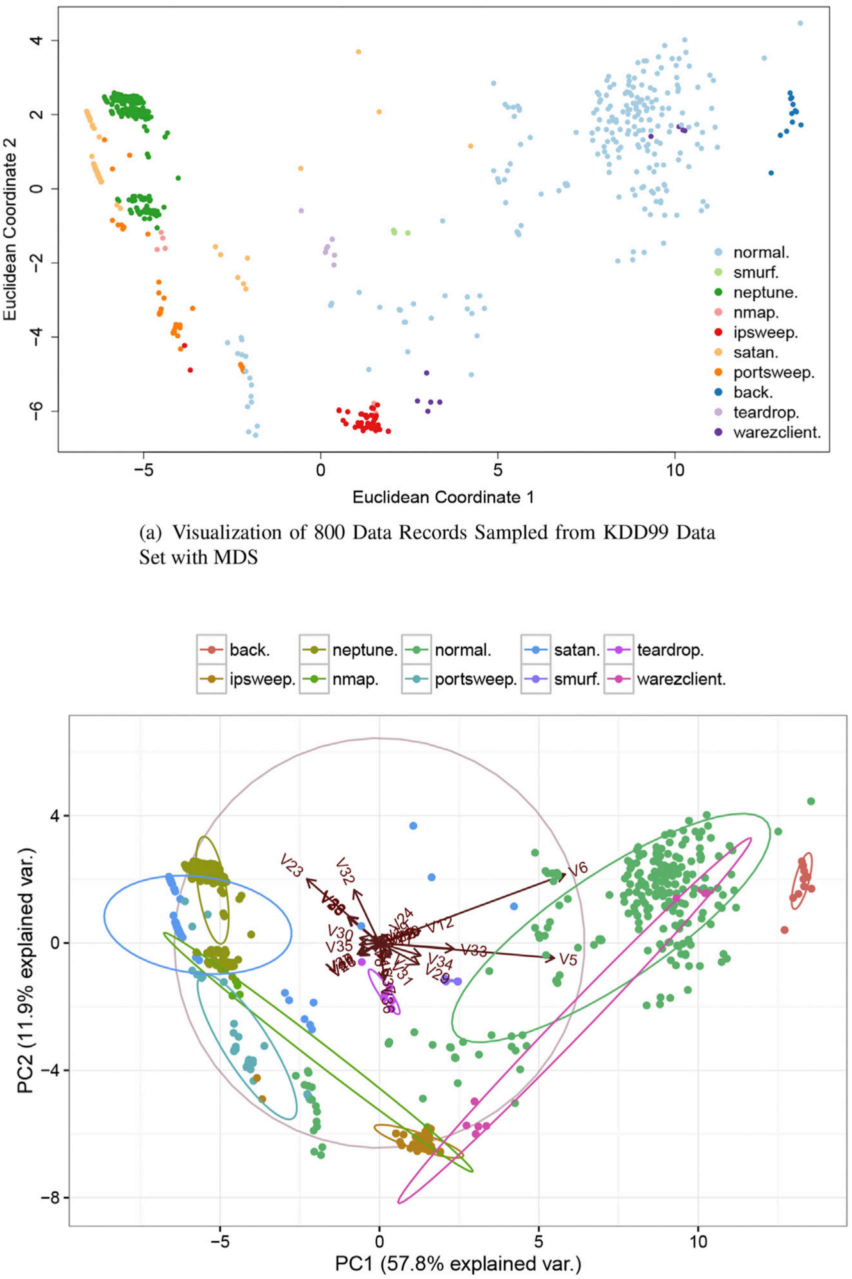
Two techniques were utilized in Ref. [7], that is, Multi-Dimensional Scaling (MDS) and PCA, to perform better big data visualization. These two techniques were proved to be beneficial to the performance of visualization in both criteria of inter-class distinctness and compactness. The current study continued to exploit MDS and PCA as important tools in big data visualization.

