A Review on Features' Robustness in High Diversity Mobile Traffic Classifications

Yichiet Aun, Selvakumar Manickam and Shankar Karuppayah

National Advanced IPv6 Centre of Excellence (NAv6) 6th Floor,
School of Computer & Mathematical Sciences Building,
Universiti Sains Malaysia, Malaysia

***Abstract*:** Mobile traffics are becoming more dominant due to growing usage of mobile devices and proliferation of IoT. The influx of mobile traffics introduce some new challenges in traffic classifications; namely the diversity complexity and behavioural dynamism complexity. Existing traffic classifications methods are designed for classifying standard protocols and user applications with more deterministic behaviours in small diversity. Currently, flow statistics, payload signature and heuristic traffic attributes are some of the most effective features used to discriminate traffic classes. In this paper, we investigate the correlations of these features to the less-deterministic user application traffic classes based on corresponding classification accuracy. Then, we evaluate the impact of large-scale classification on feature's robustness based on sign of diminishing accuracy. Our experimental results consolidate the needs for unsupervised feature learning to address the dynamism of mobile application behavioural traits for accurate classification on rapidly growing mobile traffics.

***Keywords*:** Traffic classification, Survey, Protocol identification, Features & Signature, Payload, Statistical, Application profiling.

1. Introduction

Traffic classification is a machine learning technique for autonomous identification of traffic classes using correlated network parameters. It is an important domain in machine learning for services discovery [1], selective QoS treatment and network management. Traffic classification is gaining momentum in the avenue of network monitoring and traffic management driven by the needs for pervasive network management. In addition, the proliferation of mobile applications has induced diversification of traffic classes and subsequently adds to the complexity of traffic classification [2]. Existing approaches are designed for small-scale classification [3] on standard protocols and small numbers of some randomly selected user applications. Thus, the resulting classifier models are only optimized for local scope and become less effective for non-homogenous use. Meanwhile, the traffic composition on modern networks are more diverse and less predictable due to increased populations of user applications. As the result, classification accuracy is adversely affected if existing classifier models are adopted directly without contextual optimization [4].

Traffic classes are discriminated based on some observable communication traits in network metrics known as *features* [5, 6]. The popular features in the literature of traffic classification include *flow statistics*, *payload signature* and *heuristics features*. Feature attributes evaluator such as *CFS* select features that are highly correlated to traffic class to be paired with classifier of synergy for optimal classification [7, 8, 9]. Then, classifiers evaluate flow instances by the principle of correlations of attributes to the trained feature vectors. Traffic classification is accurate when the vector space of features is highly unique; while vector conflicts is likely to result in high numbers of false positive.

Currently, classification methods are lacking in the features department in terms of feature quality and quantity towards high diversity traffic classification. The discriminative properties for f*low features* such as *packet size*, *flow duration* and *inter-arrival times statistics* are diminishing in increased traffic classes' diversity especially for categorically homogenous applications [1]. They are more susceptible to network conditions rather than application behaviors in a congested network. Existing *payload signature* is not calibrated for mobile traffic space due to some previously negligible signature noise [10]. For example, the variance of payload feature that uses dictionary word such as *'flv'*, *'mp3'*, *'get'* and so on are diminishing as these words become more common across payload bytes in multiple classes. Besides, payload signature using trivial terms inevitably leads to higher false positive in mobile traffic classification [6]. For example, the keyword '*facebook´* are found in abundance in payload of applications (like *eBay*, *Spotify*, *iFlix)* that are inheriting Facebook API services. Lastly, protocol traits decoding also become infeasible as heuristics feature extraction becomes more time consuming given the enormous classification scope; despite being the highly accurate [5].

In this paper, we evaluate the performance of existing classification works on mobile traffic in high diversity in the context features efficacy. The main section of this paper is dedicated to address new criterions emerged in feature engineering towards transitioning traffic classification techniques for large scale and less-deterministic classification. The first half of the paper implement and discusses on the aforementioned evaluations. Then, some potential future works on feature learning are discussed in the remaining section.

2. Feature Engineering in Traffic Classification

Feature engineering is a process to discover useful discriminators using domain specific knowledge to empower machine learning classification [7]. A feature is an individual measurable property of a phenomenon being observed [9]. Features are the basic discriminant unit for traffic classifications. The role of features is to discriminate domain specific entities based on their respective properties as presented in feature vectors. Features are used in *nearest neighbor*, *neural networks* [11], and *Bayesian* classification approaches [12]. There are multiple types of features; some domain specific and some generic. Features are domain

sensitive; for example, features are *histograms* in character recognition, *phonemes* in speech recognition, *structural properties* in spam detection, *edges and objects* in computer vision [13]. Feature can be represented in multiple datatypes, including *integer*, *double*, *float*, *string* or complex *data structure*. Features types include *binary*, *categorical*, *ordinal*, *integer* value, *real* value and *composite* types. A set of numerical features can be conveniently described by a feature vector [7]. Any attribute useful to the model could be a feature, but not all attributes are feature worthy. Features are evaluated based on how correlated they are towards target traffic classes; based on *Pearson's-Correlations Coefficient*. Due to the inequality among features, only useful features are selected for classifications. The relevancy of feature attribute on a specific classification problems can be expressed by attribute evaluator such as *CFS*, given by:

$$CFS = max_{x\epsilon\{0,1\}^n}[\frac{(\sum_{i=1}^{n} a_i x_i)^2}{\sum_{i=1}^{n} x_i + \sum_{i\neq j} 2b_{ij}x_i x_j}]$$

We discuss the types of features used in the domain of traffic classification in the next section.

3. Flow Statistic

Statistical classification approaches use flow features statistics inferencing to deduce traffic classes. Flow statistics features are some correlated per-class attributes discriminators that are conveyed in network metric *[5]* s. Statistical features can be interpreted at packet level, flow level, multi-flows level and connections level *[14, 15]*. In flow statistics processing (FSP), network parameters in TCP headers and IP headers that are highly correlated to unique traits of respective traffic classes are extracted as flow features. Flows-based classification discriminate at headers level; thus, they are effective for encrypted traffic classification *[5, 16]*. Besides, flow features are suitable for unsupervised feature learning given their universal properties in all TCP/IP based traffic classes. FSP is a common feature extraction method in this domain; that explains the highly similar feature sets used in majority of existing classification works. Some common flow statistics features are shown in table 1.

