Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2020) 28: 1442 – 1457
© TÜBİTAK
doi:10.3906/elk-1907-84

Comparative analysis of classification techniques for network fault management

Mohammed MADI[1,*], **Fidaa JARGHON**[2], **Yousef FAZEA**[2], **Omar ALMOMANI**[3],
Adeeb SAAIDAH[3]

[1]Department of Computer Engineering, Hasan Kalyoncu University, Gaziantep, Turkey
[2]School of Computing, Universiti Utara Malaysia, Kedah, Malaysia
[3]Computer Network and Information Systems Department, The World Islamic Sciences & Education University, Amman, Jordan

Received: 12.07.2019 • **Accepted/Published Online:** 10.02.2020 • **Final Version:** 08.05.2020

Abstract: Network troubleshooting is a significant process. Many studies were conducted about it. The first step in the troubleshooting procedures is represented in collecting information. It's collected in order to identify the problems. Syslog messages which are sent by almost all network devices include a massive amount of data that concern the network problems. Based on several studies, it was found that analyzing syslog data (which) can be a guideline for network problems and their causes. The detection of network problems can become more efficient if the detected problems have been classified based on the network layers. Classifying syslog data requires identifying the syslog messages that describe the network problems for each layer. It also requires taking into account the formats of syslog for vendors' devices. The present study aimed to propose a method for classifying the syslog messages which identify the network problem.This classification is conducted based on the network layers. This method uses data mining instrument to classify the syslog messages. The description part of the syslog message was used for carrying out the classification process.The relevant syslog messages were identified. The features were then selected to train the classifiers. Six classification algorithms were learned; LibSVM, SMO, KNN, Naïve Bayes, J48, and Random Forest. A real data set was obtained from an educational network device. This dataset was used for the prediction stage. It was found that that LibSVM outperforms other classifiers in terms of the probability rate of the classified instances where it was in the range of 89.90%–32.80%. Furthermore, the validation results indicate that the probability rate of the correctly classified instances is >70%.

Key words: Classification algorithms, SVM, network fault management, machine learning, network management

1. Introduction

Most organizations rely on networks to manage their transactions. Any failure or error occurring in the network shall negatively affect the achievements, productivity, and service quality of organizations. The telecommunication networks complexity has been increasing. Thus, the management of networks has become more difficult, especially in terms of detecting and classifying network problems. Detecting and classifying network problems are considered essential things in the maintenance processes [1]. Therefore, it is necessary to diagnose and detect the reasons behind experiencing network failures and problems. These reasons must be identified to address such problems and decrease the probabilities of experiencing such problems in the future. Network troubleshooting can become an efficient process. That can be done if a systematic approach is adopted. Adopting such an approach shall reduce the confusion associated with the troubleshooting process and reduce

[*] Correspondence: mohammed.madi@hku.edu.tr

the time needed for carrying out this process. Diagnosis and recovery of any problem experienced by a system begins with reviewing the system's log files [2]. These files presents the activity record of the system. They present the sources of problems. Logging is a way of tracking down the historical events that can enable the administrators identify any failure, warning, error, success, or any type of problem occurring in the system. Syslog messages include a massive amount of data that concerns the network problems. This data is sent almost by all network types of equipment, such as the routers, switches, and firewalls [3]. Network troubleshooting is carried out through using the Layered Model [4]. Through this model, problems are normally described in terms of a specific model layer [4]. Network errors could be distributed into the network layers depending on the TCP/IP model (network access layer, Internet layer, transport layer, application layer) [5]. Therefore, network problems can be classified based on their causing layers, and the layer elements problems. Through carrying out the classification process, the intended layer that causes the network problems or failures can be detected and handled immediately. Detecting the problems is insufficient by itself to identify the source of the problem. In fact, the task of reading the detected syslog messages one by one must be carried out for understanding and identify the problem of the layer causing the network problems. The detection and classification of network problems requires incurring cost, devoting time and exerting effort [6]. These things are required to extract the data related to network problems, and classify this data based on the network layers. In fact, it is very difficult to classify syslog messages. That's attributed to the following reasons: (1) Having various types of logs which list messages with low or high severity [6]. In addition, the type of syslog data is represented in unstructured textual data, and there is a variety of formats and a big volume [7], (2) the increasing number of network elements which indicates that there's a massive volume of complex log data, and thus, it's necessary to extract information accurately and efficiently in order to make correct maintenance decisions, and (3) the log format, which depends on each vendor or service [8]. Thus, classifying syslog messages requires having deep domain knowledge of each log format, and the component of each layer and its potential problems. Classifying network problems based on network layers can enhance the maintenance decision, and the network troubleshooting. Applying machine learning techniques on syslog data shall enhance the capability of detecting syslog messages. These messages describe network problems, and classify them based on the related network layer. This study compares the predictive performance of six selected machine learning classifiers with each other. This comparison is conducted to detect the network fault and analyze the classification framework. This comparison is conducted to provide recommendations related to the model selection. The main contribution of this study lies in classifying syslog data based on the TCP/IP layers. This classification process aims at enhancing the processes of the network troubleshooting, and raising the efficiency of the maintenance processes. The rest of this paper is structured as follows: Section II presents an overview about the works related to network fault management. Section III presents an overview of five machine learning classifiers deployed in the proposed model. Section IV presents the results of experiments and validation of the proposed framework. Finally, Section VI presents the conclusion and recommendations for future work

2. Related works

Many significant studies were conducted about the network detection issues from log data using data mining techniques [9, 10]. Wei et al. [8] proposed a system for the detection of runtime problems by mining console logs. The latter researchers converted free-text console logs into numerical features. These features were analyzed through using Principal Component Analysis (PCA). A similar approach was adopted by Liu and Jiangang [11]. Those researchers used PCA learning algorithm for detecting the anomalies existing in syslog messages.

They created features which capture various correlations among different types of log messages. On the hand, Tongqing et al. [12] developed an automated tool syslog digest that transforms the massive volume of routers syslog messages into a smaller number of meaningful network events. Then, they identified the signature of syslog messages that capture the network behavior over a period of time. They grouped these messages based on their nature and severity. Fukuda [13] used syslog messages to detect the unusual events occurring in a network. That was done through assigning a global weight, based on a global appearance of a message type in the whole data set. Kimura et al. [6] analyzed two types of data: SNS data from tweeter and syslog messages. That was done to detect and diagnose any network failure. The latter researchers used non-negative matrix factorization (NMF) machine learning algorithm in order to analyze syslog messages. They supported the vector machine in order to analyze tweeter messages.