3.2. Sampling method

As mentioned earlier, redundancy could be within any real-world data sets; determining a proper way for reducing redundancy is critical for data representation. In the KDD99 data set, there exists severe duplications, and according to the Ref. [20], 78% of data records in the KDD99 data set are repeated. The existence of redundant data records not only compromises the performances of IDSs, it also negatively affects the visualization of the data set. There is requirement for a sampling method that allows users to make a reasonable selection of data records from a big data set to be plotted without duplications, while gaining sufficient information. Moreover, the imbalance problem should be emphasized before visualization; the serious biased distribution of data records could lead to the absence of some classes when sampling. The sampling method proposed in this paper addressing these two problems in big data sets, and provides an acceptable solution and affordable operational time. This is illustrated with the use of the KDD99 data set.

Before conducting the sampling, we need a weight table. Owing to the large volume of big data, it is always difficult or unaffordable for analysts to examine the whole data set to obtain the required information; thus, we take the first 10,000 sets of data as a fundamental aspect to retrieve a weight table to avoid the influence of imbalanced problems of different classes on the next stage. We assigned large weights to smaller classes to ensure that the data records in those classes could be selected in this sampling method. To explain this, assume that we received three classes from the first 10,000 data records; 5000 of those data records were class 1; 1000 of them were class 2; and the remaining 4000 were class 3. Accordingly, the proportion of each class in the whole data set is $P(1) = 50\%$, $P(2) = 10\%$, and $P(3) = 40\%$. The weight of class 1 will be calculated by equation (1), and weight of classes 2 and 3 could be calculated by the same equation. Table 2 presents the weight equation is used to reverse the weight of different classes within a data set. The final weight table of the data set. The weight equation can be used to modify the weight among classes within a data set. Before the application of the equation, class 1 had the most proportion; thus, it is more likely to be selected in random sampling. The condition is relieved when utilizing the weight table. To avoid the biased problem in the KDD99 data set, the weight equation and table are necessary.

$$W(class_i) = \frac{\sum_{j,k(j,k \neq i, j \neq k)}^{n} P(class_j)P(class_k)}{\sum_{j,k(j \neq k)}^{n} P(class_j)P(class_k)} \quad (1)$$



(a) Visualization of 800 Data Records Sampled from KDD99 Data Set with MDS

(b) Visualization of 800 Data Records Sampled from KDD99 Data Set with PCA

Fig. 5. Images produced by 800 sampling data records from original data set. (a) The KDD99 visualization by using MDS technology; the legend on the right lower corner is the color presentation. (b) The KDD99 visualization by using PCA technology; the legend on top is the color presentation. The color points represent data records in corresponding attack type and "normal" class. Ellipses in Fig. 5(b) are confidential intervals and can represent clusters of attack types and "normal" classes. The result shows that the "normal" cluster has a clear boundary and is easily identified by human eye; thus, the visualization is satisfactory.

In the sampling method, three tables are involved: *Selection, Blacklist,* and *Remains tables*. First, we randomly selected data records from the KDD99 data set, retrieved the hash code of the records, stored the data record and the hash code in the *Selection* tables, and placed the remainder of the KDD99 data set into the table *Remains table*. Next, we randomly selected 10 data records from *Remains*, placed the existing data records according to the table *Selection* into *Blacklist*, and ran a weighted selection according to the previous gained weight table, that is, Table 2 on whatever data records were remaining. We then placed the winner and its hash code in *Selection*, with the others in *Blacklist*. We also deleted those 10 processed data records from *Remains*. Then, we repeated the above-mentioned steps until enough data records were acquired in *Selection.*

There are three main advantages of this novel sampling method: less time consumption, no duplication, and bias mitigation of biases. By employing our proposed method, analysts could decrease execution time remarkably as there is no need to run the whole data set; this is a critical, effective, and necessary when dealing with large volumes of big data. In addition, no redundancies exist after the sampling; this is a considerable benefit for both procedures of data mining for intrusion detection and big data visualization. Moreover, the imbalanced classes problem could be moderated by utilizing this sampling method.

4. Experiments, results, and evaluation

To deliver a more detailed and accurate evaluation of our sampling method and visualization technique, the KDD99 data set was employed as an example when using our proposed methods. We aimed to produce a visualization system with an aspect of high adaptability that could be applied in big data sets to gain key insights.

