

ISP 向け低コストトラフィック分類手法の提案

後藤 崇行^{1,a)} 佐々木 力¹ 立花 篤男¹ 阿野 茂浩¹

受付日 2011年5月20日, 採録日 2011年11月7日

概要: ISP において, プロトコルごとのトラフィック量を監視することは, 効率的なトラフィックエンジニアリングやネットワーク設計のために有益である. プロトコルを正確に分類する方法として, 対象トラフィックをフルキャプチャし DPI (Deep Packet Inspection) を適用する方法が考えられるが, 専用の DPI 装置は高価であり, ISP ネットワークにおいて多地点監視を行う場合, 設備コストが増大する問題がある. そこで筆者らは, 多地点計測によるプロトコルごとのトラフィック量を低コストに監視することを目的とする. このために, パケットサンプリングと教師あり学習によるトラフィック分類手法を用いるトラフィック監視を提案する. また, 学習用トラフィックをフルキャプチャし, このトラフィックに対して複数回パケットサンプリングを適用することで得られるフロー特徴量を用いた学習を行う. 本稿では, 提案手法の詳細について述べるとともに, 公開トラフィックトレースを用いた実験により, 提案手法の有効性を示す.

キーワード: トラフィック分類, パケットサンプリング, 機械学習

Low-cost Traffic Classification Method for Large-scale ISP

TAKAYUKI GOTO^{1,a)} CHIKARA SASAKI¹ ATSUO TACHIBANA¹ SHIGEHIRO ANO¹

Received: May 20, 2011, Accepted: November 7, 2011

Abstract: Quantifying the traffic amount of each protocol is useful for ISP to perform effective traffic engineering and network designing. A straight forward approach for protocol identification is to first capture full traffic and apply DPI (Deep Packet Inspection) technology. However, it is difficult to deploy the dedicated DPI hardware to large ISP networks because the hardware is expensive and should be deployed to multiple measurement points. Therefore, we aim at low-cost monitoring of the traffic amount of each protocol at many measurement points. In order to achieve this goal, we propose a cost-effective traffic classification method which is composed from packet sampling and flow features-based traffic classification which uses supervised learning. We also propose a new learning method that uses flow features by capturing traffic data for learning fully and applying packet sampling multiple times to the traffic data. In this paper, we describe the detail of our proposed method and represent the effectiveness through the experiment with publicly available traffic traces.

Keywords: traffic classification, packet sampling, machine learning

1. はじめに

ISP (Internet Service Provider) はネットワークを流れているアプリケーションごとのトラフィック傾向に基づき, トラフィックエンジニアリングやネットワーク設計, 監視を効率的に行う必要がある. たとえば, トラフィックのアプリ

ケーションごとに要求条件が異なるため, VoIP (Voice over IP) 等の遅延に敏感なトラフィックを優先し, その他はベストエフォートに処理するといった制御により, VoIP ユーザの満足度を向上させることが考えられる. また, アプリケーションごとのトラフィック量を監視し続けることで, 特定アプリケーションのトラフィック量のみが著しく低下した場合には, そのアプリケーションに問題が生じたと判断でき, 障害時の迅速なトラブルシューティングにも有用である.

¹ 株式会社 KDDI 研究所
KDDI R&D Laboratories, Fujimino, Saitama 356-8502, Japan

^{a)} goto@kddilabs.jp

アプリケーションごとのトラフィック傾向を把握するには、各トラフィックフローのプロトコルを正確に識別可能な DPI (Deep Packet Inspection) を適用するのが一般的である。ISP バックボーントラフィックは年々増加傾向にあり、現在のバックボーンで広く用いられている 10 Gbps 回線等の大容量トラフィックを解析可能な、高価な専用の DPI ハードウェアが必要となる。また、大規模な ISP ネットワークでは多地点でトラフィックを計測しなければならない。よって、多地点かつ大容量トラフィックをプロトコルまで含めて監視するには、膨大な設備コストが必要となる。一方で、ネットワークの設備コストならびに運用コストの削減は ISP にとって重要な課題であり、本研究で対象とするトラフィック監視に関しても、低コストかつ高信頼な実現が求められている。

そこで本稿では、大規模 ISP を想定し、多地点計測でのプロトコルごとのトラフィック量を低コストに監視することを目的とする。具体的には、高価な専用の DPI ハードウェアの代わりに、安価な汎用 PC にソフトウェアをインストールすることで設備コストの抑制を実現する。しかし、汎用 PC によるソフトウェアでは大容量トラフィックを処理するのは困難であるため、パケットサンプリングを適用する。さらに、サンプリングを適用した (サンプルド) トラフィックに対しても高精度なプロトコル識別を実現するために、教師あり学習によるフロー特徴量ベーストラフィック分類手法を利用する。すなわち、提案する監視手法は、パケットサンプリングと教師あり学習によるフロー特徴量ベーストラフィック分類手法からなる。本稿では、公開トラフィックトレースを用いた実験を通じて、提案する監視手法を用いることで、安定したプロトコル識別性能が達成可能なことを示す。