All of the aforementioned studies aimed to detect network problems only. However, they did not classify the problems in the aim of increasing the efficiency of the maintenance and troubleshooting problems. The present study aimed to propose methods to analyze syslog data, detect network problems and diagnose their causes, and classify these problems in terms of network layers.

3. Fault classification techniques

Machine learning technique, and supervised learning techniques are used for text classification. Through using these techniques, the class labels shall be defined early [14, 15]. The algorithms for the text classification that are considered popular the most include [16, 17]: Support Vector Machines (SVM) [17], Naive Bayes (NB), K-Nearest Neighbour (k-NN) [18], and Decision Tree algorithms [19].

3.1. Support vector machine

The support vector machine is a classifier. It was proposed by Bottou and Vapnik [17]. It is a supervised learning technique that is applicable to both classification and regression, by applying linear classification techniques to non-linear data [20]. SVM is a machine learning technique. It adopts the structural risk minimization principle. The SVM splits the data of training set into two classes. It makes a decision depending on the ”Support Vectors”. Through it, the effective elements are selected from the training set [21]. Two main method have been proposed for meeting this purpose; one-against-one method [22], one-against-all method [23].

One-Against-One In this scheme, one SVM classifier, SVMi,j , is constructed for every pair of classes (i,j). There are N(N +1)/2 SVM classifiers in total [24]. In this study four classes are considered, therefore there are 4(4 +1)/2 SVM classifiers, that means there are ten SVM classifiers.

One-Against-All It constructs k SVM models where k is the number of classes, this study has four classes (models). The ith SVM is trained with all of the examples in the ith class with positive labels, and all other examples with negative labels. Library for Support Vector Machines (LibSVM) is an algorithm that apply the concepts of one-against-all method.

3.2. K-nearest neighbor KNN

The KNN is one of the simplest supervised machine learning algorithm. It is the algorithm used the most for classification. KNN classifies a data point based on the way in which its neighbors are classified. It stores all the available cases and classifies the new cases based on a similarity measure [25]. KNN stores all the available cases and classifies new cases based on similarity measure (i.e. the distance function). For that reason, the distance between objects from the training set is computed in order to be used by the KNN classifier for selecting objects from the training set [25].

In fact, KNN is an algorithm that's based on feature similarity. In K-NN, examples are classified based on the class of their nearest neighbors. That means that the intuition underlying Nearest Neighbour classification is quite straightforward. In general, a case is classified based on the majority of its neighbors, with having a new case being assigned to the class that is the most common amongst its K nearest neighbors measured by the distance function [26]. It should be noted that the KNN is known as Lazy Learning, and the induction process is delayed to run time. Since the training sets must be in the memory at run-time, it is also denoted as Memory-Based Classification [27].

3.3. Decision trees

Decision Tree (DT) or recursive partitioning model refers to a decision support tool. It uses a tree-like graph of decisions and their possible consequences. DT creates a type of flowchart which consists of nodes (called "leafs") and a set of decisions to be made based on a node (called "branches"). The leaf and the branch structure forms a hierarchical representation that mimics the form of a tree. Decision tree learning is one of the most widely used methods for carrying out the inductive inference [28]. It is an important tool for carrying out the predictive analysis process. The main problem in the DT classification algorithm is represented in the way of constructing the optimal classifier [29]. In general, the DT classifier can be built from a set of features. In the classification task, the size of search space is exponential and the accuracy of some trees is more precise than other trees.

3.4. J48 algorithm

J48 is an algorithm used to generate a decision tree that is generated by Quinlan's C4.5 [30]. C4.5 algorithm is used for building a decision tree. It is an extension of the ID3 algorithm that uses a predictive machine-learning model. J48 generates rules that identify the identity of the data. For gaining the equilibrium of flexibility and accuracy, the decision tree is progressively generalized.

3.5. Random forests

Random Forest (RF) is an ensemble classifier that uses many decision tree models, ensemble models combine the result of different classifier. The result resulting from the ensemble model is usually different from one resulting from the individual model [31]. RF employs the DT concept by producing a large number of decision trees. The approach first selects a random sample of the data and identifies a key set of features in order to grow each decision tree. Then, the decision trees have their own error rate determined. After that, the collection of decision trees are compared to find the joint set of variables that produce the strongest classification model. In order to make the prediction of the class label of a case y through majority voting, the equation below shall be used [31]: $I(y) = argmax_c(\sum_n^N T_{h_n}(y) = c)$ Where I the indicator function and h_n the nth tree of the RF.

3.6. Naïve bayes algorithm

Naïve Bayes classifier is a simple probabilistic classifier that relies on bayes. Theorem with independence assumptions between predictors is particularly suited when the dimensionally of the inputs is high. It is mainly used in case the categorical input variables are compared to numerical variables. In other words, Naïve Bayes operates by assuming independence (i.e. the presence of some feature will not affect the other features [32, 33].

"Naïve Bayes" classifier is considered as a "supervised learning" algorithm. It relies on the probability model. In many practical applications, the method of maximum likelihood is used in parameter estimation for "Naïve Bayes" models [34]. Naïve Bayes model is easy to build with no complicated iterative parameter estimation. That makes it useful for the large data sets. Despite its simplicity, the Naive Bayes model operates surprisingly well. It is widely used because it often outperforms the classification methods that are more sophisticated.

Naïve Bayes Algorithm is a statistical classifier that predicts class membership probabilities, such as: the probability that a given tuple belongs to a particular class. It uses the prior probability of each category given that here is no information provided about the item. As a result, categorization produces a posterior probability distribution over the possible categories.