4.1. KDD99 data set

Since the Third International Knowledge Discovery and Data Mining Tools Competition of 1999, KDD99 [21] had been widely utilized as the basic data set for the assessment of anomaly detection mechanisms. The KDD99 data set was built based on the DARPA98 data set, which was a collection of raw data through 7 weeks of network traffic, captured in 1998. There are 4,898,431 data records described by 42 features derived from the DARPA98 data set, showing the normality of those data records by labeling the records as "normal" or a specific attack type. In addition, 23 classes are included in the KDD99 data set, including one normal and 22 different types of attacks.

Fig. 3(a) shows a pie chart with the data distribution of the 23 classes. The original data set suffers from a severe imbalanced problem, as indicated in the pie chart. Except for classes "normal", "neptune", and "smurf", which stand for 20%, 22%, and 57% for the KDD99 data set, respectively, the remaining 20 classes consist of only 1% of the whole data set. Attack types "spy", "perl", "phf", "multihop", "ftp_write", "loadmodule", and "rootkit", contain no more than 10 data records in each class. This kind of biased situation continues even after removing the redundancies from the original data set, as shown in Fig. 3(b). The total number of records within the filtered data set are reduced to 1,074,992; majority of the redundant records are observed in class "smurf". Classes 'normal' and 'neptune' consist of 99% of the filtered data set, with respective percentages of 76% and 23%.

4.2. Visualization

The inherent deficiencies of the KDD99 data set are in the maintaining of the massive redundancies within the data set and the severe biases occurring among the 23 classes comprising the accurate visualization of the data set. Redundant records could lead to repeats in one location on the canvas; this cannot be observed but is critical to the results of the examination. By employing our sampling method proposed in Section 3, duplications could be avoided during the progress of sampling. However, the problem of biases could lead to the lack of information about certain types of attack as there are very little records for sampling. The novel sampling method we employed addresses the imbalance to this problem to some extent but is not efficient for the observation of the whole data set. To provide further functional solutions, visualization of KDD99 data set was performed in two aspects: the outline for rough pictures that help capture the major information and details, specific information which could not be noticed in the previous pictures.

The visualization was evaluated based on the criteria employed in our previous work [7]: ***distinctness*** and ***compactness***. Inner-class compactness is essential when defining whether a cluster is well organized, and inter-class distinctness indicates whether if two clusters were well separated and identifiable. Thus, the two factors were introduced to be the key factors when assessing an image by using our algorithm.

4.2.1. The outline of the data set

As mentioned in Section 3, the number of data points within a figure is 800. That is, regarding the KDD99 data set, 800 is approximately the maximum number of data points that could be relatively and clearly observed from a figure sized at 3.5 × 2.2 inch, with a minimal loss of information precision. By using our sampling method, 800 data records were retrieved from the KDD99 data set and composed a new data set "Outline" and the class distribution, as shown in Fig. 4 and Table 3. The sampling method helped reduce the influence of bias problem and prevent the duplicated data points in the figures. We must clarify that the occupancies of several classes within the filtered data set shown in Table 3 are 0%; this implies that there are few data records in these classes instead of none. Table 3 indicates an increase of occupancies in the small classes within the Outline data set, compared with those in the filtered data set. A remarkable decrease occurred in the largest class "normal", from 75.61% in the filtered data set to 31.75% in the Outline data set. In this data set, 10 types of classes, including "normal", and 9 attack types were chosen; however, the numbers of data records in other classes were so small, they could not be acquired during the sampling method. Owing to the inability to analyze a whole big data set, we tried to make this scenario similar to a real-world scenario. Thus, some classes in the KDD99 data set were ignored at this stage.

