Indian Journal of Computer Science and Technology
Volume 3, Issue3 (September-December 2024), PP: 20-23.
www.indjcst.com
ISSN No: 2583-5300

Feature Engineering: The Key to Advanced Intrusion Detection

Semanpreet Singh
Assistant Professor, Department of Computer Science, Thapar Institute of Engineering and Technology, Patiala, Punjab, India.

To Cite this Article: *Semanpreet Singh, "Feature Engineering: The Key to Advanced Intrusion Detection", Indian Journal of Computer Science and Technology, Volume 03, Issue 03 (September-December 2024), PP: 20-23.*

Abstract: *The researchers of data science aim at getting actionable insights from raw data by applying techniques from multiple fields including statistics and machine learning. Machine learning provides many supervised learning algorithms like K-Nearest Neighbor (KNN), Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), Artificial Neural Network (ANN), Rule-based classifiers and Logistic Regression, etc. that support for building IDS models. The suitability of a model is determined based on the type of features. Specifically, ANN, Logistic Regression, KNN, etc. are preferred to build classifiers using numeric features while, DT, RF, Rule-based classifiers, etc. supports building classifiers by involving categorical features. When a dataset contains mixed types of features model selection is influenced by the type of majority features. Since most of the datasets have mixed type of features, there is a requirement to convert numerical features into categorical features and vice versa. Converting numerical features to categorical form is well addressed through different types of discretization methods.*

Keywords: *K-Nearest Neighbor, Decision Tree, Random Forest*

I.INTRODUCTION

The conversion of categorical features into numeric form is comparatively less explored. It wasobserved that the R-programming language represents categorical features as factors that are lexicographically ordered implicitly. This representation is incompatible with the semantics of the values of the categorical features especially in the context of distance estimation. For example, the categorical feature "Profession", when represented as a factor with distinct values Doctor, Engineer, and Lawyer, leads to a misconception that Doctor is closer to Engineer than a Lawyer. This is merely based on the lexicographic order of the string of characters representing the values rather than actual semantics. Hence, a systematic algorithm for transforming categorical features into numeric form needs to be established to build classifiers related to application areas like IDS, medical diagnosis, Business Intelligent systems, etc. The datasets consisting of features which are predominantly numeric with a few categorical features are classified by building models that deal with numeric data,after converting the few categorical features into numeric form. This thesis focuses on exploring the applicability of the machine learning technics for building intrusion detection systems with high accuracy and reduced False Positive Rate (FPR). In the context of IDS, most of the standard datasets have predominantly numeric data with the few categorical features and hence calls for the exploration of appropriate methods for handling the few categorical features. This chapter presents the details of two different methodologies explored by the author for handling categorical features by (i) partitioning the dataset for building multiple tailor-made models using a "protocol-specific approach", (ii) converting the categorical features into the numeric form using "Encoding approach". The following sections provide the details of experimentation with these approaches in the context of IDS.

II. PROTOCOL-SPECIFIC APPROACH TO IDS

Protocol specific approach to IDS performs intrusion detection based on the protocol type of the traffic packet by observing whether the packet is consistent with the expected behavior for a particular protocol or not. According to the literature (Lemonier et.al in , Das et.al in , Barry et.al in [Bar07] and Chung et.al in), many attacks are protocol-based anomalies as most of the attacks exploit vulnerabilities in the design of protocols. The protocol-based attack detection improves performance compared to the basic method and it results in simpler and faster processing of packets at real time. Hence, it contributes to the efficiency of intrusion detection systems.

III. CLASSIFIER SELECTION

The authors used K-Nearest Neighbor (KNN) [Zha05] classifier for classifying the intrusions. KNN is a supervised learning algorithm that doesn't make any generalizations and assumptions on the distribution of training data and also KNN doesn't build any model. Hence, it is considered as a non-parametric lazy classifier. As it is a lazy classifier it is intrinsically dynamic by targeting the current chunk of data for classifying an unknown instance and hence it can be made naturally incremental. This incremental nature is desirable for detecting intrusions as they evolve continuesly. For a given test instance, the label is predicted based on the majority class label of its neighborhood defined by its Knearest neighbors where, the nearness is estimated using the distance metrics like Manhattans, Euclidian, and Murkowski. This research uses the Euclidian distance metric for estimating the nearness. The selected K (number of nearest neighbors) value impacts the efficiency of the KNN classifier. Hence selecting the best K value is important.