4. Implementing fault classification algorithms

4.1. Data set

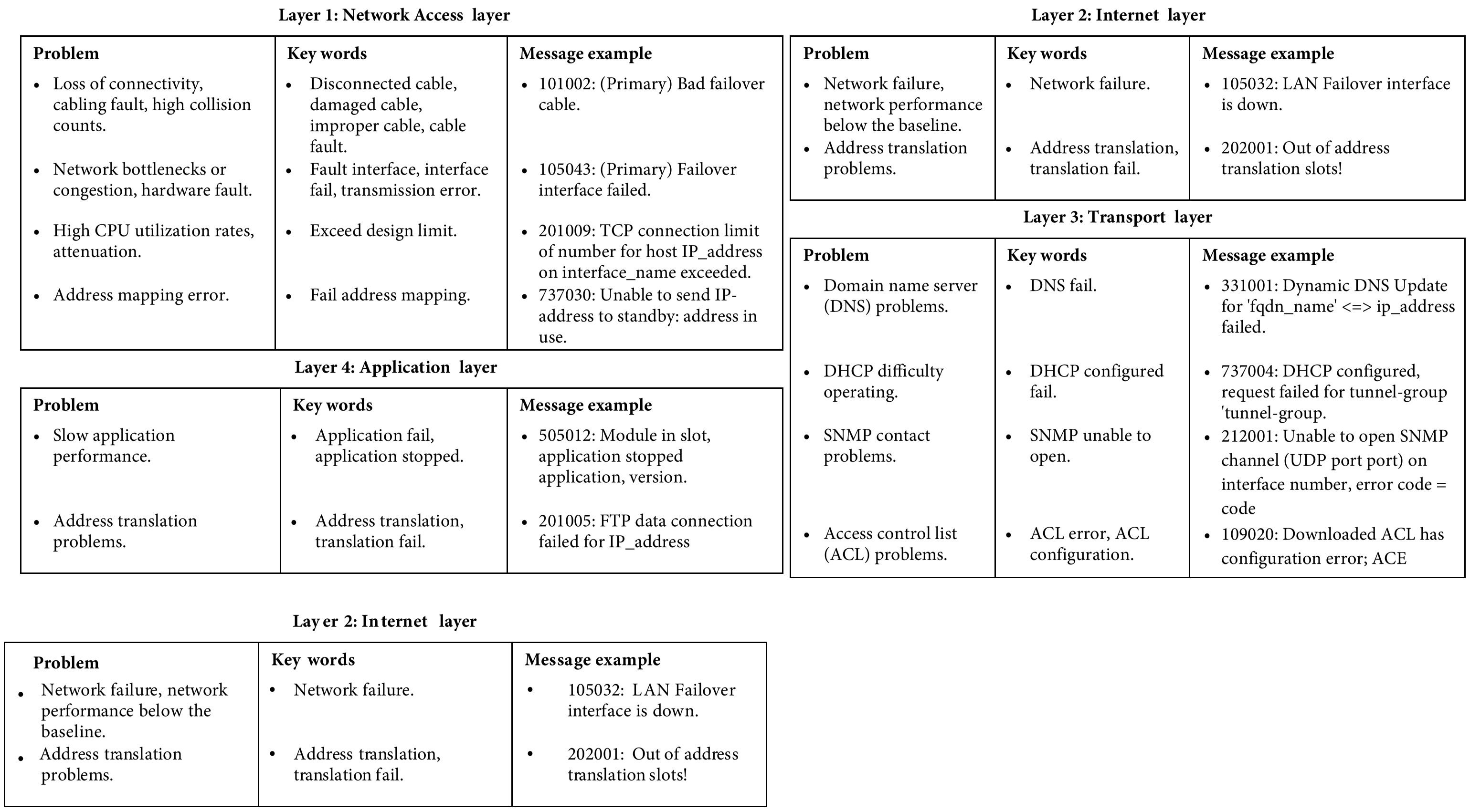
Cisco syslog manual was used to identify related syslog messages and the result could be applied to other syslog data from different vendors, as all vendors describe network problems using almost same terms and vocabularies. Cisco IOS devices have more than 500 facilities represented by integers. Facility simply is a service provides classification of the sources that generate syslog messages. The most common facilities are: IP, Open Shorted Path First (OSPF), SYS Operating System, IP security (IPsec), Rout Switch Processor (RSP), Interface (IF) [3]. The related messages are identified depending on the symptoms and causes of network problems related to the network layer. The task of identifying related messages from the manual requires reading and searching the manual in order to extract them. Searching in the manual is done by using the symptoms and the causes of each problem as key words to identify related problems. Loss of connectivity, performance lower than baseline, high collision counts, attenuation, bad cable, disconnected cables, damaged cables, improper cable types, cable length exceeds the design limit for the media, and cable fails are the symptoms and causes of cable problems that could be used as key words for searching in the manual, which include error message, its explanation, and the recommended action. The following are the extracted messages that describe cable problems from the Cisco syslog manual:

```
%PIX|ASA-1-101001: (Primary) Failover cable OK.
%PIX|ASA-1-101002: (Primary) Bad failover cable.
%PIX|ASA-1-101003: (Primary) Failover cable not connected (this unit).
%PIX|ASA-1-101004: (Primary) Failover cable not connected (other unit).
%PIX|ASA-1-101005: (Primary) Error reading failover cable status.
```

The goal is to identify syslog messages that describe network problems related to each network layer. Network problems were identified for each layer and key words that described each problem were extracted to be used for searching in Cisco syslog manual. Figure 1 illustrates problems of each network layer, the key words used for searching in Cisco syslog manual, and examples of extracted syslog messages.

4.2. Classification phases

The classification process is an attempt made in order to label a set of unlabeled conditions (i.e. assign them to classes). That is done based on a set of conditions along with using known labels. The classification approach that is commonly adopted involves two stages as shown in Figure 2:



Layer 1: Network Access layer

Problem	Key words	Message example
• Loss of connectivity, cabling fault, high collision counts.	• Disconnected cable, damaged cable, improper cable, cable fault.	• 101002: (Primary) Bad failover cable.
• Network bottlenecks or congestion, hardware fault.	• Fault interface, interface fail, transmission error.	• 105043: (Primary) Failover interface failed.
• High CPU utilization rates, attenuation.	• Exceed design limit.	• 201009: TCP connection limit of number for host IP_address on interface_name exceeded.
• Address mapping error.	• Fail address mapping.	• 737030: Unable to send IP-address to standby: address in use.

Layer 2: Internet layer

Problem	Key words	Message example
• Network failure, network performance below the baseline.	• Network failure.	• 105032: LAN Failover interface is down.
• Address translation problems.	• Address translation, translation fail.	• 202001: Out of address translation slots!