From the Outline data set, we were able to plot scatter diagrams by using dimensionality reduction techniques, MDS and PCA, and visualization methods introduced in Ref. [7], revealing insights and underlying relationships within those classes and even data records. Figs. 5(a) and 5(b) were generated after visualization by using MDS and the PCA technique respectively, and clearly shows that Fig. 5(a) the majority of "normal" clusters were gathered at the right upper corner of the image produced, with little interference from other attack types. In fact, only the "warezclient" cluster caused obstruction in the identification of a "normal" cluster. However, "warezclient" itself maintained high inner-class compactness in two locations in the image, making it easier to observe the distribution of the "normal" cluster. With the application of confidential eclipse in Fig 5(b), data records distributed through different

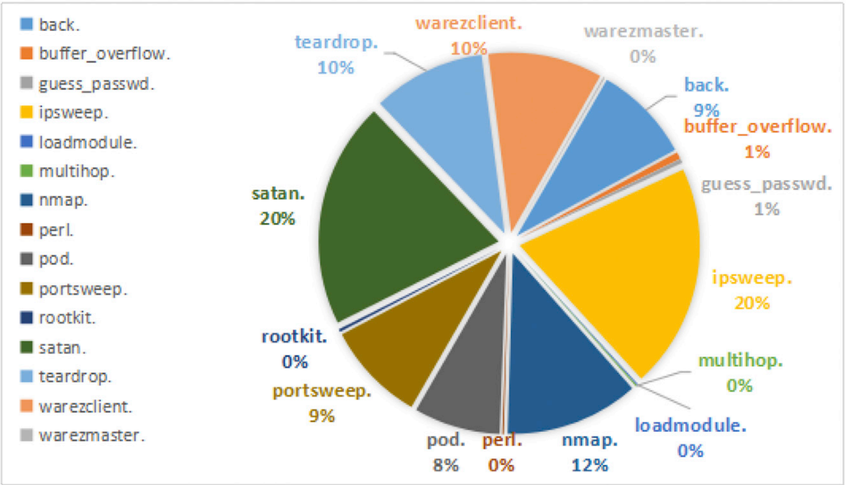


Fig. 6. Pie chart of the class composition of 800 data records sampled from the minority (1%) of the original data set.

Table 4
Comparisons between the classes distribution between filtered data set and 800 samples from the minority (1%) of the KDD99 data set. The proportion of some attack types are 0 because the figures are too small and many digits are presented instead of an empty set. The result indicates similar changes of proportion of attack types in the Detail data set as in the Outline data set.

Distribution (%)	Back	buffer_overflow	ftp_write	guess_passwd	imap	ipsweep	land	loadmodule	multihop	nmap	perl	phf	pod	portsweep	rootkit	satan	spy	teardrop	warez_client	warez_master
Filtered data set	5.69	0.18	0.05	0.31	0.07	21.87	0.11	0.05	0.04	9.13	0.02	0.02	1.21	20.94	0.06	29.49	0.01	5.39	5.25	0.12
Detail	8.75	0.63	0	0.38	0	20.13	0	0.13	0.13	12.00	0.13	0	7.75	9	0.25	20.25	0	10.13	10.25	0.13
Defference	3.06	0.45	−0.05	0.06	−0.07	−1.75	−0.11	0.07	0.08	2.87	0.11	−0.02	6.54	−11.94	0.19	−9.24	−0.01	4.73	5.00	0.01

classes are easily observed. The arrowhead lines (vectors) present 41 features, with their directions and lengths, encompassing two new coordinates. These two principal components were selected to plot an image two-dimensionally for a better observational experience. The first and second principal components respectively contained 57.8% and 11.9% of the information within the 41 features of the KDD99 data set, amounting to approximately 70% of the variation. The 5th, 6th, 23rd, and 32nd features were the important in representing the whole data set. Attack types “back”, “teardrop” and “smurf” were clearly clustered and well identified, with slight overlapping among attack types “ipsweep”, “neptune”, “nmap”, “portsweep”, and “santan”.

The results were relatively satisfactory as the “normal” cluster had a clear boundary and was easily identified by the human eye. Nevertheless, detailed information and visualization upon the other attack types are still needed for comprehensive research and study.