Table 1. Common flow statistics features

Features	Description
Packet length	The size of IP packets given by *IP HDR_LEN* and *IP Total_LEN*
Packet length statistics	Min, max, mean, median, SD of *packet length*
Inter-arrival time (IAT)	The differences of arrival time/delta time/timestamp among packets
IAT statistics	Min, max, mean, median, SD of *IAT*
Packet counts	Number of packets in a defined period/interval/quartile/sample size
Bytes counts	Numbers of bytes in a defined period/interval/quartile/sample size
Flow counts	Numbers of flows in a defined period/interval/quartile/sample size
Timing intervals	Idle time, keepalive time, arrival time, connection time statistics
Flow size	The number of packets in a flow (average)
Flow duration	Length of flows in average per session/interval
TCP port	The port number as read by network monitor (as identifier)
TCP window size	The advertised *TCP* sliding window size
Burst size	The burst intensity of packet in a defined interval
TCP count with PUSH	Number of *TCP* packets with PUSH set to 1
Packet arrival order	The pattern of packets arrival based on sequence number
Effective bandwidth (entropy)	Network utilization parameters
Bytes flow	Number of bytes transferred from client to server; server to client
IP header parameters statistics	The aggregation of *IP* header attributes values
TCP header parameters statistics	*TCP* usage characteristics as conveyed in TCP header parameters such as flags (*SYN, PUSH, FIN, ACK*) and window size etc.

Feature vector for flow features are commonly represented in *numerical* space [17]. For example, *packet size* statistics are represented in decimal value in terms of bytes; such as 1 bytes, 10 bytes or 65,548 bytes. The range of minimum (0) to maximum packet size (65,548) defines the *feature space boundary*. For example, we can say that *SMTP* with average packet size of 300bytes can be discriminated from *DNS* with average of 500bytes. Similarly, *IAT* is used to isolate

applications based on packet arrival time disparity among some applications set (based on respective packet timestamp). However, it is less effective since *IAT's* vectors are susceptible to network noise [18, 19, 20].

4. Statistical Classification Approaches

Statistical approaches are developed to address the challenges in encrypted traffics classification. One of the early work in statistical classification proposed by [21] looks at packet & flow vectors based on *Vector Space Model* (VSM). The algorithm computes the associations among flows based on *Root-Mean-Square* (RMS) distance, *Euclidean* distance and the angle between the vectors and it achieves true positive of 90% with 7% false positives. *Multivariate Gaussian Fitting* of Multi-Scale Traffic Characteristics [22] technique is developed to account for real time traffic analysis using highly correlated features. The author defines classification scope into legitimate and illicit traffics and evaluating the work using attack simulations gives 91% of accuracy.

A semi-supervised algorithm proposed by [23] is a good show case of the capability of statistical ML classification. It employs 6 parameters to describe flows including *flow start and stop timestamp, total number of bytes, total number of packets, average packet size, average packet/byte rate* and cumulative *TCP flags* for each flow. The author defines the classification scope to 8 protocol types; that includes *Web, FTP, P2P, Streaming, Data base, Mail, Instant Messaging, VPN* and *VoIP* traffic. The classification result achieved best case of 90% accuracy for the five of the targets, but result is less desirable for the rest of the traffics classes. The scope the work is among some of the most complete, as it supports both standard protocols and user application traffic classes.

There are methods that work at sub-flow levels that claims to improve existing statistical flow based methods in real time environments. Such classifications methods [24, 25, 26, 27] marginally improved accuracy in a bounded classification scope. These works are published sequentially; thus, they use the same output traffic classes to evaluate incremental improvements. *Generic attributes* [24] are used to identify *Skype* application; it is limited to version 2, 3 & 4 and there has been no discussion on scalability towards application update. Refinements are made in *WEKA* using *J48 algorithm* coupled with new parameters including *mean packet length*, *packet length dispersion, two packet difference* and *inter-arrival time* to attain some commendable results at 97% *precision* and 86% *recall*. To account for cost efficacy, *Sinusoidal Voice Over Packet Coder* (SVOPC) [28] presents the algorithm that minimize *Skype* data requirement to 5 seconds of sampling duration while manage to achieve higher precision at 98%. *Enhanced SVOPC* [29] extend and implement *SVOPC* on wider scale, where the classification scope grows beyond *VoIP* traffics. An innovative method [30] proposed using *Chi-Square* to distinguish *Skype* from web traffic as workaround for traffic encryption; but at the tradeoff of classification granularity. Other related works like *rapid identification* [31] employs unique features to classify *non-VoIP traffic*, in particular *Bit Torrent* on this case using the characteristics *Packet Ratio, Small Packet Ratio, Large Packet Ratio* and *Smaller Standard Deviation* were used with the *C4.5 classifiers*. *Recall* of 98.2% and *precision* of 96.5% is remarkable, however the classification is again at extremely coarse level which is only between *P2P* and non-P2P traffics classes.

Statistical approaches have grown to accommodate more diversity of traffic classes in recent times. The first 20 packets statistics are integrated with *Markov-model* [3] in small-scale mobile applications classification and achieved up to 100% of accuracy. Similarly, [32] *HMM* is used on packet size and *IAT* sequences to classify *BitTorrent, FTP, POP3, QQ, DNS* and *SSL* with best case precision of 100%. *ClassifyDroid* [33] extends the traffic scope in [3] to 200 instances; using some API level features that is more difficult in training data construction. *Youtube* and some other streaming services are the next popular single class traffic classification after *Skype*. Notable works on these include *Youtube QoE Estimation* [34] and *online video flows classification* [35] which use *throughput, packet size* and *IAT* statistics and achieved up to 83.8% of detection.

Despite the observed accuracy improvement in sub-flow discipline, it remains an open question on the impact of growing traffic classes diversity on the performance and scalability of these aforementioned methods using flow statistics features. We summarize some of prominent statistical approaches not discussed in table 2 due to space limitation. Note that the reported accuracy is empirical to respective scopes.