Layer 3: Transport layer

Problem	Key words	Message example
• Domain name server (DNS) problems.	• DNS fail.	• 331001: Dynamic DNS Update for 'fqdn_name' <=> ip_address failed.
• DHCP difficulty operating.	• DHCP configured fail.	• 737004: DHCP configured, request failed for tunnel-group 'tunnel-group.
• SNMP contact problems.	• SNMP unable to open.	• 212001: Unable to open SNMP channel (UDP port port) on interface number, error code = code
• Access control list (ACL) problems.	• ACL error, ACL configuration.	• 109020: Downloaded ACL has configuration error; ACE

Layer 4: Application layer

Problem	Key words	Message example
• Slow application performance.	• Application fail, application stopped.	• 505012: Module in slot, application stopped application, version.
• Address translation problems.	• Address translation, translation fail.	• 201005: FTP data connection failed for IP_address

Layer 2: Internet layer

Problem	Key words	Message example
• Network failure, network performance below the baseline.	• Network failure.	• 105032: LAN Failover interface is down.
• Address translation problems.	• Address translation, translation fail.	• 202001: Out of address translation slots!

Figure 1. Summary of syslog network problems, key words, and message example.

1) Training the classifier through using the pre-labeled set of conditions, and using a selected subset of probes which are considered as the selected features.

2) Using the trained classifier in order to label other conditions.

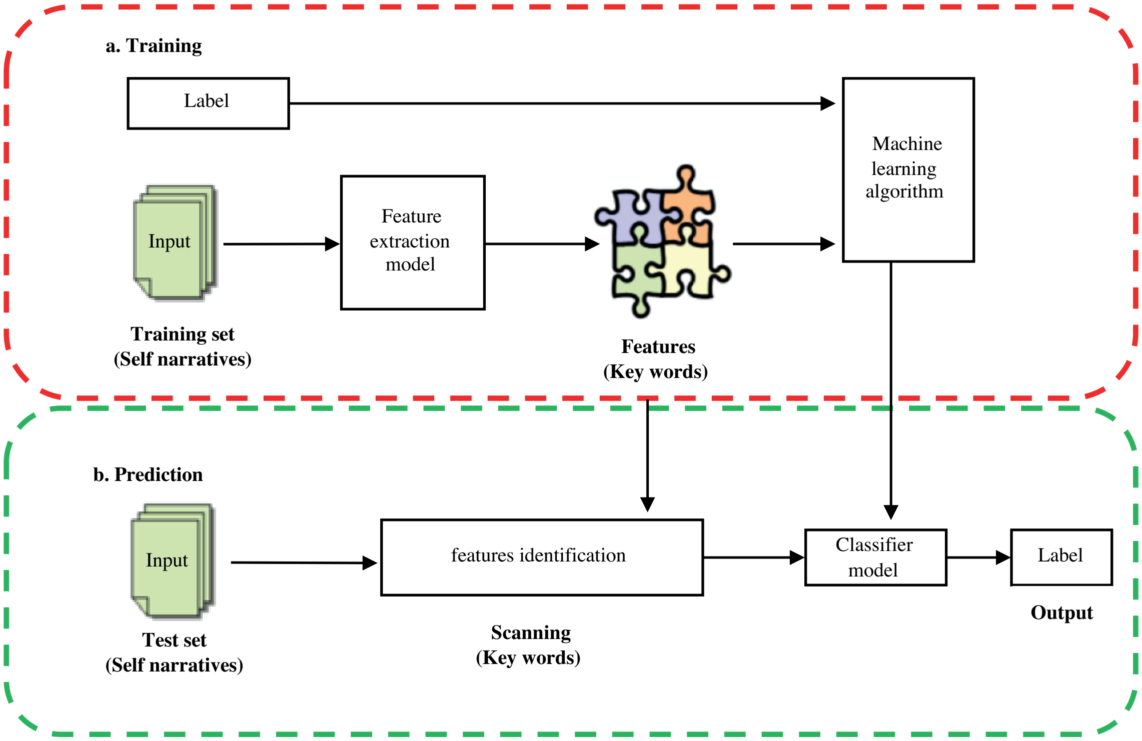


Figure 2. Classification phase [35].

Regarding the training stage, the files of training dataset along with a different number of features were prepared. The training dataset is derived from the syslog messages which were extracted from the Cisco syslog manual during phase two and processed (cleaned, removed stop words, stemmed, removed duplicated words). The training dataset was in ARFF extension, to be used in the Weka data mining tool. The classification algorithm was applied several times through using the same dataset. With different number of features. The obtained results were compared in order to identify the best feature and identify the best classification model. First, the algorithm was applied to four training data files with a different number of features. The first training file represents the data the employs all the features (terms of vector space). The second training file represents the data that employs 1000 features which have the highest frequency. The third training file represents the data using 500 features which have the highest frequency. The fourth training file represents the data using 200 features which have the highest frequency.

The accuracy rate was calculated. In addition, the results were compared to identify the best number of features. That was done to use these features for representing syslog data files. The accuracy rate indicates that the ratio of correctly classified instances; and the performance of the classifier model would be better when reaching a high accuracy rate. The accuracy rate was calculated through using the following equation:. $accuracy\,rate = n \ /N$ Where n is the number of correctly classified instances, and N is the number of all classified instances.

The results showed that the performance of six classifiers scored the highest values by using a training file that was represented with 500 features, Figure 3 illustrates the process of training stage. They also showed that four classifiers: libSVM, RF. SMO and J48 had recorded performance value higher than NB, and KNN.

The latter four algorithms (i.e. libSVM, RF. SMO and J48) were relearned by another two training files with different features number. The relearning process used 1 file which represents data that uses 550 features. It also used another file which represents data using 450 features. The relearning process aims to make sure that using 500 features is the best number of features to be used for representing datasets. The results of the four classification algorithms were compared in terms of the accuracy rate, and performance through using a training file that was represented with 500 features.