4.2.2. Detailed information of the data set

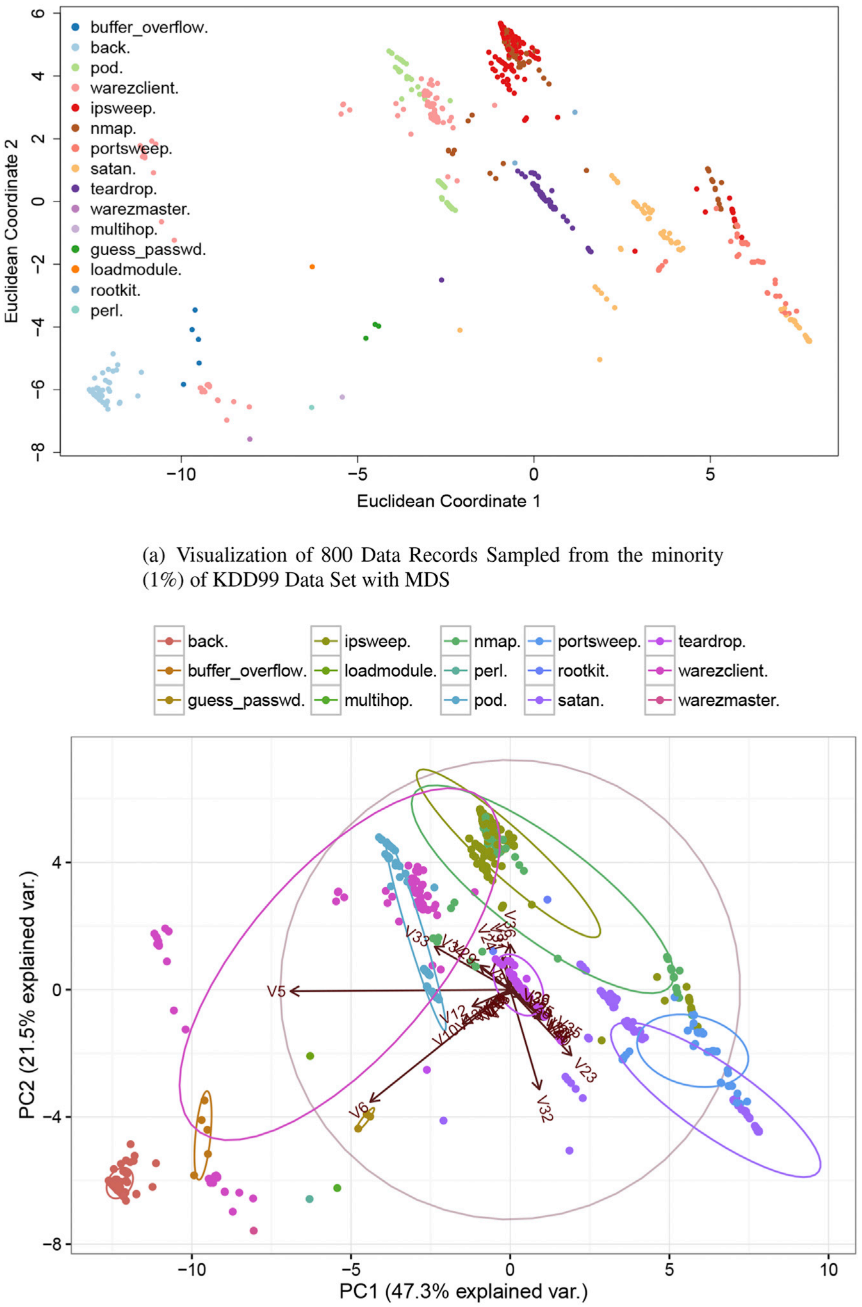
Minority (1%) of the KDD99 data set (Excluding the “normal”, “smurf” and “neptune” classes) was used as the research object for the application of the sampling method. In addition, 800 numbers of data points were selected for the visualization in the next progression and formed a new data set called Detail. The class distributions are shown in Fig. 6 and Table 4. It is obvious in Fig. 6 that the imbalanced problem was moderated in this sample. Note that the class distribution of the filtered data set excluded the classes “normal”, “smurf” and “neptune” as well for accurate comparison. Similar to the results of the Outline data set obtained using our sampling methods, the issue of parallelism appeared in the Detail data set. The novel sampling method favoured classes with a small occupancy and weakened the bias problem by decreasing the distribution in large classes. In this case, small classes with occupancy larger than 5% were beneficial, and large classes with occupancy larger than 20% were undermined. However, classes with occupancy less than 0.11% underwent a decrease in occupancy as a consequence of a few data records selected. Fifteen attack types were selected in the sampling method, and by using the new data set Detail, we were able to generate a detailed visualization upon these attack types.

