Journal of Computer Science and Electrical Engineering
ISSN: 2663-1946
DOI: https://doi.org/10.61784/jcsee3050

NETWORK INTRUSION DETECTION METHODS BASED ON MACHINE LEARNING

JingYa Sun[1*], ZiJie Cao[2]
[1]Mathematics and Physics Teaching Department, Hebei GEO University, Shijiazhuang 050030, Hebei, China.
[2]School of Information Engineering, College of Science & Technology Ningbo University, Ningbo 315300, Zhejiang, China.
Corresponding Author: JingYa Sun, Email: sunjingya2021@163.com

Abstract: With the rapid development of Internet technology, traditional intrusion detection methods have limitations. Although machine learning technology provides new ideas for intrusion detection, its efficiency and accuracy in large-scale data environment still need to be optimized. This study aims to propose an intrusion detection method that combines feature selection with optimized machine learning algorithms to solve the problems of data redundancy and category imbalance, and to reduce the false alarm rate. Based on the UNSW-NB15 dataset, ANOVA, chi-square test and Gini coefficient are used for feature selection, combined with principal component analysis (PCA) dimensionality reduction technique. The model is constructed by algorithms such as logistic regression and random forest, and hyperparameter optimization is carried out using GridSearchCV, and data imbalance and outliers are handled by stratified sampling and RobustScaler. The experiments show that the balanced accuracy of the logistic regression model is 70%, and the accuracy of the random forest model is 67.33%. Feature selection significantly improves the model performance. The method proposed in this study demonstrates high efficiency and reliability in large-scale network data and provides a technical basis for the design of real-time intrusion detection systems.
Keywords: Network intrusion detection; Feature selection; ANOVA; Logistic regression

1 INTRODUCTIONS

The rapid evolution of network technology demands advanced intrusion detection systems (IDS) addressing high-dimensional data, redundancy, and class imbalance. Machine learning (ML) offers innovative solutions through feature engineering, model optimization, and data preprocessing.

Wei Jintai and Gao Qiong [1] integrated information gain with a random forest classifier using Quantum Particle Swarm Optimization (QPSO) and ReliefF for dimensionality reduction, while Zhu Linjie et al. [2] applied mutual information (MI) to filter low-redundancy features, achieving 99.8%/99.6% accuracy on NSL-KDD. He Hongyan et al. [3] improved small-sample attack recall via ET-RFE feature selection and optimized LightGBM. Samantaray et al. [4] advocated MaxAbsScaler for IoT feature scaling, enhancing generalization. Ali et al. [5] developed a genetically optimized ensemble model achieving 0.00145 MAE on Kaggle. Nabi and Zhou [6] reduced CICIDS2017 features to 10 dimensions using autoencoders and PCA, retaining 99.6% accuracy with random forests. Arco et al. [7] proposed a two-step training framework to mitigate dataset contamination. Sarhan et al. [8] emphasized standardized feature sets for IoT detection. Ren Jiadong et al. [9] validated Pearson correlation-based feature selection across classification tasks.

Despite advancements, lightweight models, dynamic adaptability, and cross-domain transferability remain challenges. This study integrates ANOVA, chi-square, and Gini coefficient-based feature selection with PCA, optimizes logistic regression and random forest via GridSearchCV, and applies stratified sampling and RobustScaler to enhance detection efficiency and reduce false alarms in large-scale networks.

2 ALGORITHMIC PRINCIPLES FOR NETWORK INTRUSION DETECTION BASED ON MACHINE LEARNING

With the rapid development of network technology, network security is facing increasingly severe challenges. Network intrusion detection methods based on machine learning have become a research hotspot, but still face the problems of high dimensionality, redundancy and imbalance of data features. In order to overcome these problems, feature selection, dimensionality reduction techniques, and hyperparameter optimization have become important means to improve the model performance.