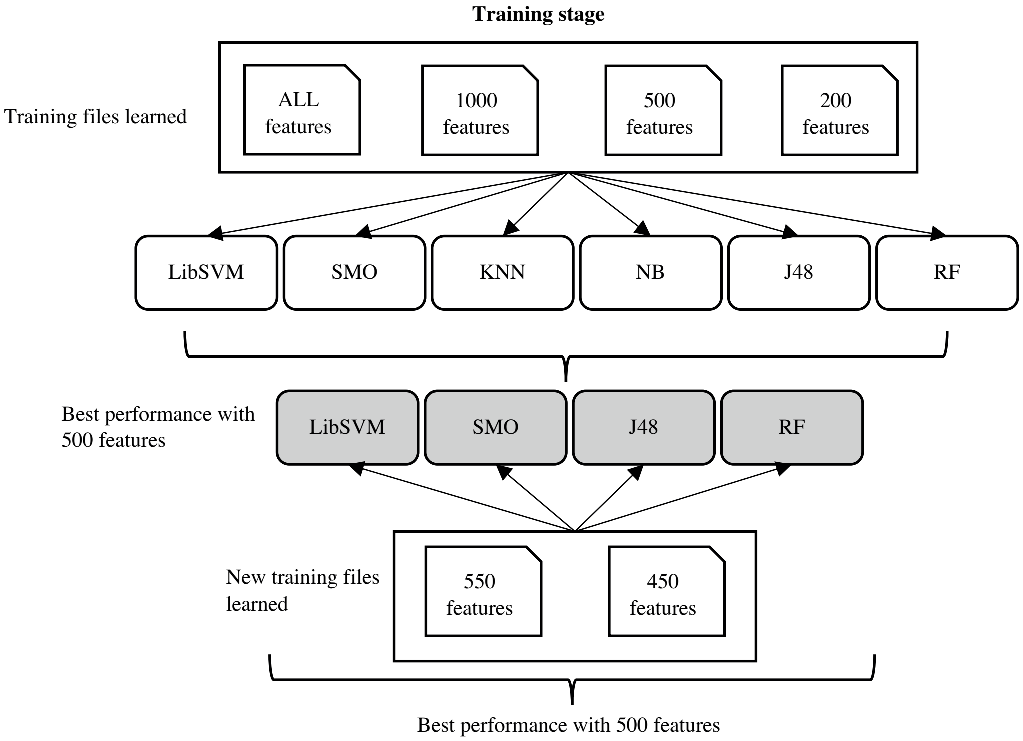


Figure 3. Process of training stage.

Regarding the prediction stage, four classification algorithms based on the training stage. These algorithms are: libSVM, RF. SMO, and J48. They were used as classifiers to classify a testing dataset in the prediction stage. These algorithms show a good performance during the training stage. Testing dataset, obtained from UUM network devices, was processed and represented with 500 features in ARFF files; this 500 of features is the best number to be used as shown in the results of training in Figure 4. Classifiers results were compared in terms of probability rate. It was found that LibSVM is the best model to be used for syslog data classification. Probability rate indicates the proportion of accuracy that the classified instance relays to the specific class. Probability rate for each classified instance is displayed in the table of classification results. It is calculated by Weka.

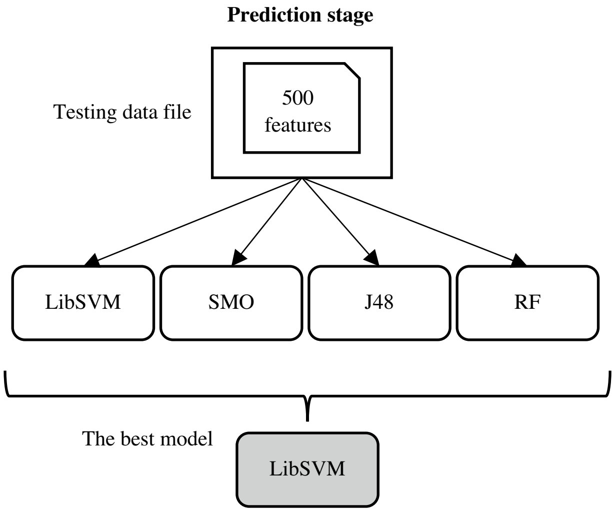


Figure 4. Process of prediction stage.

5. Results

5.1. Results of training stage

Training data set files with all features (i.e. the one with 1000 features, the one with 500 features and the one with 200 features) were generated to be used during the training stage. They were used to evaluate the performance of text classification algorithms. The obtained results were compared for identifying the best number of features and the best models. Training dataset which contains 263 instances (syslog message) was used during the training stage.

Six classification algorithms were applied to the above training dataset files and performance rate (i.e. the accuracy rate). The latter rate represents the ratio of the instances that are correctly classified for each classifier. The results of the performance of the six algorithms through using above training datasets are presented below:

As shown in Table 1, the performance values recorded through the training file indicate that the 500 features show the highest values. That's because the number of the correctly classified instances are higher than the values of other files. That means that this file includes the best number of features that can be used for the prediction stage. As shown in Figure 5, it should be noted that the accuracy rate of Random Forest, LibSVM, SMO, and J48 show the highest values. These algorithms classified instances correctly and in a manner that is better than the other algorithms Two training dataset files were generated. One of them includes 450 features and the other one includes 550 features. These files were used for making sure that the file including 500 features include the best features to be used during the prediction stage. Four classification algorithms: Random Forest, LibSVM, SMO, and J48 were applied to the new two training dataset. It should be noted that the performance level was recorded. That was done through calculating their accuracy rate. The results were compared with the results of the previous training dataset that includes 500 features. Table 2 displays the accuracy rate of each classifier of the four ones. That was done through using training files represented by different numbers of features.

As shown in Table 2, the accuracy rate of the algorithms show fewer values when using training dataset files of 550 features and 450 features. That means that the training dataset file including 500 features include the best features to be used during the prediction stage. It is clearly seen by Figure 6, the results of the experiments

Table 1. Accuracy rate of six classifiers using training files represented by different numbers of features.