A new set of images are plotted in Fig. 6. In addition, Fig. 7(a) and 7(b) focus on the MDS PCA techniques, respectively. Fig. 5(a) indicates that except for the attack types “back”, “guess_passwd”, and “teardrop”, which satisfied the criterion ***distinctness*** and formed clusters without overlapping with other classes, all other attack types suggested unclear boundaries. The confidential ellipse in Fig. 7(B) displays server overlapping among those attack types. The first principal component represents 47.3% of variation, and the second represents 21.5% of variation, summed at approximately 70% of variation. Similarly, in Fig. 5(b), the 5th, 6th, 23rd, and 32nd features are the most important in visualization. It is worth noticing that the clusters formed by data records in “ipsweep” and “nmap” were completely overlapped, while the “pod” cluster was nearly inside the “warezclient” cluster, and more than half of the “portsweep” cluster was overlapped on the “satan” cluster.

4.3. Discussion

The previous sections provide comprehensive comparison and representation of the two sets of images produced by the combination of the novel sampling method and visualization algorithm that we employed with sufficient explanation. Further discussion are as follows:

- The novel sampling method provides an effective way to avoid redundancies in the data set, reduce the influence caused by the bias problem, and significantly decrease execution time by browsing part of the data set instead of scanning the whole data set.
- The big data visualization algorithm introduced in our previous work [7] was employed in the experiments, and provided satisfactory results. By combining human factors, such as human vision limitation



(a) Visualization of 800 Data Records Sampled from the minority (1%) of KDD99 Data Set with MDS

(b) Visualization of 800 Data Records Sampled from the minority (1%) of KDD99 Data Set with PCA

Fig. 7. Images produced by 800 sampling data records from the original data set The KDD99 visualization by using MDS technology; the legend on the left upper corner is the color presentation. The KDD99 visualization by using PCA technology; the legend on top is the color presentation. The color points represent data records in the corresponding attack type. Ellipses in Fig. 7(b) are confidential intervals and can represent clusters of attack types.

and color perception with the computer display, the produced figures retained as much precision while providing less information loss.

- Visualization techniques MDS and PCA were utilized in plotting, helping with dimensionality reduction, and generating images in a two-dimensionally manner for easily manual observation by the analysts.
- As a result of a few data records in some attack types and to save operation time, our sampling method could not retrieve data records from those classes. Thus, the visualization was presented in two aspects: Outline and Detail. In this way, the classes with few data records were visualized in Detail, while those in the Outline visualization were not effected.
- The criteria introduced in Ref. [7] were used in the current study as well. Inner-class compactness and distinctness are critical factors when judging if clusters inside an image are well organized.
- The experimental results showed that the "normal" class was easily identified from other attack classes. This also explained the existence of high accuracy rates in IDSs [8,9] when dealing with this data set.
- There still exists severe overlapping among clusters of several attack types, which should be treated in the future work.

5. Conclusions and future work

Our contributions of this paper can be summarized into three key aspects: the novel sampling method, proof of the visualization algorithm in the KDD99 data set, and the idea of reduction of the data points within a figure instead of their addition. The advantage of our proposed sampling method is the management of the inherent flaws that big data sets comprise in terms of volume, variety and velocity. By exploiting the hash algorithm, weight table, and weight equation that we proposed, we addressed the bias and redundancy problems caused by volume and variety of the KDD99 data set. The sampling method remarkably reduced time consumption as a measurement toward velocity. Additionally, the experiments conducted show the evidence of the adaptability of the visualization algorithm proposed in Ref. [7]. The combination of the sampling method and the visualization algorithm provides an aesthetic and comprehensive presentation of information. Moreover, visualization through the decrease of data points within figures is a new idea. Previous visualization studies mainly focused on the absolute precision of the information in terms of operational time, aesthetics, and massive displays of data. However, by changing the perspective and using the appropriate visualization method, KDD99 visualization could be sufficiently accurate while providing less information loss. We believe that researchers focusing on big data visualization could be inspired by our work and produce more helpful big data visualization methods.