Table 2 Traffic classification approaches using flows statistics features

Methods	Features	Scope	Accuracy
Bayesian Analysis [18]	Flow duration, TCP Port, IAT, payload size, bandwidth entropy, Fourier transform on IAT	By Application Category: Bulk, Database, Interactive, Mail, Services, WWW, P2P, Attack, Games, Multimedia	>95%
Class-Of-Service Mapping [12]	Variance, RMS, size of packet, flow duration, mean data per flow, connection level info, IAT variance, multi-flows correlations	By Class: Interactive (Telnet), Bulk (FTP, Kazaa), Streaming (Realmedia), Transactional (DNS, HTTPS)	>90%
Statistical Machine Learning Approach [36]	Sequential forward selection (SFS) product, packet length mean and variance, flow size & duration	FTP, Telnet, SMTP, DNS, HTTP, AOL, Napster, Half-Life	86.5%
Flow Clustering With ML [37]	IAT, byte counts, connection duraction, number of transition, idle time, packet size statistics	Statistical Induced Clusters – Not At Per Application Level	relative
Traffic Classification With Clustering [19]	Number of packets, mean packet size, mean payload size, transferred data size, IAT mean	Dns, Ftp, Http, Irc, Limewire, Nntp, Pop3, Socks	relative

Backbone Internet Traffic Profiling [38]	Port connection behaviors (srcIP, dstIP, srcPrt, or dstPrt)	Web, Dns, Email, NAT Box, Web Proxies, Crawlsers, Scanners, Exploits Traffic	87%
Bulk Data Statistics [39]	*TCP* bidirectional flags, packet size	Voip, Video, P2P	relative
Voip Classification [40]	*IAT*	G711, G723, G729 Codec	relative
Early Traffic Detection [41]	First-4 packets size statistics	Nntp, Pop3, Smtp, Ssh, Https, Pop3s, Htp, Ftp, Edonkey, Kazaa, Bittorrent, Imap, Irc, Ldap, Msn, Mysql	67%-98.9%
Cellular Traffic Classification [21]	Average packet size in the uplink downlink and direction	Browsing, Data, Voip, Video	78%
Online Streaming Traffic Detection [35]	Packet size average, *IAT* for uplink and downlink, stream distributions statistics	Online, Vod	relative
Mobile Traffic Classification [3]	Packet size, iat, flow volume, flow duration	Line, Whats'app, Youtube, Spotify, Tunein Radion, Heartstone	97%-100%
Youtube Traffic Detection With ML [34]	Packet length statistics, size of transferred data in 5s intervals, packet count statistics, interarrival time statistics, throughput, statistics, and TCP flags count	Youtube	90.87%-83.94%
TCP/IP Services Detection [42]	Time(s), number of packets (S), number of packets (c), number of data from S, number of data from C, port, IP	Ftp, Telnet	<95%

5. Payload Signature

Payload features are some highly accurate traffic discriminators based on payload content inferencing. Signature extraction is time consuming; but the problem can be addressed with automated feature learning. Payload features can be interpreted in numerous representations, including *raw bytes*, *resolved string data* and *hexadecimal values*. Classification techniques that discriminates with unique payload patterns are based on the principle that traffic classes exchange some sets of unique payload data in data communication. The structure of payload is analyzed to identify some deterministic traits correlated to traffic class behaviors. Some of these correlations properties includes bytes-to-bytes sequences, bytes distribution and frequency, dictionary words detection, and pattern matching using *k*-bytes encoding. Some typical payload signatures are shown in table 3.

Table 3 Common examples of payload signatures

Payload signature	Signature content
Common substring	http/1.1, 200, GET, POST, content-type, client-hello
Common MIME types	.xml, .jpeg, .png, .swf, .zip, .bz
Long string	from='xiaomi.com'; Set_cookie:userID=40061929; Host: static.home.mi.com
Dictionary word	Image, data, music, album, server, stream
Raw bytes	<K|0d||0a|Server:>, <s|0a|ebaystatic|03|com>, < <200> <|01||00||00||01|> <s|0a|ebaystatic|03|com> >, RqbBbwc9otuoW

We summarize the payload types of some prominent deep packet inspection classification methods in table 4.

Table 4 Deep-packet inspection based traffic classification

Methods	Features	Algorithms (Profiling)	Scope	Accuracy (local)
Static application signature [43]	P2P specific signatures	Manual signature identification & annotation	5 types of P2P applications	up to 99.99%
ACAS [44]	TCP/UDP header information	Naïve Bayes, AdaBoost, SLI-Max classifier	FTP, SMTP, POP3, IMAP, HTTPS, HTTP, SSH	up to 100%
Autograph [45]	Heuristics	Content based signature creation (COPP)	HTTP worms	Not specified

Hamsa [46]	Content based signatures	Simple greedy signature generation	Worms: Code-red II, Apache-knacker, ATPhttpd, CLET, TAPiON	up to 100%
StriD^{2}FA [47]	String-matching	Length-based regex matching (LBM)	not explicitly stated	Not specified
VS-DFA [48]	Snort & ClamAV pattern	Variable stride block matching	Snort & ClamAV supported attacks scope	Not specified
LW-DPI [49]	Number of packets, payload length	Light-weight simplified string matching	60 supported applications in L7-filter: P2P, web, chat. NM, streaming, mail, VoIP	up to 99.95% for *P2P*
SLTC [50]	Time Correlational Metrics + LASER [50]	Least common subsequence (LCS)	P2P application: Gnutella, E-Donkey, BitTorrent, Skype, Kazaa	up to 99.61% for *P2P*
LASER [51]	Packet count, minimum substring length, packet size	Least common subsequence (LCS)	Limewire, BitTorrent, Fileguri, Others	up to 90%
Bernaille Early Application Identification [41, 52]	Signature from Traffic Designer + statistical stream properties	K-mean, Gaussian Mixture Model, Euclidean space and spectral clustering on HMM	NNTP, POP3, SMTP, SSH, HTTPS, POP3S, HTTP, FTP, Edonkey, Kazaa	up to 99.9%
BLINC [6]	Graphlets behavior signature + heuristics	Dedicated string pattern matching	Web, p2p, data, network management, mail, news, chat, streaming, gaming, non-payload	up to 99.9%
SVM [53]	Not specified	Distributed Support Vector Machines	40 unspecified applications	Not specified
Content Matching [54]	Layer 4 header data	Edit Distance	Not specified	Not specified
HMM sequence [32]	Payload bytes' sequences	Hidden-Markov Model	Bittorrent, eDonkey, QQ, SSL, FTP, DNS, POP3, Skype	up to 100%
'Elaborative' Payload [11]	Payload bytes'	Not specified	Youtube, Facebook	up to 100%
API Discrimination [33]	Android API string matching	Multinomial Naïve-Bayes	200 Androids Applications	Not specified

6. Deep Packet Inspection Approaches

There are four distinct branch under DPI methods in regards to the type of pattern matching implementation, namely automaton based, heuristics based, hashing based and bit-parallelism based. Payload approaches are concerned with performance optimization and accurate classification. *Variable-Stride Multi-Pattern Matching Algorithm* (VS-DFA) is classify using string matching using block oriented pattern scheme over conventional byte oriented pattern [55] to optimize memory usage. It fares better due to the variable number of byte scanned in one pass using *Winnowing* algorithm. Next, the author uses a FA construction approach, a derivative of *Aho-Corasick DFA* algorithm for pattern extraction. The algorithm is benchmarked with *Snort* & *ClamAV* pattern sets and pro-claimed less memory usage for every 3 bytes of character pattern. A two-stage self-learning traffic classifier (SLTC) is proposed to classify P2P traffic based on flow timing correlations [51]. The first stage use *Time Corrélation Matrix* (TCM) to distinguish P2P flows from mass traffic; High-Speed Monitors (HSM) is deployed to speed up payload signature based classification. The second stage infers on connection patterns and direction to deduce P2P application for remaining unclassified flows. Classified *Rabin-Karp* with Binary search and Two-level hashing (CRKBT) [56] is another string matching algorithm that emphasis on computational performance optimization, while preserving classification accuracy and completeness. The author evaluates the enhanced *RKBT* algorithm and evaluates the improvement on *ClamAV*, *DansGuardian* and *Snort* package and claimed noticeable speed enhancement across different packages.

