Applied Soft Computing 54 (2017) 1–22

Contents lists available at ScienceDirect

Applied Soft Computing

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

Optimizing an artificial immune system algorithm in support of flow-Based internet traffic classification

Brian Schmidt*, Ala Al-Fuqaha, Ajay Gupta, Dionysios Kountanis

Computer Science Department, College of Engineering and Applied Sciences, Western Michigan University, Kalamazoo, MI, USA, USA

ARTICLE INFO

Article history:
Received 24 November 2015
Received in revised form 9 January 2017
Accepted 10 January 2017
Available online 13 January 2017

Keywords:
Artificial immune system
Internet traffic classification
Multi-class classification
Machine learning

ABSTRACT

The problem of classifying traffic flows in networks has become more and more important in recent times, and much research has been dedicated to it. In recent years, there has been a lot of interest in classifying traffic flows by application, based on the statistical features of each flow. Information about the applications that are being used on a network is very useful in network design, accounting, management, and security. In our previous work we proposed a classification algorithm for Internet traffic flow classification based on Artificial Immune Systems (AIS). We also applied the algorithm on an available data set, and found that the algorithm performed as well as other algorithms, and was insensitive to input parameters, which makes it valuable for embedded systems. It is also very simple to implement, and generalizes well from small training data sets. In this research, we expanded on the previous research by introducing several optimizations in the training and classification phases of the algorithm. We improved the design of the original algorithm in order to make it more predictable. We also give the asymptotic complexity of the optimized algorithm as well as draw a bound on the generalization error of the algorithm. Lastly, we also experimented with several different distance formulas to improve the classification performance. In this paper we have shown how the changes and optimizations applied to the original algorithm do not functionally change the original algorithm, while making its execution 50–60% faster. We also show that the classification accuracy of the Euclidian distance is superseded by the Manhattan distance for this application, giving 1–2% higher accuracy, making the accuracy of the algorithm comparable to that of a Naïve Bayes classifier in previous research that uses the same data set.

Published by Elsevier B.V.

1. Introduction

Because of recent changes in the regulatory climate of the Internet, as well as the business needs of Internet Service Providers, the traffic classification problem has become an important research topic. Some concerns are: the struggle between malicious users and security professionals, network neutrality, and the use of networks for sharing copyrighted material. The task of optimizing the flow of traffic across a network is also related to this problem, since some applications rely on low latency to provide Quality of Service, while others are unaffected by it. The goal of network operators has been to classify network traffic according to the application that generated it, which is highly correlated to the type of information contained in it. On the other side, application developers have sought to hide the identity of their application's packets on the network by obfuscating their signature.

In the early days of the Internet, an application could easily be identified by the port number used in the transport layer. This approach is very simple and still useful, but is not accurate enough in the modern Internet, since certain applications are able to fool this method by negotiating port numbers dynamically, making it impossible to reliably identify them.

Deep packet inspection is also useful when doing network traffic classification, and involves analyzing the contents of packets. To find the patterns used by certain applications, regular expressions are often used. There are a few shortcomings to this approach as well, since using encryption allows users to easily hide their data from this method, while also being very resource intensive, since every packet has to be examined. There are also concerns about the privacy of users [1].

A non-intrusive technique used in recent years is traffic flow classification based on features of the flow, in which the statistical properties of traffic flows are calculated and used to identify the generating application by comparing the information

* Corresponding author.
E-mail addresses: brian.h.schmidt@wmich.edu (B. Schmidt), ala.al-fuqaha@wmich.edu (A. Al-Fuqaha), ajay.gupta@wmich.edu (A. Gupta), dionysios.kountanis@wmich.edu (D. Kountanis).

http://dx.doi.org/10.1016/j.asoc.2017.01.016
1568-4946/Published by Elsevier B.V.

to previously-learned models. Some example features are: inter-packet arrival time, average packet size, and packet counts.

Another way to identify the applications that are generating network traffic is by looking at the interactions that a host engages in, and comparing them to behavior signatures that are associated with certain application servers. This approach to traffic classification depends strongly on the topological location of the monitor, and performs well when the monitor sees both directions of the flow under inspection [1,2].

The focus of this paper will be to utilize the statistical features of network flows to identify the generating application. We will accomplish this by using a multi-class Artificial Immune System inspired classification algorithm. We are encouraged to try this approach because of the use of AIS algorithms in similar network traffic classification problems.

Our proposed approach uses fewer parameters than other natural computing algorithms and does not incur the training costs associated with discovering such parameters. For example, the performance of the genetic algorithms highly depends on the mutation and cross-over operations and parameters. Similarly, the performance of artificial neural network, deep learning and extreme machine learning based approaches depends highly on the number of hidden layers, the number of neurons in each layer and the employed activation function. Also, the performance of SVM is highly dependent of the Kernel function used and its parameters. In this paper, we propose an optimized AIS algorithm that needs few parameters and produces results comparable to these produced by the optimal parameters of the aforementioned methods. Therefore, the proposed approach eliminates all the overhead and subjectivity involved in the selection of the parameters in other biologically inspired approaches.