This study proposed a new sampling method, and the “normal” class could be visually identified because it possessed aspects of inner-class compactness and distinctness. Some improvements could yet be made in future work. The clusters of several attack types remain overlapped. Further study focusing on inter-class distinctness among those attack types could provide information and underlying relationship of those classes.

References

[1] L. Van Der Maaten, Accelerating t-sne using tree-based algorithms, J. Mach. Learn. Res. 15 (1) (2014) 3221–3245.

[2] X. Jin, B.W. Wah, X. Cheng, Y. Wang, Significance and challenges of big data research, Big Data Res. 2 (2) (2015) 59–64.

[3] S. Yu, M. Liu, W. Dou, X. Liu, S. Zhou, Networking for big data: a survey, IEEE Commun. Surv. Tutorials 19 (1) (2017) 531–549.

[4] C.P. Chen, C.-Y. Zhang, Data-intensive applications, challenges, techniques and technologies: a survey on big data, Inf. Sci. 275 (2014) 314–347.

[5] P. Zikopoulos, C. Eaton, et al., Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data, McGraw-Hill Osborne Media, 2011.

[6] G. Bello-Orgaz, J.J. Jung, D. Camacho, Social big data: recent achievements and new challenges, Inf. Fusion 28 (2016) 45–59.

[7] Z. Ruan, Y. Miao, L. Pan, Y. Xiang, J. Zhang, Big Network Traffic Data Visualization, Multimedia Tools and Applications Under Review.

[8] A. George, Anomaly detection based on machine learning: dimensionality reduction using pca and classification using svm, Int. J. Comput. Appl. 47(21).

[9] S.H. Hashem, Efficiency of svm and pca to enhance intrusion detection system, J. Asian Sci. Res. 3 (4) (2013) 381.

[10] O.I. Aladesote, A. Olutola, O. Olayemi, Feature or attribute extraction for intrusion detection system using gain ratio and principal component analysis (pca), Commun. Appl. Electron. 4 (3) (2016) 1–4. Published by Foundation of Computer Science (FCS), NY, USA.

[11] N.T. Tam, I. Song, Big data visualization, in: Information Science and Applications (ICISA) 2016, Springer, 2016, pp. 399–408.

[12] C. Ware, Information Visualization: Perception for Design, third ed., Elsevier, 2012.

[13] S. Nativi, P. Mazzetti, M. Santoro, F. Papeschi, M. Craglia, O. Ochiai, Big data challenges in building the global earth observation system of systems, Environ. Model. Softw. 68 (2015) 1–26.

[14] M. Chen, S. Mao, Y. Liu, Big data: a survey, Mob. Netw. Appl. 19 (2) (2014) 171–209.

[15] M. Zaharia, M. Chowdhury, T. Das, A. Dave, J. Ma, M. McCauley, M.J. Franklin, S. Shenker, I. Stoica, Resilient distributed datasets: a fault-tolerant abstraction for in-memory cluster computing, in: Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation, USENIX Association, 2012, pp. 2–2.

[16] E. Glatz, Visualizing host traffic through graphs, in: Proceedings of the Seventh International Symposium on Visualization for Cyber Security, ACM, 2010, pp. 58–63.

[17] E. Glatz, S. Mavromatidis, B. Ager, X. Dimitropoulos, Visualizing big network traffic data using frequent pattern mining and hypergraphs, Computing 96 (1) (2014) 27–38.

[18] L. Van Der Maaten, Barnes-hut-sne, arXiv preprint arXiv:1301.3342.

[19] M.H. Cynthia Brewer, T. P. S. University, Colorbrewer 2.0, 2013. http://colorbrewer2.org/#type=qualitative&scheme=Set1&n=5. (Accessed 4 July 2016).

[20] M. Tavallaee, E. Bagheri, W. Lu, A.A. Ghorbani, A detailed analysis of the kdd cup 99 data set, in: Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009. IEEE Symposium on, IEEE, 2009, pp. 1–6.

[21] T.U.K.A. Information, I. Computer Science University of California, 1999. Kdd cup 1999 data, https://archive.ics.uci.edu/ml/machine-learning-databases/kddcup99-mld/kddcup99.html. (Accessed 15 January 2017).