Length-Based Matching (LBM) with accelerating scheme for RegEx matching, a variation of RegEx matching that use enhanced *Dual-Finite Automata* (DFA) called *Stride-DFA (StriD^{2}FA)* is proposed to speed up pattern matching [47]. The algorithm takes cues from data compression techniques, it first converts byte stream into integer stream in form of Stride-Length (SL) before feeding it to StriD^{2}FA. The novelty is mainly observed in improved computation speed and reduce memory consumption. However, the compression is a lossy process and some wrongly discarded data may result in erroneous classification. A lightweight DPI [49] is proposed with the objective to minimize DPI induced performance bottleneck. The algorithm selectively processes random payload unit among all flows, and randomly read partial segments of individual payload to reduce operational cost. The author uses readily available signature from L7-filter [51] that covers for 60 applications definition. The author claims that computation overhead is significantly reduced without any accuracy tradeoff. A simple pattern matching using first 2 Bytes of payload [57] with some inferred statistical vector is proposed for user-centric application classification. The algorithm works by checking on the similarity index of some baseline vectors against the

extracted vector from unknown sequential flows to distinguish application classes. The scope of classification includes *BitTorrent*, *Emule*, *YouTube*, *Fileguri* and *Afreeca*. The author claims 98% accuracy can be achieved using just 2 bytes of data in small-scale classification; although the stability towards bigger scope and unknown flows are not accounted for. A generic content matching algorithm based on edit distance is proposed to evaluate payload structure effectively. *Content-based pattern matching* [54] introduces the concept of sliding windows to adaptively shift the *k*-bytes length of payload data based on distance calculations to minimize signature size. *APSC system* [32, 10] argues that temporal cues are useful for discriminating traffic class; it proposes using the sequence correlations of payload bytes instead of payload itself for signatures construction. *Multi-level signature* [11] solve the trivial problems in payload inference methods such as dictionary words and common strings using 3-level interpretation; on packet, flow and connection level respectively. Lastly, *ClassifyDroid* [33] is statistics-payload hybrid method developed for large-scale Android applications classification. The author claims that modern applications invoke multiple functions call and these behaviors are observable in *android_sdk* or *API* call patterns.

7. Dataset Description

Client networks in modern days contains of traffic classes multiple diversity. We collect mobile (*iOS*) applications traffics data in large-scale to evaluate the robustness of existing traffic classification methods in context of increased diversity. The data collection period span over 1 hours of capture per traffic classes for 50 applications on *MYREN* network with *expectation-maximization (EM)*. Application instances are selected based on regional (MY) *Apple's Appstore* top 50 popularity. Training data are annotated with per-class label manually. The dataset does not include standard protocols; all members in the scope runs on ephemeral ports. Each traffic class correspond to a unique entity at application level, rather than categorical. The exhaustive compositions of supported traffic classes are shown in table 5.

Table 5 Datasets compositions of considered traffic classes

Exhaustive Compositions of Supported Traffic Classes				
Facebook	Facetime Audio	Waze	Clash of Clan	Steam
Appstore	Facetime Video	Groupon	Mudah.my	Ebay
Lazada	Skype	Google Drive	11street	Google Maps
Amazon	Wechat	Hulu	Shopee	Uber
miHome	Whatapp	Pokemon Go	Carousell	Pandora
Spotify	Messenger	Soundcloud	Zalora	Pinterest
Youtube	Line	Minecraft	Taobao	Gmail
iFlix	Snapchat	Wish	Lelong.my	Super Mario
Netflix	Instagram	Tinder	AliExpress	Ikea
Apple Music	Twitter	Go Shop	Gemfive	Tripadvisor

8. Evaluation Methodology

We use *Naïve-Bayes (NB)* classifier on default hyperparameters to evaluate the robustness of existing feature sets in large-scale classifications. *Naïve-Bayes* is used throughout all evaluations to place result's emphasis on features attributes rather than on classifiers attributes. Packets data are reassembled to flows to satisfy flow features pre-exquisite. Training data are constructed based on flow and payload features using *Tshark*. Traffic classification is performed on *10-fold* cross validation method. We use *CFS attribute selector* based on *Pearson-Correlation* for feature attributes evaluation. First, we evaluate and rank individual features performance for the 50 applications of interest. Features ranking is useful to estimate the extent of diminishing feature's performance against increased traffic classes and behavioral dynamism. Next, we evaluate classification performance of features set in some prominent classification methods in the new problem space. In this test, we are interested to determine does existing features have sufficient feature vectors to accommodate classes diversity growth. Then, we identify the most optimal feature currently available in respect to individual traffic classes using *NB*. The derived findings are useful for targeted feature selection corresponding to application classes of interest. All of these evaluations are implemented in *WEKA*; we use *true positive (TP)* as the accuracy metric.

9. Single-feature Attribute Evaluation

In this section, we evaluate the impact of increased traffic classes on classification accuracy using flow statistics features and payload signature. The features of interests are derived from table 1 and 3. We summarize the attained accuracy (*TP)* in table 6.