2.1 Feature Selection and Dimension Reduction Techniques

The purpose of feature selection is to select the most relevant features for the target variable from the original feature set. In machine learning-based network intrusion detection, the original feature set often contains a large amount of redundant and irrelevant information, and feature selection aims to select the most relevant features for the target variable (intrusion judgement) from the original feature set to reduce the spatial dimensionality of the features while retaining the classifier's ability to effectively interpret the data patterns. The most relevant features for the target variable (intrusion judgement) are selected from these original features to reduce the spatial dimension of the features, while retaining the classifier's ability to effectively interpret the data patterns. Common feature selection methods can be divided into the following categories:

(1) Filtering method: By calculating the correlation between features and target variables, features are filtered based on the statistical relationship between features and target variables. This method does not rely on specific learning algorithms and therefore has a high computational efficiency.

The chi-square test measures the degree of influence by calculating the chi-square value between the characteristics and the target variable, and the central idea is to compare the difference between the observed and expected values, which is given by the formula:

$$\chi^2 = \sum_i \frac{(O_i - E_i)^2}{E_i} \# \tag{1}$$

Where O_i is the observed value and E_i is the expected value. When the chi-square statistic is larger, it indicates a stronger correlation between the feature and the target variable.

(2) Packing method: the best performing feature subset is selected by constructing several different feature subsets and evaluating them directly using a predictive model. Recursive Feature Elimination (RFE) is a typical representative, which is based on the initial full set of features and evaluates the performance of feature subsets by constructing a prediction model, the core steps of which are to train the model, compute the feature importance scores, and gradually eliminate the weakest features. Each iteration removes the least contributing features and re-evaluates them until the preset conditions are satisfied. In addition, forward stepwise selection and backward stepwise elimination are also common packaging methods, with the former gradually adding features starting from the empty set and the latter gradually removing features starting from the full set. This method is more computationally intensive, but it can obtain the feature combination with optimal performance.

(3) Embedding method: feature selection is performed during model training, and feature importance assessment and model construction are completed at the same time, which has higher efficiency. Commonly used embedding methods include:

Decision tree: the decision tree selects the optimal split features based on indicators such as information gain, information gain rate or Gini index when constructing the tree structure, so as to automatically screen out the features that play an important role in classification; the information gain is calculated as:

$$IG(T,X) = H(T) - \sum_{i=1}^{n} \frac{|T_i|}{|T|} H(T_i) \# \tag{2}$$

where $H(T)$ is the entropy before feature splitting, $H(T_i)$ is the conditional entropy after splitting, and $|T_i|$ and $|T|$ are the sizes of the subset and the full set, respectively.

Lasso regression achieves feature selection by adding an L1 regularization term to the loss function, which makes the coefficients of some of the features zero. Its loss function is:

$$L = \frac{1}{2n} \sum (y_i - \hat{y_i}) + \lambda \sum_{j=1}^{p} |\beta_j| \# \tag{3}$$

where λ is the regularisation factor that controls the strength of the contraction on the coefficients.

Dimensionality reduction techniques aim to reduce the dimensionality of the data and reduce the computational complexity while retaining as much of the main information of the data as possible. Common methods of dimensionality reduction are Principal Component Analysis (PCA).

(1) Principal Component Analysis (PCA), which projects the data into a low-dimensional space consisting of principal components by constructing the covariance matrix and extracting its eigenvalues and eigenvectors. The covariance matrix of the original data is first calculated, which reflects the correlation between the features. The covariance matrix is then eigenvalue decomposed to obtain the eigenvalues and eigenvectors. The eigenvalues represent the variance of the data in each direction, the larger the variance is, the more information is available. PCA sorts the eigenvectors according to the eigenvalues from the largest to the smallest, and selects the first k eigenvectors (k is the dimensionality after dimensionality reduction), and then projects the original data to the low-dimensional space composed of these k eigenvectors to achieve dimensionality reduction. The principal component corresponds to the eigenvector with the largest eigenvalue.

Feature selection and dimensionality reduction are closely related techniques that can often be used in combination to improve the efficiency and accuracy of intrusion detection systems. In addition, these techniques can reduce noise interference and improve the generalization ability and reliability of the model.