Building Multiple Tailor-made Classifiers

An integrated protocol-specific intrusion detection system is developed using multiple KNN classifiers one each to be selectively used for identifying intrusions based on the protocol type of the network traffic packet being classified. KNN being a non-parametric method that relies on distance estimations for the formation of the neighborhood, a specific methodology is devised to process categorical features. Pre-processing is done by initially removing insignificant categorical features. Conversion of two-valued categorical features into binary valued numeri's is consistent with the semantics of distance estimation and hence, all two-valued categorical features are converted into binary-valued numeri. It is established in the literature by being the rationale for developing proven algorithms like Attribute Oriented Induction (AOI), a categorical attribute with too many distinct values is insignificant and hence removable. Accordingly, all categorical attributes with too many distinct values (more than ten) were ignored to develop protocol-specific classifiers. The process of building protocol-specific classifiers starts by partitioning the dataset based on protocol type and the individual modules are referred by the protocol name. Since some classes are more prominent and some classes are very less prominent for a specific protocol, significantly frequent classes for each protocol module were identified. Accordingly, each of the protocol module is modified to contain only the identified frequent classes by removing the less frequent class instances and is made balanced either by oversampling the less frequent class instances or under-sampling the highly frequent class instances. The instances of each module are normalization to transform all numeric features into a uniform scale.

5×2 cross-validation [Eth20] is conducted on each protocol specific module to determine the protocol-specific best K value. Initially, training and validation sets are taken randomly from each module in equal proportions. Euclidian distance is estimated between the pairs of traffic packets across the training and validation data sets and accordingly, K-nearest neighbors are identified for each validation traffic packet based on specific K value and determine the majority class label in its neighborhood. The same experimentation procedure is again done by swapping training and validation sets and this completes the first fold out of five folds in 5×2 cross-validation method. The process is repeated for the remaining four folds by repeatedly taking a random sampling of training and validation sets in equal proportions without replacement to complete the 5×2 cross-validation process. The experimentation was repeated for different K values and the best value of K that results in the highest detection accuracy is found for each protocol-specific module. It may be noted that even though the KNN classifier employs a lazy learning approach it was found beneficial to experiment on validation sets using a 5×2 cross-validation method and predetermine the best K value specific to a protocol which will be stored as a part of the integrated IDS system. In this way, protocol-specific models are formed for different protocols containing balanced training set formed with significant class instances and the evaluated best-K value.

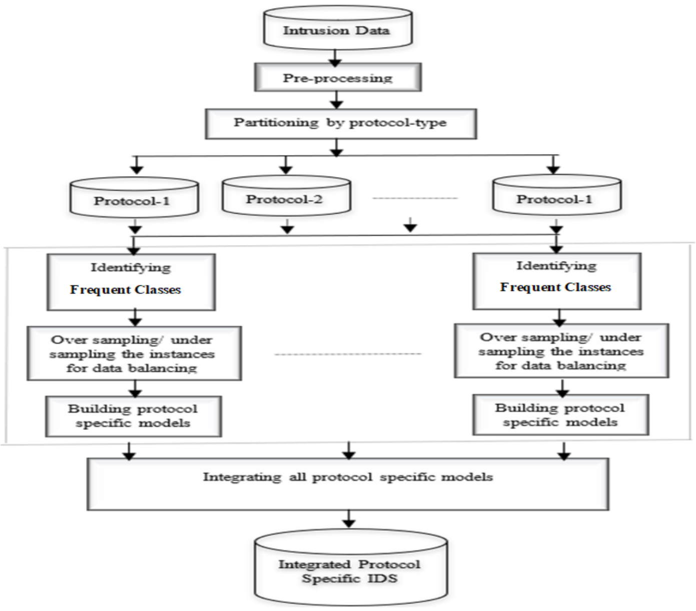
Intrusion Data
Pre-processing
Partitioning by protocol-type
Protocol-1
Protocol-2
Protocol-1
Identifying Frequent Classes
Over sampling/ under sampling the instances for data balancing
Building protocol specific models
Identifying Frequent Classes
Over sampling/ under sampling the instances for data balancing
Building protocol specific models
Integrating all protocol specific models
Integrated Protocol Specific IDS

Figure 1: The process of constructing integrated protocol-specific IDS

Finally, all the protocol models are integrated to form the integrated protocolspecific IDS to classify a given test instance. The process of constructing integrated protocol-specific IDS is presented in Figure 1. The given test instance is classified based on its protocol type by invoking the appropriate protocolspecific model that forms neighborhoods based on its evaluated best-K value. The process of detecting intrusions using integrated protocol-specific IDS is presented.

IV. EXPERIMENTATION & RESULTS

This research uses NSL-KDD dataset for experimentation. NSL-KDD isthe refined version of the KDD CUP 99 dataset. It consisting of a total of 4GB network traffic data provided by DARPA in the form of tcp- dumpfiles. The dimensions of the NSL-KDD are 147907 × 43. A list of variables representing the features of the dataset is presented in Table 3.1. The characteristics of the dataset are given below.