Training file	SVM(Lib)	SVM(SMO)	Naïve bayes	KNN	J48	Random forest
All features	73.80%	71.10%	41.10%	64.30%	64.30%	70.70%
1000 features	74.10%	71.10%	52.90%	64.30%	64.30%	73.00%
500 features	74.50%	71.10%	60.10%	64.30%	69.30%	75.70%
200 features	65.40%	65.40%	55.50%	49.80%	58.20%	67.70%

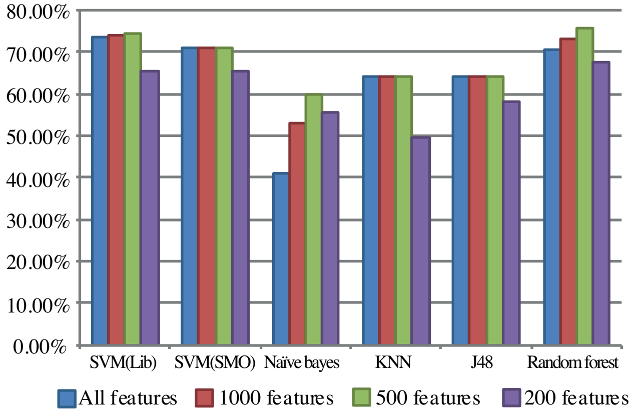


Figure 5. The comparison between LibSVM, SVM(SMO), NB,KNN, J48, RF algorithms in terms of accuracy rate.

Table 2. Accuracy rate of four classifiers using training files represented by different numbers of features.

Training file	SVM(Lib)	SVM(SMO)	Random forest	J48
550 features	71.10%	70.30%	71.10%	64.30%
500 features	74.50%	71.10%	75.70%	64.30%
450 features	71.10%	70.40%	71.10%	64.30%

conducted during the training stage indicate that the features of number 500, had the best performance by applying Random Forest, LibSVM, SMO, and J48 classification algorithms.

5.2. Results of prediction stage

In the prediction stage, four classification algorithms: Random Forest, LibSVM, SMO, and J48 were applied to the testing dataset. Classification algorithms were chosen for good performance during the training stage. An obtained testing dataset consists of 2610 instances (syslog message) from firewalls and switches devices involving a short period of time (less than one minute). The testing dataset was preprocessed similar to training dataset (cleaned, removed stop words, stemmed, removed duplicated words). Testing data set file was generated with 500 features in ARFF format to be classified using the Weka data mining tool. The results of the comparison between the four classification algorithms-Random Forest, LibSVM, SMO, and J48-involving testing dataset are presented in Table 3.

Table 3 shows the number of instances, classified into each layer with the percentage of all testing dataset sample. Four algorithms Random Forest, LibSVM, SMO, and J48 gave convergent results; LibSVM, SMO, and

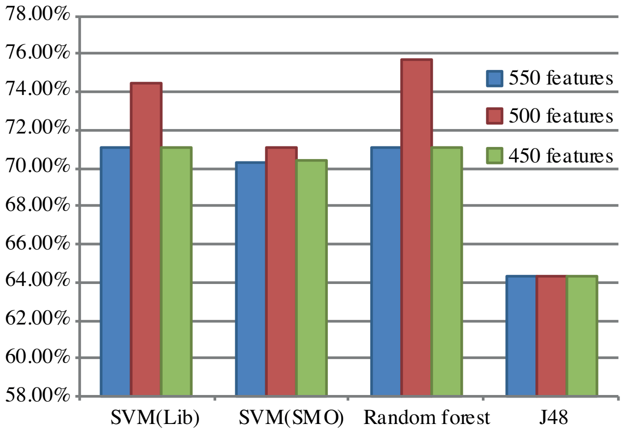


Figure 6. The comparison between dataset files in terms of algorithms accuracy rate.

Table 3. Prediction stage results for four classifiers.

Algorithm	layer1	layer2	layer3	layer4
LibSVM	172 (6.59%)	2383(91.30%)	55 (2.11%)	0 (0%)
SVM (SMO)	134 (5.13%)	2351 (90.08%)	126 (4.83%)	0 (0%)
J48	250 (9.58%)	2320 (88.89%)	40 (1.53%)	0 (0%)
RF	170 (6.51%)	2395 (91.76%)	43 (1.65%)	2 (0.08%)

J48 algorithms classified all instances into three classes (layer1, layer2, and layer3), while RF algorithm classified all instances into four classes (layer1, layer2, layer3, and layer4), layer4 got only two instances, as shown in Figure 7. In fact, the obtained results comply with the fact saying that lower layer of TCP/IP model are where the most of issues occur. Therefore, it is advisable in network management process, to start troubleshooting from the physical layer and gradually proceed to the upper layers.

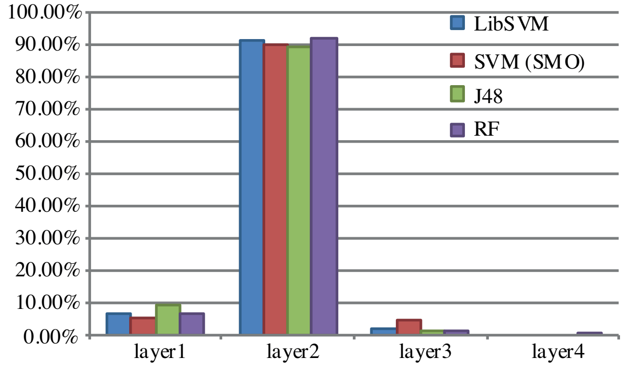


Figure 7. The comparison between four classifiers in terms of prediction stage results.

Syslog data contains information of all network events with various types and severity levels and almost all syslog messages are either informational or problem messages. Since classification algorithms classify all instances of the testing dataset into specific classes, the results need to be analyzed deeply in terms of the probability rate of each classified instance, as a way to identify informational instances and problems ones. Probability rate indicates to the proportion of accuracy that the classified instance relays to the specific class. The following table compares the range of probability rate of classified instances for results of Random Forest, LibSVM, SMO, and J48.

As shown in Table 4, the results of LibSVM algorithm show probability rates higher than others classifiers. LibSVM gave higher values for both maximum and minimum rate in each layer range, except for the first layer range whose maximum value is less than RF. Depending on the previous results, the results of LibSVM algorithm is considered as the best. The result is divided into two parts: one for the lower probability to be validated as informational messages, and the other for the higher probability to be validated as problems messages. Each part is compared to the training dataset to validate the result. The following table shows the numbers of instances and their percentage with a probability rate >= 50% and < 50% for each class.

Table 4. Probability range of classified instances for used classifiers.