2.2 Model Training and Optimization

In the development of machine learning frameworks, the pivotal stages of training and optimization play a critical role, as they are intrinsically linked to the model's predictive accuracy and its ability to generalize to unseen data. The selection of suitable algorithms, coupled with effective optimization strategies, is essential for ensuring that the intrusion detection system operates with both efficiency and precision. Machine learning methodologies can be broadly categorized into linear models, nonlinear models, and ensemble learning techniques. Among the prevalent machine learning algorithms are:

Logistic Regression: This methodology is primarily employed for binary classification tasks and is particularly effective for linear separable problems. The fundamental principle involves utilizing a logistic function (Sigmoid function) to constrain the output of linear regression within the range of 0 to 1, thereby facilitating the classification of samples into positive (anomalous) or negative (normal) categories. Logistic regression is characterized by its low computational overhead, straightforward training process, and the ability to determine weight parameters through optimization techniques such as gradient descent, which can achieve rapid convergence even with extensive datasets. Moreover, the interpretability of logistic regression is notable, as the weight parameters provide clear insights into the impact of individual features on the predictive outcomes.

Decision Tree: This algorithm constructs a hierarchical tree structure by iteratively partitioning the dataset to perform classification or regression tasks. The criteria for partitioning include metrics such as information gain, gain ratio, and Gini impurity. The tree is built upon the distinct values of the input features, where information gain quantifies the contribution of these features to the classification task by assessing the variability in information entropy before and after the split; a higher information gain indicates a more significant capability of the feature to segregate the data effectively.

K-Nearest Neighbor Algorithm (K-NN): This classification technique operates by assessing the proximity between test samples and training samples, subsequently identifying the K nearest neighbors for classification. It is particularly advantageous in scenarios involving small datasets or when there is a strong correlation among features. The K-NN algorithm is noted for its simplicity and intuitive nature, requiring no complex model training, and exhibits commendable performance with limited sample sizes and intricate data distributions. However, it is computationally intensive, necessitating the retention of all training samples in memory, and it can be sensitive to the selection of the K parameter, impacting classification outcomes significantly.

3 EXAMPLE ANALYSES

3.1 Data Sources

In terms of data scale (Data sources: https://research.unsw.edu.au/projects/unsw-nb15-dataset), the dataset has more than 80,000 records to provide rich support for model training and testing, which is conducive to the model learning normal and abnormal network behaviour patterns, and improving generalization and detection accuracy. In terms of feature completeness, 45 features cover multiple dimensions of network connections, which can comprehensively describe network traffic characteristics, enable the model to identify intrusion behaviours in multiple dimensions, and enhance the detection effect. On the target variable setting, the fields attack_cat (attack category) and label (whether it is an attack) indicate that it is designed for intrusion detection, and is suitable for multi-classification and bi-classification tasks respectively, which is convenient for researchers to model and evaluate. In summary, although this dataset has the potential problem of category imbalance, it is suitable for the research needs of intrusion detection in terms of data size, feature completeness and target variable setting, and has research value.

3.2 Data Pre-Processing

Firstly, data import and merge, read the training set and test set data from the specified path and merge them for subsequent unified processing. Then variable categorization is performed to clearly classify the category variables, numeric variables and target variables. After that, the category variables are encoded using LabelEncoder to convert the category data in string form into numeric values. Then data partitioning was performed to divide the combined data into training and test sets through stratified sampling. Finally, data exploration was performed to analyze the dependent variables as shown in Figure 1:

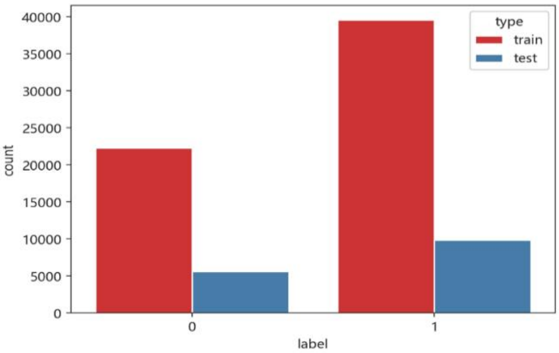

Figure 1 Analysis of Dependent Variables

The data was found to have a significant category imbalance problem, which may affect the training and generalization ability of the model; also, outliers were observed in the numerical variables in the independent variables, and a large number of outliers were found, therefore, data processing with RobustScaler was chosen to reduce the impact of outliers.