- Distribution: Packet-based
- Number of specific attacks: 39
- Number of attack instances: 70940
- Number of normal instances:76967
- Number of features: 43
- Numeric features: 35
- Bi-valued categorical features: 4
- Multivalued categorical features: 3
- Class label feature:1

Classifier Evaluation

The pre-processed NSL-KDD is divided into an 80:20 ratio where the 80% data part is used to create multiple tailor-made models and the 20% data part is used as a test set. By considering the 80% data part, it was found that the categorical feature protocol-type contains three values: tcp, icmp, and udp. Accordingly, the 80% data part is divided into three modules namely tcp-module, icmp-module, and udp-module. The class-wise distribution of the protocol-specific modules is presented in Figure As shown in the figure, some classes are more prominent and some classes are very less prominent for a specific protocol. From the figure, the classes "Guess_passwd", "Neptune", "satan", "port sweep" and "Normal" are identified as significant classes for the tcp-module, the classes “Ipsweep”, “Nmap”, “Pod”, “Smurf” and “Normal” are identified as significant classes for the icmp-module and the classes “Satan”, “Snmpget”, “Snmpguess”, “Teardrop” and “Normal” are identified as significant classes for the udp-module The test set which is taken from 20% of the NSL-KDD dataset is normalized using min-max normalization and submitted to the integrated protocol-specific IDS system for multi-class classification into a specific attack or a normal class. Initially, each packet of the test set is verified for its protocol-type, and accordingly, the corresponding protocol-specific model is invoked for attack detection.

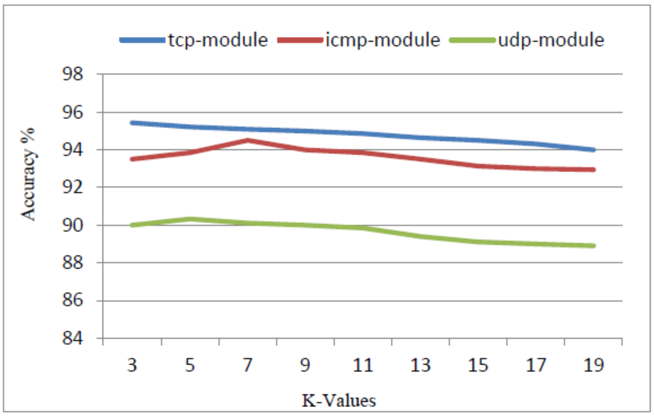

Figure 3: Validation results of protocol modules at different K-values

V. CONCLUSION

Protocol-specific IDS that generate protocol-specific tailor-made models, integration of which results in better performance in terms of reduced prediction time compared to the direct method. The second method is the Encoding approach for IDS which can effectively encode all prominent multi-valued categorical features and generates a numeric version of the datasets based on the posterior probability of the corresponding class given the feature-value pair. The experimental results show that the proposed method for encoding the categorical features led to better classification accuracies compared to the state-of-the-art methods.

However, the success of these methods as well as the other traditional machine learning approaches relies on the availability of labeled examples for learning the signatures of the attacks. Since the attacks are ever-evolving, some of the attacks are new to the IDS as they have disparate signatures. Hence, new as well as zeroday attack detection becomes the toughest challenge taken up by the IDSs. In the next chapter, authors have proposed an Inductive Transfer Learning framework for detecting new attacks in the context of IDNs through IDS collaboration with the application of transfer learning.

Reference

1. *Florackis, C.; Louca, C.; Michaely, R.; Weber, M. Cybersecurity Risk. Rev. Financ. Stud.* ***2022****, 36, 351–407.*
2. *Insua, D.R.; Couce-Vieira, A.; Rubio, J.A.; Pieters, W.; Labunets, K.; Rasines, D.G. An Adversarial Risk Analysis Framework for Cybersecurity. Risk Anal.* ***2019****, 41, 16–36.*
3. *Mills, R.; Marnerides, A.K.; Broadbent, M.; Race, N. Practical Intrusion Detection of Emerging Threats. IEEE Trans. Netw. Serv. Manag.* ***2021****, 19, 582–600.*
4. *Maseno, E.M.; Wang, Z.; Xing, H. A Systematic Review on Hybrid Intrusion Detection System. Secur. Commun. Netw.* ***2022****, 2022, 9663052.*
5. *Zipperle, M.; Gottwalt, F.; Chang, E.; Dillon, T. Provenance-based Intrusion Detection Systems: A Survey. ACM Comput. Surv.* ***2022****, 55, 1–36*