Table 6 Classification accuracy of respective flow statistics features against increasing diversity

Feature	Diversity Count (*n*)				
	n=10	*n=20*	*n=30*	*n=40*	*n=50*
Packet Size	38.85	37.78%	33.29%	31.33%	30.19
Average Packet Size	47.38	44.07%	30.17%	27.55%	22.11
Max Packet Size	12.68	12.22%	10.07%	9.54%	7.10
Packet Size Median	22.83	22.22%	25.88%	23.72%	21.77
Packet Size Standard Deviation	36.96	32.22%	29.33%	25.78%	22.35
Average Segment Size	27.55	23.33%	19.86%	15.12%	12.89
Inter-arrival Time (*IAT*)	38.96	20.00%	13.36%	12.28%	7.33
Average Payload Size	28.80	27.16%	27.33%	26.97 %	26.32
Payload Size SD	43.22	42.36%	43.35%	42.28%	42.70
Packet Count per Flow	23.36	21.53%	17.22%	14.38%	14.23
Packet Count with PUSH flags	41.10	38.72%	39.23%	39.18%	29.58
Flow Duration	32.33	21.11%	23.21%	20.88%	20.17
First 4 Payload Bytes	71.11	70.63%	70.88%	69.72%	72.19
First 20 Payload Bytes	72.43	71.55%	70.73%	68.28%	65.12
P2P signature	0.00%	0.00%	0.00%	0.00%	0.00%
ACAS signature	0.00%	0.00%	0.00%	0.00%	0.00%
Autograph signature	0.00%	0.00%	0.00%	0.00%	0.00%
LASER signature	8.07	8.12	7.33	8.19	7.79
BLINC signature	38.8	23.1	31.72	23.33	23.10
Payload HMM signature	72.12%	76.41	77.18	65.53	59.88
API calls signature	58.38	62.25	44.32	42.96	42.46

Based on collective feature specific accuracy, we can imply that no single feature is capable to achieve more than 90% accuracy. Thus, the needs for feature attributes evaluators such as *CFS* to select some high-synergy features into feature sets based on problem context. Payload signature is more prone to hit-and-miss; it is observed that classification with signatures achieve better accuracy than flow statistics counterparts when they are working. However, in some cases none of the traffic classes can be classified as in *ACAS* and *Autograph*. The rationale is that payload signature is hard-coded to some corresponding class; thus, they are not discriminative outside homogenous domain. Implicitly we know that signature accurateness come with increased computational complexity and are less universal. Besides, previously unique payload data patterns (as in *LASER*) show sign of diminishes over time due to application update that provoke behaviors changes. *HMM* payload sequence integrates temporal cues to improvise static payload deduction has seen significant success is preserving classification accuracy. The ramification is that *HMM* sequence exploits on temporal cues; which is a universal element to all traffic classes. Specifically, using certain generic payload attributes such as first-*k* bytes that are common to all traffic classes rather than identifying some class specific payload traits improved signature cross-domain scalability. Meanwhile, statistical features are unsparingly less accurate in relative terms due to lack of per-class specific traits. Despite that, the universal properties of statistics enable pervasive classification and we do not observe any case of failed classification. *Packet size* achieved comparably better accuracy than *IAT*, probably because the latter is more susceptible to network conditions such as congestion, retransmission and delay. Some techniques use *advance statistics*, *Fourier-Transform* and *entropy* to enhance network metrics discrimination and enjoy considerable accuracy improvements. Some flow or connection-level features like *flow duration* are defined to interpret traffic behaviors traits at higher abstraction level; on the assumptions that flow view are better cues are better discriminators than packet level cues. The principle is consolidated based on the resulting 32.23% classification accuracy achieved on *flow duration statistics*; for example, *average packet size* discriminates better than primitive *packet size*. Flow features have taken strides in recent works

to address the growing amount of encrypted traffic classification despite comparably inferior classification performance. In summary, the unique advantage of flows statistics and signature features made feature selection in respect to problem context challenging and interesting at the same time.

We observed that the increased traffic classes diversity is directly correlated to classification performance of some subset of features. The finding is evident in flows statistics features that show consistent decline in TP over diversity size increase; despite at some non-deterministic magnitude. The diminishing effect extends to some non-hardcoded payload features such as in *hmm* payload sequence signature. Apart of this, the rest of payload signatures are immune to traffic classes growth and with fairly consistent accuracy throughout multi-variate application count test. The rationale is explained with *Jaccard index (J(A,B))*; *Jaccard* coefficient measures similarity between finite sample sets, and is defined as the size of the *intersection* divided by the size of the union of the sample sets. We define *Jaccard index* in general notion as

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cup B|}$$

The accuracy in traffic classification is given by feature's variance; and these variances become more distinguished when *Jaccard index* is higher. The increased traffic class diversity introduces more potential for cross-classes behavioral similarity. This results in decreased non-intersection zone that carries the per-class unique attributes traits; and subsequently leads to feature vector collision. In summary, further optimization in feature learning department is needed to methodologically identify better correlated features to accommodate for the increased traffic diversity.

10. Evaluating Robustness of Existing Classifier Model

The robustness of current classification methods on mobile traffic classification is evaluated in this section. Specifically, we use the feature sets and classifier algorithm found in existing works to construct some new classifier models in respect to mobile traffic classes. Robustness in this context measures how well is the classification performance when being extended to the diversified scope. In this test, the relative contribution in accuracy performance is given by features and classifiers collectively. Note that the evaluation shown is not exhaustive due to features set in certain works are not publicly available. Table 7 shows the accuracy performance of some prominent classification works on local scope compared to mobile traffic scope.

Table 7 Comparison of accuracy of existing traffic classification methods in homogenous and heterogenous scope

Methods	Classifiers	Features	Accuracy (homogenous)	Accuracy (heterogenous)
Weighted Features [58]	Autoclass	Forward-Pkt-Len-Var, Backward-Pkt-Len-Var, Backward-Bytes, Forward-Pkt-Len-Mean, Forward-Bytes, Backward-Pkt-Len-Mean, Duration, Forward-IAT-Mean	0.865	0.627
ClassifyDroid [33]	Multinomial Naïve-Bayes	C-API	0.990	0.595
Auto-Android [53]	Multinomial Naïve-Bayes	C-API	0.850	0.795
H-SOM [17]	String-matching	POP3, SMTP, IMAP, HTTP, Soul-Seek, BitTorrent	1.000	0.000
SLTC [51, 40]	Time-Correlation Metric	L7 signature	0.950	0.313
Bayesian Technique [18]	Naïve-Bayes Kernel Estimation	Flow duration, TCP port, IAT, payload size statistics, Fourier-transform on IAT, bandwidth entropy	0.950	0.689
Class-of-Services Mapping (CSM) [12]	QDA, LDA	Mean packet size, variance, RMS size, flow duration, flow size, connection properties, multi-flows statistics	0.430	0.727
Port-View [14]	K-means	Payload size/average size of first-forth packets, average IAT	0.989	0.646
Traffic Clustering [19]	K-means, Autoclass	Total number of packets, mean packet size, mean payload size excluding headers, number of bytes transferred. mean inter-IAT of packets	0.935	0.629
TCP/IP Detection [42]	Multilayered Feed Forward Neural Networks	time(s), number of packets (S), number of packets (c), number of data from S, number of data from C, port, IP	0.950	0.000
Nmap [47]	Pattern-matching	Hardcoded (140) protocols signature	0.990	0.000
Packet Sequences [32]	Pattern-matching	HMM on payload bytes	1.000	0.819
Multi-level Signature [11]	Pattern-matching	Multi-level payload signature	0.910	0.345
Early Detection [41]	Naïve-Bayes	Source IP:port, destination IP:port	0.900	0.727