3.3 Feature Engineering

*3.3.*1 *Correlation analysis*

Analysis of variance (ANOVA): used to analyze the relationship between continuous features and discrete target variables, calculating the F-value and P-value for each numerical feature and ranking the features according to the F-value, as shown in Figure 2

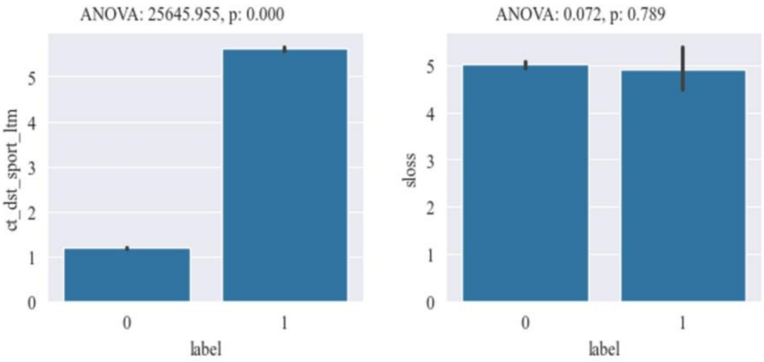

Figure 2 Analysis of Variance

The resultant visualization shows strong correlations between multiple numerical features and the target variable.
Chi-square test: used to assess the correlation between discrete features and discrete target variables, the chi-square and p-value of each categorical feature is calculated and ranked as shown in Figure 3:

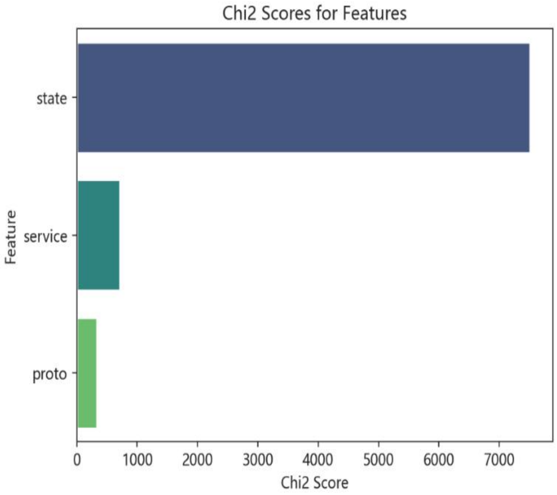

Figure 3 Chi-Square Test

Visualization through bar charts revealed a high correlation between the characteristics of STATE, SERVICE and PROTO and the target variables.

*3.3.*2 *Feature Selection and Dimension Reduction*

Gini coefficient feature selection, The Gini coefficient of importance of features is calculated with the help of Random Forest model to identify features of lower importance as shown in Figure 4.