Algorithm	layer1	layer2	layer3	layer4
LibSVM	(72.20 - 32.80)%	(67.00 – 33.00)%	(89.90 - 36.20)%	0.00%
SVM (SMO)	(50.00 - 40.30)%	(55.00 - 40.3)%	(50.00 - 40.3)%	0.00%
J48	(60.50 - 45.30)%	(50.00 - 40.00)%	(50.30 - 40.30)	0.00%
RF	(79.00 - 26.00)%	(78.00 - 34.00)%	(58.00 - 36.00)%	38.00%

As shown in Table 5 and Figure 8, layer1, and layer3 have small numbers of instances with probability rate with probability rate >= 50%. However, layer2 have large numbers of instances with a probability rate < 50%. This is because of repeated messages for the same problem with one different word.

Table 5. Probability rate for classified instances using LibSVM

Probability	layer1	layer2
$\geq$ 50%	49 (1.88%)	2218 (84.98%)
< 50%	123 (4.71%)	165 (6.32%)

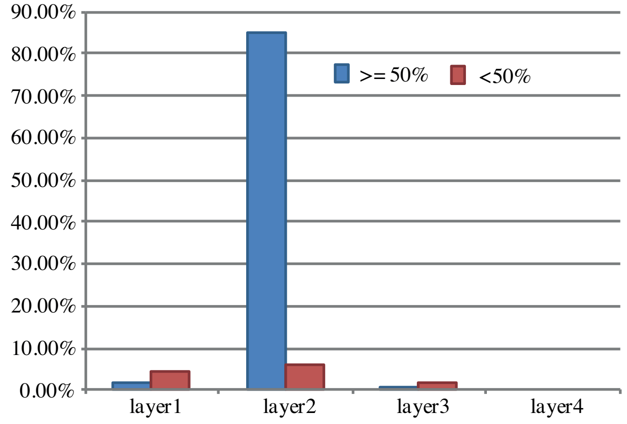


Figure 8. Classified Instances in Terms of Probability Rates for Each Class

5.3. Results of validation phase

Based on the results of the training stage, the classification algorithms that show good performance are: libSVM, RF SMO, and J48. These four algorithms were used during the prediction stage. They were applied to the testing dataset. During this phase, the results of the prediction stage were validated. The validation process was carried out through comparing the instances of each class to the corresponding training dataset. During the validation phase, the results of the prediction stage were analyzed. That was done to make sure that each instance belongs to its class, and refers to a problem in one layer.

LibSVM classifier has classified all instances of testing dataset into three layers- (layer1, layer2, layer3). As mentioned above, the classified instances for each layer were divided into two parts based on the prediction probability. The instances with probability rate $>= 50\%$ were validated through comparing them to the corresponding training dataset. However, the instances with a probability rate $< 50\%$ were validated as informational messages.

5.4. Layer 1 validation

The classifier had classified 49 instances with a probability rate $>= 50\%$, to layer1. These instances were compared to the instances that belong to class one in training dataset. Only three instances indicated that there is a network problem. The probability rate of these instances is $> 70\%$. These three messages described the problem of “TCP connection to firewall server had been lost, restricted tunnels are now allowed full network access”. That was repeated three times. By referring to syslog manual, this problem indicate that the TCP connection to the security appliance server was lost. Thus, that requires checking the server and network connections. The latter problem belongs to the first layer. This layer is a network access layer.

5.5. Layer 2 validation

The classifier classified 2218 instances with a probability rate of $>= 50\%$, to layer2. These instances were compared with the instances belonging to class two in the training dataset. No instance in the classified instances indicated that there is a network problem. The probability rate of them is $< 70\%$.

5.6. Layer 3 validation

The classifier had classified 8 instances with a probability rate that is $>= 50\%$, to layer3. TThese instances were compared to the instances that belong to class three in the training dataset. One instance among the all classified instances indicated that there is a network problem. The probability rate is $> 70\%$. This message described the problem of “No translation group found for protocol src”. Through referring to the syslog manual, the latter problem is attributed to the fact that the network address translation (NAT) wasn’t configured for the specified source nor the destination systems. This problem pointed to NAT issues that belong to the third layer(i.e. the transport layer).

5.7. Validation of instances with low probability

The instances that have a probability validation that’s less than 50% were compared to the training dataset. There isn’t any instance pointing to the network problems. It described network events only. The result of the validation process seems to be acceptable. The testing data sample was for a short period of time, less than one minute.

6. Conclusion and future work

Network fault detection and identification is essential for making well-informed decisions by network administrators. The present paper proposes a method for detecting and classifying network problems, in terms of network layers. That was done through analyzing syslog data. To meet the study's goals, the researchers conducted a comparison of five different machine learning classifiers. This comparison was conducted for detecting and classifying network problems. The researchers compared classifiers based on their accuracy and error rates.

It was found that (Random Forest, LibSVM, SMO, and J48) show a good performance during the training stage. These algorithms show the following rate respectively of correctly classified instances: 75.70%, 74.50%, 71.10%, and 69.30%. LibSVM algorithm classified instances with a probability rate that is higher than the counterpart rate of RF, SMO, and J48 classifiers. The probability rates of classified instances through using LibSVM are in the range of 89.90% - 32.80%. The validation results show that the probability rate of the correctly classified instances is $> 70\%$. The results of the comparison between the algorithms indicate that the SVM classifiers show the best performance during the experiments of the present study.

References

[1] Bob V. Accessing the WAN, CCNA Exploration Companion Guide. India: Pearson Education, 2008.

[2] Son HS, Lee JH, Kim TY, Lee SG. Network traffic and security event collecting system. In: Proceedings of Second International Conference on Electrical Systems, Technology and Information; Singapore; 2015. pp 439-446.

[3] Deveriya A. Network administrators Survival Guide. USA: Cisco Press, 2005.

[4] Joseph S. Network Troubleshooting Tools: Help for Network Administrators. USA: O'Reilly Media, Inc., 2001.

[5] Wilkins, Sean. Designing for Cisco Internetwork Solutions (DESGN) Foundation Learning Guide: (CCDA DESGN 640-864). Pearson Education, 2011.

[6] Kimura T, Takeshita K, Toyono T, Yokota M, Nishimatsu K et al. Network Failure Syslog and SNS. International Journal of Pure and Applied Mathematics 2018; 119 (12): 9543-9551.