以下、2 章ではプロトコル識別の関連研究を、3 章では提案する監視手法を述べる。4 章では検証実験について説明し、5 章では実験結果の考察を行う。6 章では本稿のまとめを述べる。

2. 関連研究

これまで、トラフィックフローのプロトコルを識別・推定するために、数多くのトラフィック分類手法が提案されている。これらは大まかに、ポートベース、ペイロードベース、フロー特徴量ベースの 3 種類に分類される。

ポートベース手法は、ポート番号とプロトコルの対応表 (たとえば、文献 [1]) に基づいてプロトコルを識別する手法であり、シンプルかつ高速に動作する。しかしながら、P2P ファイル共有等の使用するポート番号を動的に決定するプロトコルや、HTTP 等にカプセル化されるプロトコルを識別できないという問題がある。ペイロードベース手法は、パケットのペイロードを参照し、あらかじめ解析されたプロトコル特有のシグネチャやパターンとマッチング

することによりプロトコルを識別する。この手法は一般的に DPI と呼ばれる。シグネチャやパターンを検出するために、フローに含まれるすべてのパケットに DPI を適用する必要があるため、膨大なトラフィックを高速に処理可能な専用のハードウェア (たとえば、文献 [2]) が必要となる。このようなハードウェアは高価であるため、大規模な ISP ネットワークにおいて多地点展開することは困難である。一方、ソフトウェア実装として OpenDPI [3] が公開されているが、大容量トラフィックを汎用 PC で処理するためにパケットサンプリングを適用し、このトラフィックに対して DPI を適用すると、シグネチャやパターンが観測できなくなるため識別精度が著しく低下する。図 1 は ITOC トレース^{*1}を 1/10 パケットサンプリングし、OpenDPI を適用した結果である。フルトラフィックに比べ、1/10 サンプリングによる結果は識別不可 (unknown) が大幅に増加し、プロトコルの識別精度が低下していることが分かる。

フロー特徴量ベース手法は、機械学習 (教師ありまたは教師なし) を用い、ヘッダ情報のみから算出可能なパケットの到着間隔やパケット長等のフロー特徴量に基づいてプロトコルを分類する [4]。本稿では、フローを 5-tuple (送信元/宛先 IP アドレス, 送信元/宛先ポート番号, プロトコル番号) が同一のパケット群として定義する。教師あり学習によるトラフィック分類手法は、事前に学習用トラフィックの各フローに対してフロー特徴量とプロトコルを解析し、推定に必要な分類器を生成する (学習フェーズ)。実際のプロトコル推定時においては、対象トラフィックのフローごとの特徴量を計算し、学習時に生成した分類器にフロー特徴量を入力することによりプロトコルを推定する (推定フェーズ)。本手法は、学習で用いたトラフィックのプロトコル群が、対象トラフィックのプロトコル群を包含している場合には、推定精度が高くなるという特徴がある。教師なし学習によるトラフィック分類手法は、推定対象のフロー群を、フロー特徴量の類似度に基づきグループ化する。この手法は、未知のプロトコルを検出できる可能性があるものの、別途、各グループのプロトコル識別を行う必要がある。

フロー特徴量の算出においては必ずしもフロー内のすべてのパケットを必要としないためサンプリングの適用が有効であるが、フロー特徴量の算出精度とサンプリングレートとはトレードオフの関係にある。たとえば、サンプルドトラフィックに対してフロー内パケット数を算出する場合、サンプリングレートを p 、フルトラフィックでのパケット数を n とすると、相対誤差は平均 0, 分散 $(1-p)/(np)$ の確率変数として見積もられる [5]。そのほか、代表的なフロー特徴量とサンプリングレートとの関係は文献 [5] において議論されている。このように、一般的にはサンプリングレートを大きくすれば誤差は小さくなり、サンプルドトラ