HMM-Statistics [3]	K-means, GMM	HMM on packet size statistics	1.000	0.823
Youtube QoE Estimation [34]	OneR, Naïve-Bayes, J48, Random Forest	Packet size and throughput statistics	0.890	0.000

Traffic classification approaches are optimized for specific traffic classes composition; that explains the high accuracy consistently being attained in classification on local scope. We find an average of 0.9-1.0 of accuracy for most methods except for *CSM* [12] at 0.43; regardless of the size of supported traffic classes. Implicitly classification is most optimal when the classifier model is specifically trained in respect to the problem scope. The rationale is that optimal features set and classifiers can be selected in respect to per-class with variable compositions. Based on this finding, we are interested to know how well these classifier models performed in heterogenous environment. We note that true effective accuracy is composite product of features correlations to traffic classes and classifier machine learning's efficacy. For simplicity, we assumed classifiers attributes impact is to be minimal for the rest of the discussions.

Corresponding to this, we evaluate the classification accuracy on our dataset using the features and classifier choice of existing methods. We observed deterministic accuracy decline between local classes and heterogenous classes based on collective performances. Worst case accuracy is 0.000 as in [42, 47, 34] while best case is given by [3] at 0.823. On examination, we justified the performance based on the quality of features used in [3] which account for temporal behaviors; in addition to the close similarity of signature scope to our dataset compositions. We observed that flow statistical methods collectively achieved accuracy in the range of 0.60-0.70. This imply that flow features are more universal and can be easily adopted for multi-classes traffic classification with heterogenous compositions. Incidentally the accuracy disparity among statistical-based approaches [18, 14] and [19] are presumed to be from variable classifiers efficacy. We identified a common trait for methods [42, 47] and [17] that achieved 0.000 accuracy; that they are all using payload signature for traffic discrimination. Our investigation shows that payload features are hard-coded for specific classes of interest; thus, they are less discriminative for non-homogenous classification. However, we notice that the classifier model built from [33] Android applications payload features are capable to classify our iOS classes at 0.595 accuracy. We examined the features and find some common payload traits for some sets of applications at categorical level. Thus, we imply that payload features have some non-deterministic cross-context value despite not having 1-to-1 signature matching of training to testing classes. Another example is seen on [11] with 0.345 accuracy on iOS traffics despite only having payload signature for 2 classes. Our investigation shows that both of these signatures correspond to a subset of the most dominant traffic classes in our dataset; namely *Youtube* and *Facebook*. Thus, we imply that classification accuracy is partly influenced by the classification performance on dominant traffic class if the instances of traffic classes are not equally distributed. Lastly, we also observed that method with flow features performed better than payload counterparts; possibly due to encrypted flows in multiple iOS application classes. In summary, we conclude that classifier model is most effective when they are trained and tested on homogenous data. In summary, we conclude that existing traffic classification approaches perform averagely for large-scale traffic classification regardless of underlying features types. Further research in the domain of feature learning and classifier optimization is needed to accommodate the diversification of traffic classes for accurate classification.

11. Future Direction

We propose some research prospects to address the growing diversity of traffic types as below:

Context-aware – traffic classification can adaptively select optimal features and classifiers to suite problem needs; such as swapping *CFS* for *PCM* in feature selection, or Naïve-Bayes for *J48* in classification as the algorithm sees fit.

Machine-learning (ML) – ML techniques have been widely used in traffic classification; however, the features used in classification are extracted with heuristics and yet to benefit from ML. Renewed emphasis of ML in feature-learning for unsupervised and scalable signature extraction is important to deal with growing amount of traffic classes

Active traffic classification – Assuming traffic diversity is increasing at a rate faster than the identification of new features; active traffic classification can ease some of the challenges, such as using agent or protocol to request application information when they talk on the network.

Fine-grain – flow statistics and signature can be interpreted at application functionalities level instead of at per-application level to address their diminishing feature space. The rationale is that user applications performed a series of actions that show behavioral disparity; and this information can be exploited to deduce application identity.

Two-tier classification – traffic classification can use existing features to classify application at categorical level first; and reuse the feature sets to classify individual application among the same category to logically expand feature space

Search space – existing feature extraction techniques source for feature in flow of application of interest itself; and do not consider neighboring protocols that are highly correlated to the applications such as *TLS* [59] and *HTTP*

Connection-level – the proliferation of IoT [60] enriched the communication patterns of applications that can be exploited to discriminate traffic classes; such as *push/pull* behaviors, service discovery patterns and its corresponding access sequences.

12. Conclusions

This paper evaluates the performance of existing flow-statistics and payload features on large-scale mobile traffic classifications. We synthesized some common features in traffic classification domains and discussed their respective advantage in dealing with diversified and dynamic user application's behavioral traits. First, existing classification methods are categorized in terms of respective features set and classifiers algorithm. Next, we compare the average

accuracy of individual flow features and payload signature in large-scale contexts. Our findings show that signatures are relatively more accurate than flow-statistics but are comparably more expensive and ineffective on encrypted data; vice versa. Then, we experiment existing classifier model on self-collected iOS dataset containing 50 unique traffic classes. We observed diminished classification performance review-wide due to the emerging diversity and behavioral dynamic challenges. Lastly, we proposed using machine learning in feature engineering department in future research to identify better correlated features for accurate user application traffic classification.

13. Acknowledgement

This work is supported by MyBrain 15 PhD and National Advance IPv6 Centre of Excellence.

References

[1] S. Ayad, O. Kazar, N. Benharkat and L. Terrissa, "Cross-Layer Routing Based on Semantic Web Services Discovery with Energy Evaluation and Optimization in MANET," *International Journal of Communication Networks and Information Security (IJCNIS)*, vol. 8, no. 1, pp. 47-56, 2016.

[2] J. Liu, J. Liu, H. Li, H. Zhu, N. Ruan and D. Ma, "Who Moved My Cheese: Towards Automatic and Fine-Grained Classification and Modeling Ad Network," in *Global Communications Conference (GLOBECOM), 2016 IEEE*, 2016.

[3] I.-C. Hsieh, L.-P. Tung and P. B.-S. Lin, "On the classification of mobile broadband applications," in *Computer Aided Modelling and Design of Communication Links and Networks (CAMAD), 2016 IEEE 21st International Workshop on*, 2016.