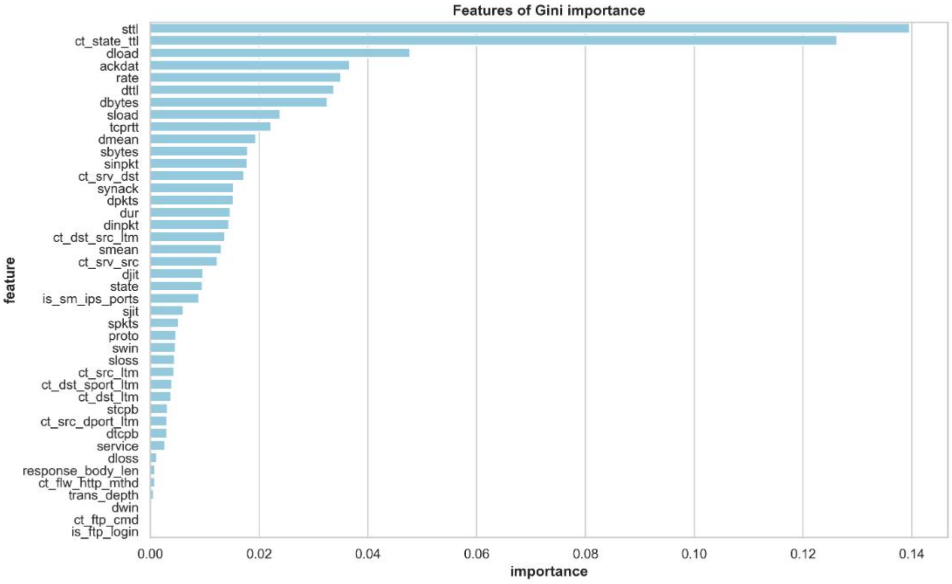

Figure 4 Gini Coefficient Feature Selection

The “sttl” corresponds to the highest importance value, close to 0.12, with the longest bar. The other features have decreasing importance values in order and the bars are correspondingly shorter in length.

3.4 Model Construction and Evaluation

Firstly, model construction and optimization are carried out, adopting logistic regression model and systematically tuning the hyperparameters of the model with the help of GridSearchCV and cross-validation techniques to seek the best combination of hyperparameters, so as to improve the overall performance of the model. Then the model evaluation is carried out to evaluate the performance of the model on the test set using the balanced accuracy as the evaluation criterion, and the confusion matrix and classification report are calculated at the same time in order to comprehensively analyze the classification effect of the model as shown in Figure 5:

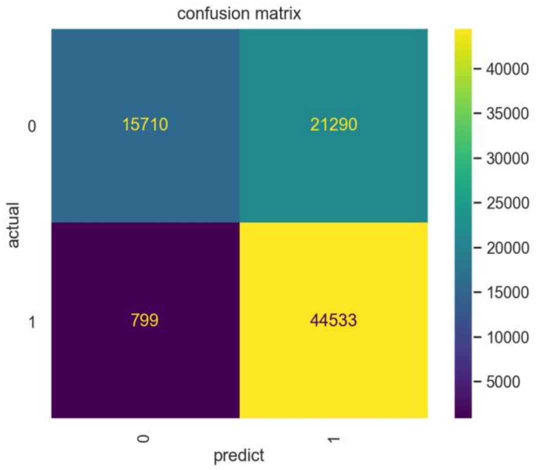

Figure 5 Logistic Regression Model

The results show that the logistic regression model has an equilibrium accuracy of 0.70 on the test set and that there are significant differences in precision, recall and F1 values across categories.

Random Forest Model, the accuracy of the model on the test set was calculated by “accuracy-score” as 0.67, which indicates that the model is correct in about 67.33% of its predictions on the test set, but this accuracy is not considered high and there may be room for optimization. Feature Importance, the importance of each feature is calculated and shown in Figure 6.