[7] Viktor M, Kenneth C. Big Data: A Revolution That Will Transform How We Live, Work, and Think. Boston, New York: Eamon Dolan/Mariner Books Houghton Mıın Harcourt, 2013.

[8] Wei X, Ling H, Armando F, David P, Michael J. Detecting large-scale system problems by mining console logs. In: Proceedings of the ACM SIGOPS 22nd Symposium on Operating Systems Principles; Montana, USA; 2009. pp. 117-132.

[9] David SK, Saeb AT, Al Rubeaan K. Comparative analysis of data mining tools and classification techniques using weka in medical bioinformatics. Computer Engineering and Intelligent Systems 2013; 4 (13): 28-38.

[10] Zupan, Blaz, Janez D. Open-source tools for data mining. Clinics in laboratory medicine 2008; 28 (1): 37-54. doi: 10.1016/j.cll.2007.10.002

[11] Liu M, Jiangang Y. An improvement of TFIDF weighting in text categorization. In: International Proceedings of Computer Science and Information Technology; Singapore; 2012. pp. 44-47.

[12] Tongqing Q, Zihui G, Dan P, Jia W, Jun Xu. What happened in my network: mining network events from router syslogs. In: Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement; Melbourne, Australia; 2010. pp. 472-484.

[13] Fukuda, Kensuke. On the use of weighted syslog time series for anomaly detection. In 12th IFIP/IEEE International Symposium on Integrated Network Management and Workshops; Dublin, Ireland; 2011. pp. 393-398.

[14] Genkin A, David D, David M. Large-scale Bayesian logistic regression for text categorization. Technometrics 2007; 49 (3): 291-304. doi: 10.1198/004017007000000245

[15] Guiying W, Xuedong G, Sen W. Study of text classification methods for data sets with huge features. In 2nd International Conference on Industrial and Information Systems; Dalian, China; 2010. pp. 387-406.

[16] Pilászy I. Text categorization and support vector machines. In: Proceedings of the 6th International Symposium of Hungarian Researchers on Computational Intelligence; Budapest, Hungary; 2005. pp. 1-10.

[17] Bottou L, Vapnik V. Local learning algorithms. Neural computation 1992; 4 (6): 888-900. doi: 10.1162/neco.1992.4.6.888

[18] Kim JW, Lee BH, Shaw MJ, Chang HL, Nelson M. Application of decision-tree induction techniques to personalized advertisements on internet storefronts. International Journal of Electronic Commerce 2001; 5 (3): 45-62. doi: 10.1080/10864415.2001.11044215

[19] Murty, MR, Murthy, JVR, Reddy et al. A Survey of cross-domain text categorization techniques. In 1st IEEE International Conference on Recent Advances in Information Technology; Dhanbad, India; 2012. pp. 499-504.

[20] He, Ji, Ah-Hwee T, Chew L. A Comparative Study on Chinese Text Categorization Methods. In: PRICAI Workshop on Text and Web Mining; Melbourne, Australia; 2000. pp. 1-12.

[21] Chiang H , Wang T. One-against-one fuzzy support vector machine text categorization classifier. In: IEEE International Conference on Industrial Engineering and Engineering Management; IEEE Singapore; 2008. pp. 1519-1523.

[22] Liu Y , Zheng YF. One-against-all multi-class SVM classification using reliability measures. In: International Joint Conference on Neural Networks; Montreal, Canada; 2005. pp. 849-854.

[23] Hsu, Chih-Wei, Chih-Jen Lin. A comparison of methods for multiclass support vector machines. IEEE transactions on Neural Networks 2002; 13 (2): 415-425. doi:10.1109/72.991427

[24] Pawar , Gawande S. A Comparative Study on Different Types of Approaches to Text Categorization. Machine Learning and Computing 2012; 2(4) 423-426. doi:10.7763/IJMLC.2012.V2.158

[25] Cunningham P, Sarah D. k-Nearest neighbour classifiers. In: Multiple Classifier Systems; Prague, Czech Republic; 2007. pp.1-17.

[26] Suha O. Enhanced ntology-based text classification algorithm for structurally organized documents. PhD, Universiti Utara Malaysia, Kedah, Malaysia, 2015.

[27] Liao Y, Vemuri VR. Use of k-nearest neighbor classifier for intrusion detection. Computers and Security 2002; 21 (5): 439-448. doi: 10.1016/S0167-4048(02)00514-X

[28] Kaur, Gaganjot, Amit C. Improved J48 classification algorithm for the prediction of diabetes. International Journal of Computer Applications 2014; 98 (22): 13–17. doi:10.5120/17314-7433

[29] Faheema AG, Subrata Ra. Feature selection using bag-of-visual-words representation. In IEEE 2nd International Advance Computing Conference; Patiala, India; 2010. pp 151-156.

[30] Quinlan J. Induction of decision trees. Machine learning 1986; 1 (1): 81-106. doi:doi.org/10.1007/BF00116251

[31] Liparas D, HaCohen-Kerner Y, Moumtzidou A, Vrochidis S, Kompatsiaris I. News articles classification using random forests and weighted multimodal features. In: Information Retrieval Facility Conference; Copenhagen, Denmark; 2014. pp. 63-75.

[32] Korada N, Kumar NSP, Deekshitulu. Implementation of naïve Bayesian classifier. International Journal of Information Sciences and Techniques 2012; 2 (3): 63-75. doi: 10.5121/ijist.2012.2305

[33] Frank E, Bouckaert R. Naive bayes for text classification with unbalanced classes. In: Proceedings of the 10th European Conference on Principle and Practice of Knowledge Discovery in Databases; Springer, Berlin, Heidelberg; 2006. pp. 503-510.

[34] Dietterich, Thomas G. Ensemble methods in machine learning. In: multiple classifier systems, Lecture Notes in Computer Science; Berlin, Heidelberg; 2000. pp. 1-15.

[35] He Q, Veldkamp BP. Classifying unstructed textual data using the Product Score Model: an alternative text mining algorithm. In: Eggen TJHM, Veldkamp BP (editor). Psychometrics in practice at RCEC. Netherlands: RCEC, Cito, University of Twente. 2012, pp. 47-62.