[4] T. Schreck, D. A. Keim and C. Panse, "Visual feature space analysis for unsupervised effectiveness estimation and feature engineering," in *In ICME, IEEE*, pp. 925-928, 2006.

[5] T. T. T. Nguyen and G. Armitage, "A Survey of Techniques for Internet Traffic Classification using Machine Learning," in *IEEE Commun. Surveys & Tutorials*, Vol. 10, No. 4, pp. 56-76, 2008.

[6] T. Karagiannis, K. Papagiannaki and M. Faloutsos, "BLINC: Multilevel Traffic Classification in the Dark," in *Proc. of the Special Interest Group on Data Communication conference (SIGCOMM) 2005*, 2005.

[7] J. Dong, N. Karianakis and D. Davis, "Multi-View Feature Engineering and Learning," in *In: Proceedings of IEEE CVPR*, 2014.

[8] U. Khurana, D. Turaga, H. Samulowitz and S. Parthasrathy, "Cognito: Automated Feature Engineering for Supervised Learning," in *Data Mining Workshops (ICDMW), 2016 IEEE 16th International Conference on*, 2016.

[9] C. Thornton, F. Hutter, H. H. Hoos and K. Leyton-Brown, "Auto-WEKA: combined selection and hyperparameter optimization of classification algorithms," in *In Proc. 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 847-855, 2013.

[10] A. Dainotti, W. D. Donato, A. Pescape, S. Rossi and et al., "Classification of Network Traffic via Packet-Level Hidden Markov Models," in *In Proceedings of GLOBECOM, IEEE*, pp. 1-5, 2008.

[11] Y.-H. Goo, K.-S. Shim, S.-K. Lee and M.-S. Kim, "Payload Signature Structure for Accurate Application Traffic Classification," in *Network Operations and Management Symposium (APNOMS), 2016 18th Asia-Pacific*, 2016.

[12] M. Roughan, S. Sen, O. Spatscheck and N. Duffield, "A Statistical Signature-based Approach to IP Traffic Classification," in *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, Taormina, Sicily, Italy, 2004.

[13] M. A. Hall, Correlation-based Feature Selection for Machine Learning, Hamilton, NewZealand: Department of Computer Science, The University of Waikato, 1999.

[14] G. Cheng and Y. Tang, "PortView: Identifying Port Roles based on Port Fuzzy Macroscopic Behavior," in *Journal of Internet Services and Applications*, 2013.

[15] S. Kraijak and P. Tuwanut, "A survey on internet of things architecture, protocols, possible applications, security, privacy, real-world implementation and future trends," in *Proceedings of ICCT*, pp. 26-31, 2015.

[16] P. Gupta and N. McKeown, "Algorithms for Packet Classification," in *IEEE Network*, 2001.

[17] R. G. Goss and G. S. Nitschke, "Automated network application classification: A competitive learning approach," in *Proceedings of the IEEE Symposium Series on Computational Intelligence (IEEE SSCI 2013)*, 2013.

[18] A. Moore and D. Zuev, "Internet traffic classification using Bayesian analysis techniques," in *ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS)*, 2005.

[19] J. Erman, M. Arlitt and A. Mahanti, "Traffic Classification Using Clustering Algorithms," in *MineNet '06: Proc. 2006 SIGCOMM workshop on Mining network data*, pp. 281-286, 2006.

[20] P. Chhabra, A. John and H. Saran, "PISA: Automatic Extraction of Traffic Signatures," in *4th International IFIP-TC6 Networking Conference*, 2005.

[21] P. Piskac and J. Novotny, "Using of time characteristics in data flow for traffic classification," in *in Proc. 5th international conference on Autonomous infrastructure, management, and security: managing the dynamics of networks and services, ser. AIMS'11. Berlin, Heidelberg: Springer-Verlag*, pp. 173-176, 2011.

[22] E. Rocha, P. Salvador and A. Nogueira, "Detection of Illicit Network Activities Based on Multivariate Gaussian Fitting of Multi-Scale Traffic Characteristics," in *in Communications (ICC), 2011 IEEE International Conference on*, pp. 1-6, 2011.

[23] F. Risso, A. Baldini and F. Bonomi, "Extending the NetPDL Language to Support Traffic Classification," in *in Global Telecommunications Conference, GLOBECOM '07. IEEE*, pp. 22-27, 2007.

[24] A. B. Mohammed and M. N. Sulaiman, "Near Real Time Online Flow-Based Internet Traffic Classification Using Machine Learning (C4.5)," in *International Journal of Engineering (IJE)*, pp. 370-379, 2009.

[25] M. N. Sulaiman and A. B. Mohd, "Towards a Flow-based Internet Traffic Classification for Bandwidth Optimization," in *International Journal of Computer Science and Security (IJCSS)*, pp. 146-153, 2009.

[26] G. Vennila, N. S. Supriya and M. Manikandan, "Navie Bayes Intrusion Classification System for Voip Network Using Honeypot (Research Note)," in *Ije Transactions A: Basics*, Vol. 28, pp. 44-51, 2015.

[27] M. Rawlins and A. Gordon-Ross, "An application classification guided cache tuning heuristic for multi-core architecture," in *In Proceedings of the Asia and South Pacific Design Automation Conference (ASP-DAC)*, Piscataway, NJ, pp. 23-28, 2012.

[28] P. A. Branch, A. Heyde and G. J. Armitage, "Rapid identification of Skype traffic flows," in *in Proc. 18th international workshop on Network and operating systems support for digital audio and video, ser. NOSSDAV '09. :*

ACM, New York, NY, USA, pp. 91-96, 2009.

[29] L. H. Do and P. Branch, "Real Time VoIP Traffic Classification," in *CAIA, Tech. Rep.*, 2009.

[30] E. Freire, A. Ziviani and R. Salles, "Detecting skype flows in web traffic," in *in Network Operations and Management Symposium, NOMS. IEEE*, pp. 89-96, 2008.

[31] J. But, P. Branch and T. Le, "Rapid Identification of BitTorrent traffic," in *in Local Computer Networks (LCN), 2010 IEEE 35th Conference on*, pp. 536-543, 2010.

[32] Z. Yuan, Y. Xue and Y. Dong, "Harvesting unique characteristics in packet sequences for effective application classification," in *Communications and Network Security (CNS), 2013 IEEE Conference on*, 2013.

[33] F. Dong, Y. Guo, C. Li, G. Xu and F. Wei, "ClassifyDroid: Large scale Android applications classification using semi-supervised Multinomial Naive Bayes," in *Cloud Computing and Intelligence Systems (CCIS), 2016 4th International Conference on*, 2016.