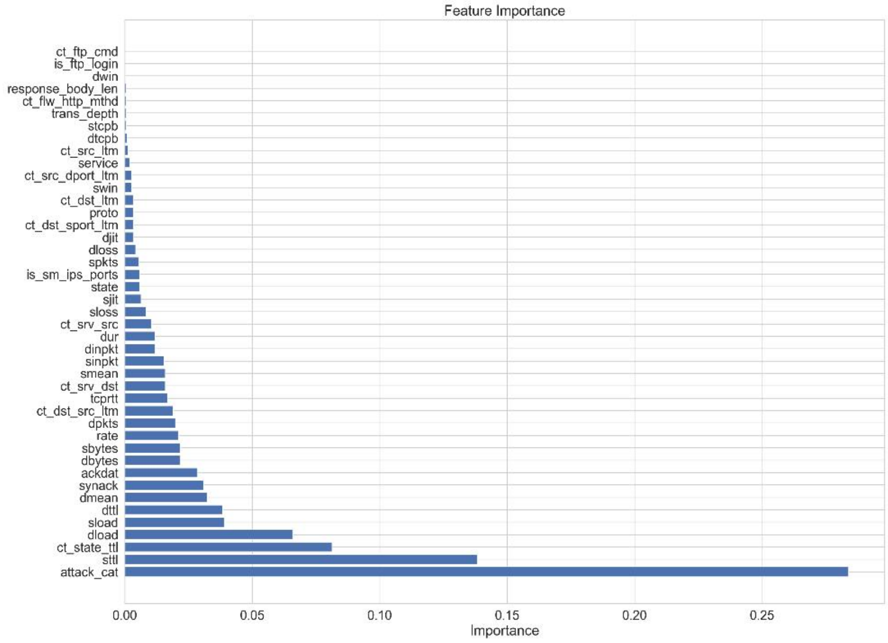

Figure 6 Random Forest Model

The importance of the "attack-cat" feature is 0.218024, which is relatively high among all the features, indicating that it plays a large role in the model decision-making process; while the importance of features like "ct-ftp-cmd" and "is-ftp-login" is only 0.000007, which has less impact on the model decision.

This paper focuses on the task of classifying network traffic data, and integrates a variety of technical tools such as data processing, feature engineering, and model training and evaluation, with the goal of constructing effective classification models to identify cyber-attacks.

Through multiple analysis methods in feature engineering, the importance and relevance of different features to the target variables are clarified, providing a clear direction for subsequent model optimisation, e.g., feature screening can be performed based on feature importance, or other more complex models can be tried to improve classification performance.

4 CONCLUSIONS

This study proposes efficient machine learning-based intrusion detection method, an intrusion detection method that combines feature selection and optimized machine learning algorithms, which can operate efficiently in large-scale data environments with high accuracy and low false alarm rate. A real-time intrusion detection framework is created to design and implement a real-time intrusion detection system that can efficiently respond to changes in network traffic, ensuring a balance between detection accuracy and real-time performance.

From the evaluation results of the classification model, the logistic regression model achieves a certain degree of accuracy in dealing with this network traffic classification task, but there is still room for further improvement in the model performance due to the category imbalance of the data and the complexity of the network traffic data.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] Wei Jintai, Gao Qiong. Research on intrusion detection system based on information gain and random forest classifier. Journal of North Central University (Natural Science Edition), 2018, 39(01): 74-79+88.

[2] LJ Zhu, G P Zhao, L H Kang. Intrusion detection method based on the combination of MI feature selection and KNN classifier. Gansu Science and Technology, 2022, 38(15): 33-36.

[3] Hongyan He, Guoyan Huang, Bing Zhang, et al. Anomaly detection model based on recursive elimination of limit tree features and LightGBM. Information Network Security, 2022(1): 64-71.

[4] Samantaray M, Barik C R, Biswal K A. A comparative assessment of machine learning algorithms in the IoT-based network intrusion detection systems. Decision Analytics Journal, 2024: 11100478.

[5] Ali M A, Owais M S Q, Andleeb M S, et al. Robust genetic machine learning ensemble model for intrusion detection in network traffic. Scientific Reports, 2023, 13(1): 17227-17227.

[6] Nabi F, Zhou X. Enhancing intrusion detection systems through dimensionality reduction: a comparative study of machine learning techniques for cyber security. Cyber Security and Applications, 2024: 2100033.

[7] Arco M G J, Carrión M R, Gómez R A R, et al. Methodology for the Detection of Contaminated Training Datasets for Machine Learning-Based Network Intrusion-Detection Systems. Sensors, 2024, 24(2).

[8] Sarhan M, Layeghy S, Moustafa N, et al. Feature extraction for machine learning-based intrusion detection in IoT networks. Digital Communications and Networks, 2024, 10(01): 205-216.

[9] Ren Jiadong, Zhang Yafei, Zhang Bing, et al. A feature selection-based classification method for industrial internet intrusion detection. Computer Research and Development, 2022, 59(05): 1148-1159.