Furthermore, Artificial Immune System algorithms are able to operate in highly distributed systems and can be easily adapted to run on networked computers. AIS algorithms are capable of learning new patterns, remember previously learned patterns, and do pattern recognition in networked systems. At the same time, their performance degrades gracefully, in the same way as Artificial Neural Networks. In past research, AIS algorithms have been used to detect malicious activity in computer networks [3]. Because of this research and the capabilities of AIS classifiers we are encouraged to explore their performance on the task of network flow classification. Research has also shown that positive selection AIS algorithms can perform very well compared to negative selection in problems that require a comprehensive data set of negative examples. Positive selection isalso being simpler to code and faster to train. For this reason, the algorithm presented in this paper is a positive selection algorithm.

The original algorithm described here is designed to be simple and fast so that it will work well in resource-constrained systems. Because of our previous findings, we have been motivated to develop optimizations for the algorithm, to make it competitive with other Machine Learning approaches while depending on lesser configurable parameters.

When testing the optimizations made to the algorithm, a speedup of about 10x–30x was achieved in the training algorithm. A speedup of around 2x was observed in the classification portion of the algorithm. No significant differences where observed in the accuracy of the optimized and unoptimized algorithms. When testing different distance functions, it was observed that Manhattan distance was 1% to 2% more accurate for the data set used.

The rest of the paper is organized as follows. Section II introduces the traffic flow classification problem along with other solutions found in the literature. Section III introduces artificial immune systems, including their biological inspiration. Section IV introduces the problem under investigation, places our own solution in context, and describes our own classifier, inspired by AIS principles. Section V describes the changes that we made to the algorithm to optimize it. Section VI deals with an analysis of the performance of the algorithm, section VII explains the tests performed on the algorithm, including the data set used. Section VII shows the results of the tests, section IX contains our conclusions and recommendations for future work.

2. Background

2.1. The flow classification problem in machine learning

In [4], Moore and Zuev applied a Naive Bayes classifier to the traffic classification problem. A simple naive Bayes classifier did not do very well at first, with an average 65.3% classification accuracy. The accuracy rose, however, when kernel density estimation and Fast Correlation-Based Filter (FCBF) feature reduction were applied. The techniques were tested separately and jointly, with the best performance achieved when both techniques were used at the same time, achieving 96.3% classification accuracy.

In [5], a survey is carried out of the state of traffic classification, including a review of all approaches to the problem, as well as all machine learning algorithms that have been tested. Furthermore, an introduction to elephant and mice flows and early classification was provided. The authors also highlighted the processing time, memory, and directional neutrality aspects of the algorithms.

In [6], Alshammari and Zincir-Heywood, tested Support Vector Machines (SVM), Naive Bayes, RIPPER, and C4.5 algorithms using three publicly available data sets, focusing on classifying encrypted traffic. Singh and Agrawal [7] also tested several of the same ML algorithms as [6] on the task of traffic classification, the algorithms being: Bayes net, multi-layer perceptron, C4.5 trees, Naive Bayes, and Radial Basis Function Neural Networks. Both full-feature and reduced-feature data sets were tested and results compared, with the best classification accuracy achieved by C4.5 with a reduced feature data set. Lastly, [8] focused on the accurate classification of P2P traffic using decision trees, achieving between 95% and 97% accuracy.

The selection of flow features for classification has also been studied in the literature. [9] performs a survey of the reasons why some algorithms perform well on the traffic classification problem, as well as the features that are most useful. Their results show that there are three features in particular that are most useful: ports, the sizes of the first one or two packets for UDP flows, and the sizes of the first four or five packets for TCP flows. This paper also finds that Minimum Description Length discretization on ports and packet sizes improves the classification accuracy of all algorithms studied. In [10], feature selection for flow classification is tested. The best performance is achieved when using information from the first 7 packets with a one-against-all SVM classifier, confirming the findings of [9]. Specifically, [9,10] have shown that it is possible to classify a flow accurately with only limited information about it. Lastly, in [11], the data set is preprocessed by removing outliers and using data normalization, and performing dimensionality reduction. Decision Trees, Artificial Immune Networks, Naive Bayes, Bagging and Boosting classifiers are tested. Although Artificial Immune Networks are used in this research, they are substantially different algorithms from the one used in this research.

In [12], the authors use Extreme Learning Machines (ELM) to tackle the supervised learning network traffic classification problem. ELMs are like artificial neural networks, however, ELMs use randomized computational nodes in the hidden layer and generate their weights by solving linear equations. A similar approach is taken in [13], although the ELMs used are kernel based. In this