[34] I. Orsolic, D. Pevec, M. Suznjevic and L. Skorin-Kapov, "YouTube QoE Estimation Based on the Analysis of Encrypted Network Traffic Using Machine Learning," in *Globecom Workshops (GC Wkshps), IEEE*, 2016.

[35] R. Nossenson and S. Polacheck, "On-Line Flows Classification of Video Streaming Applications," in *Network Computing and Applications (NCA), 2015 IEEE 14th International Symposium on*, 2015.

[36] M. Hirvonen, "Two-Phased Network Traffic Classification Method for Quality of Service Management," in *Master's thesis, University of Oulu, The Department of Electrical and Information Engineering*, 2009.

[37] A. McGregor, M. Hall, P. Lorier and J. Brunskill, "Flow Clustering Using Machine Learning Techniques," in *In PAM*, 2004.

[38] K. Xu and Z.-L. Zhang, "Supratik Bhattacharyya, Profiling internet backbone traffic: behavior models and applications," in *Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, Philadelphia, Pennsylvania, USA, 2005.

[39] B. Kurt, A. T. Cemgil, M. Mungan, N. Polat and et al., "Network management without payload inspection: Application classification via statistical analysis of bulk flow data," in *in 2012 Future Network & Mobile Summit (FutureNetw)*, Berlin, 2012.

[40] T. Yildirim and P. J. Radcliffe, "VoIP Traffic Classification in IPSec Tunnels," in *Proc. Int'l Conf. Electronics and Information Eng. (ICEIE '10)*, Vol. 1, pp. 151-157, 2010.

[41] L. Bernaille, R. Teixeira and K. Salamatian, "Early Application Identification," in *Proceedings of the 2006 ACM CoNEXT conference*, Portugal, 2006.

[42] K. M. C. Tan and B. S. Collie, "Detection and classification of TCP/IP network services," in *Computer Security Applications Conference, Proceedings., 13th Annual*, pp. 99-107, 1997.

[43] S. Sen, O. Spatscheck and D. Wang, "Accurate, scalable in-network identification of p2p traffic using application signatures," in *Proceedings of the 13th international conference on World Wide Web*, New York, NY, USA, 2004.

[44] P. Haffner, S. Sen, O. Spatscheck and D. Wang, "ACAS: automated construction of application signatures," in *MineNet '05 Proceedings of the 2005 ACM SIGCOMM workshop on Mining network data*, pp. 197-202, 2005.

[45] H.-A. Kim and B. Karp, "Autograph: toward automated, distributed worm signature detection," in *Proceedings of the 13th conference on USENIX Security Symposium*, San Diego, CA, pp. 19-19, 2004.

[46] Z. Li, M. Sanghi, Y. Chen, M.-Y. Kao and B. Chavez, "Hamsa: Fast Signature Generation for Zero-day Polymorphic Worms with Provable Attack Resilienc," in *IEEE Symposium on Security and Privacy*, 2006.

[47] X. Wang, J. Jiang, Y. Tang, B. Liu and X. Wang, "StriD2FA: Scalable Regular Expression Matching for Deep Packet Inspection," in *Communications (ICC) 2011 IEEE International Conference on*, pp. 1-5, 2011.

[48] N. Hua, H. Song and T. V. Lakshman, "Variable-Stride Multi-Pattern Matching For Scalable Deep Packet Inspection," in *In Proc. of IEEE INFOCOM 2009*, pp. 415-423, 2009.

[49] S. Fernandes, R. Antonello, T. Lacerda, A. Santos, D. Sadok and T. Westholm, "Slimming Down Deep Packet Inspection Systems," in *NFOCOM Workshops 2009 IEEE*, pp. 1-6, 2009.

[50] C. Park, Y. Won, M. Kim and J. Hong, "Towards Automated Application Signature Generation for Traffic Identification," in *In Proceedings of the IEEE/IFIP Network Operations and Management Symposium (NOMS 2008)*, Salvador, Brazil, pp. 160-167, 2008.

[51] R. Keralapura, A. Nucci and C.-N. Chuah, "Self-Learning Peer-to-Peer Traffic Classifier," in *Proc. 2009 Proc. 18th International Conference on Computer Communications and Networks ser. ICCCN '09*, pp. 1-8, 2009.

[52] L. Bernaille, R. Teixeira, I. Akodkenou, A. Soule and K. Salamatian, "Traffic classification on the fly," in *ACM Special Interest Group on Data Communication (SIGCOMM) Computer Communication Review*, Vol. 36, No. 2, 2006.

[53] D. L. Quoc, V. D'Alessandro, B. Park, L. Romano and C. Fetzer, "Scalable Network Traffic Classification Using Distributed Support Vector Machines," in *Proceedings-6322015 IEEE 8th International Conference on Cloud Computing CLOUD 2015*, Vol. 00, pp. 1008-1012, 2015.

[54] K. Choi, J.-k. Choi, S. Ha and S. Y. Ban, "Content-based pattern matching for classification of network application," in *Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference*, 2006.

[55] S. Schleimer, D. S. Wilkerson and A. Aiken, "Winnowing: local algorithms for document fingerprinting," in *in Proc. 2003 ACM SIGMOD international conference on Management of data, ser. SIGMOD '03, ACM*, pp. 76-85, 2003.

[56] P.-C. Lin, Y.-D. Lin, Y.-C. Lai and T.-H. Lee, "Using string matching for deep packet inspection," in *Computer*, Vol 41, pp. 23-28, 2008.

[57] J. Y. Chung, B. Park, Y. J. Won and J. Strassner, "Traffic Classification Based on Flow Similarity," in *in Proc. 9th IEEE International Workshop on IP Operations and Management, ser. IPOM '09. , Heidelberg: Springer-Verlag*, Berlin, pp. 65-77, 2009.

[58] S. Zander, T. Nguyen and G. Armitage, "Automated traffic classification and application identification using machine learning," in *IEEE 30th Conference on Local Computer Networks (LCN 2005)*, 2005.

[59] Z. E. Uahhabi and H. E. uahhabi, "Calculating and Evaluating Trustworthiness of Certification Authority," *International Journal of Communication Networks and Information Security (IJCNIS)*, vol. 8, no. 3, pp. 136-146, 2016.

[60] A.-u. Rehman, S. U. Rehman, I. U. Khan, M. Moiz and S. Hasan, "International Journal of Communication Networks and Information Security (IJCNIS)," *International Journal of Communication Networks and Information Security (IJCNIS)*, vol. 8, no. 3, pp. 147-157, 2016.