Efficient Neural Computation on Network Processors for IoT Protocol Classification

Vibha Pant*, Roberto Passerone†, Michele Welponer†, Luca Rizzon‡ and Roberto Lavagnolo‡
*Dept. of Computer Science and Engineering, Amrita School of Engineering, Bengaluru,
Amrita Vishwa Vidyapeetham, Amrita University, India
†Department of Information Engineering and Computer Science, University of Trento, Trento, Italy 38123
‡Microtel Innovation Srl, via armentera 8, Borgo Valsugana, Trento, Italy 38051

Abstract—**The Internet of Things (IoT) brings forth pressing requirements on the service providers in terms of service differentiation, which plays an important role in pricing policies as well as network load balancing. In this paper, we consider differentiation of application level protocols for IoT from general application protocols through flow classification. We implement a neural network classifier that can run at wire speed reaching 100 Gbps on a network processor. In particular, we study approximations which allow us to efficiently compute the neural network output, while complying with the network processor limitations, which does not provide multiplication or other complex mathematical operations. The results show that the implementation is efficient and that the classification error is negligible.**

I. Introduction and Related Work

Traffic and packet classification is an important functionality at the basis of network management activities such as resource planning, quality of service (QoS) provisioning, load balancing and lawful intrusion detection [1]. In particular, the increasing adoption of IoT devices raises specific requirements on network operators, which must be able to distinguish IoT traffic to determine the correct QoS, pricing and provide differentiated services for applications and network monitoring [2]. Packet classification must therefore be performed at wire speed, so that communication flows can be routed to the appropriate processing device in a balanced manner according to their class of service. In this paper, we look at statistical analysis and classification using neural networks implemented on network processor (NP) architectures, which are commonly used when packets must be processed and routed at wire speed, owing to their dedicated hardware for buffering, table search and update, and forwarding [3]. On the other hand, NPs are not designed for complex computation, as their simplified instruction set does not provide multiplication or higher math operations. Our main contribution is a set of approximations that make use of only additions and subtractions, which can be implemented efficiently on NPs and significantly improve performance. Our evaluation shows that the approximations do not affect the classification accuracy appreciably.

Several methods can be used to perform packet and protocol classification. The simplest and most efficient classification technique is port matching, where the destination and source port numbers or IP addresses are matched to a set of well known values using pre-defined rules [4], [5]. This is frequently used in intrusion detection systems on the server side. Port spoofing or camouflaging however make this unreliable and has rendered it obsolete [6]. Packet Payload analysis or Deep Packet Inspection (DPI) achieve high classification accuracy, however in many scenarios the payload is not accessible due to encryption or legal privacy restrictions [7].

In this paper, we opt for Deep Flow Inspection (DFI) where we look at traffic as a flow and which does not require the analysis of the payload. Flow statistics such as flow length, packet size distribution, session start and end time can be used in this classification approach [8], [6]. Flow inspection can be treated as a pattern recognition problem where we can apply machine learning algorithms [9], [10]. We use artificial neural networks, which have shown great performance in classification and clustering of large amounts of data received from sensors [11]. Shen et al. use several statistical properties related to packet size as input features to distinguish P2P traffic [10]. We follow in particular the approach proposed by Trussell et al., who use packet size distribution to classify different application protocols [12]. The authors pre-process the data fed into the neural network and use bins in a histogram to hold the distribution of packet sizes for each application. We use a similar approach to pre-process the IoT data. Unlike previous work, we employ a network processor and take advantage of its inherent parallel and multi-tasking capabilities to support computational intensive classification at wire speed.

We first give an overview of the classification methodology in Section II. We then discuss the approximations required to implement the algorithm on a network processor in Section III. Experimental results are presented in Section IV.

II. IoT Protocol Classification Overview

Our classification strategy uses the packet size distribution of a flow to determine the likely application protocol [12]. In particular, we consider the following three protocols, which are the most used in the IoT: Constrained Application Protocol (CoAP), primarily used for communication in constrained nodes with low power, computation and communication capabilities; Message Queuing and Telemetry Transport (MQTT), which connects embedded devices through a publish, subscribe and broker environment; and Advanced Message Queuing Protocol (AMQP), which provides reliable delivery with primitives such as at most once, at least once and exactly once delivery parameters.

978-1-5090-6447-2/17 $31.00 © 2017 IEEE
DOI 10.1109/NGCAS.2017.55