*1 後述するように、本稿で使用するトラフィックトレースの 1 つ。

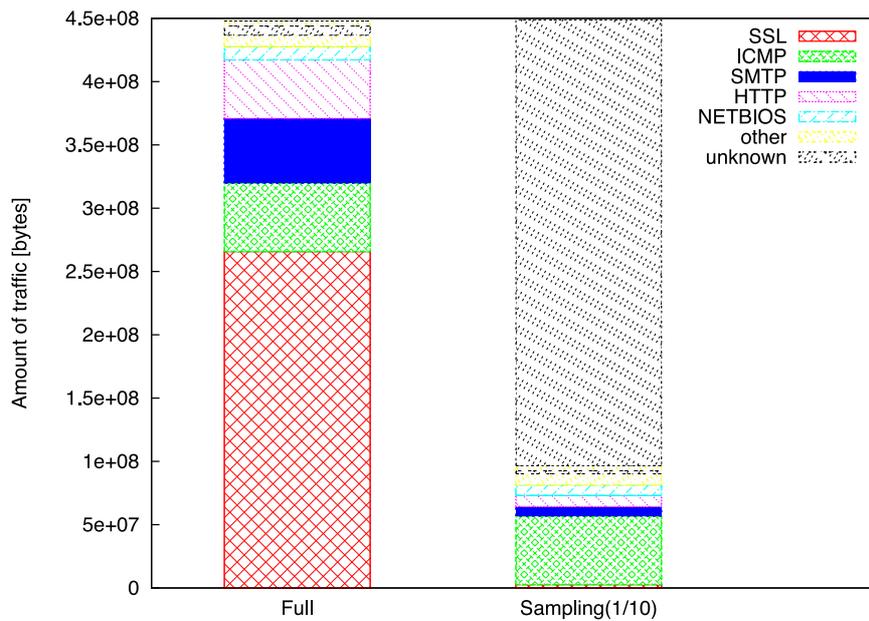


図 1 ITOC トレースの OpenDPI によるプロトコル識別結果
 Fig. 1 Result of protocol identification by OpenDPI using ITOC trace.

ヒックに対しても比較的高精度な分類を実現できる可能性がある。以上より、提案手法はフロー特徴量ベース手法を用いる。また、本稿では、事前学習が可能な状況を想定しているため、教師あり学習によるトラヒック分類に着目する。サンプルドトラヒックと教師あり学習によるトラヒック分類手法として、Carela-Espanol らの手法 [5] がある。この手法は、ルータ/スイッチに具備される Sampled NetFlow から得られるフロー特徴量を用いて学習を行う。一方、提案手法は、学習用トラヒックをフルキャプチャして多種のフロー特徴量を算出し、キャプチャしたトラヒックに対して複数回サンプリングを適用して得られるフロー特徴量を用いて学習を行う。これにより、推定性能の安定した分類器の生成が見込まれる。

3. 提案手法

低コストでプロトコルごとのトラヒック量を算出するために、筆者らはパケットサンプリングとフロー特徴量ベーストラヒック分類手法を組み合わせた監視方法を提案する。提案手法では、低コスト化実現のためにパケットサンプリングを、また、サンプリング適用時でも高精度にプロトコル分類するためにフロー特徴量ベーストラヒック分類手法を、それぞれ用いる。また、事前にトラヒック解析が可能であることを想定しているため、教師あり学習によるフロー特徴量ベーストラヒック分類手法を検討する。教師あり学習は 2 章で述べたとおり学習フェーズと推定フェーズからなるため、提案手法は以下の手順になる。

- (1) 学習フェーズ：学習用トラヒックを用いて、推定に必要な分類器を生成する。
- (2) 推定フェーズ：学習フェーズで生成した分類器を用い

て、実際にプロトコルごとのトラヒック量を推定する。それぞれについて以下で説明する。

3.1 学習フェーズ

学習フェーズでは、推定フェーズで使用する分類器を生成する。この概要を図 2 に示す。本稿では教師あり学習を用いるため、少なくとも、推定用トラヒックのプロトコル群が学習用トラヒックのプロトコル群に含まれている必要がある。そこでまず、推定対象トラヒックのプロトコル群を包含していると想定される地点において、キャプチャ装置 (Traffic capturing device) によりトラヒックをペイロードを含め計測する。具体的な地点として、推定したいリンクの 1 段階多重化されたバックボーン側のリンクがあげられる。次に、フルキャプチャしたトラヒックデータに対して、DPI 装置 (DPI device) または人的リソース (Manpower) により各フローのプロトコルを識別する。ここで、当該トラヒックはフルキャプチャされているため、フローのプロトコルを精度高く識別することが可能である。さらに、分類器ジェネレータ (Classifier generator) において、キャプチャしたトラヒックに対して、推定フェーズで使用するサンプリングレートでのランダムパケットサンプリングを適用し、サンプルドフローの特徴量 (パケットの到着間隔やパケット長等) を計算する。フルキャプチャデータを用いて高精度に識別したプロトコルとサンプルドフローの特徴量を学習させ、推定用の分類器を生成する。

ここで、同一トラヒックに対して同一サンプリングレートを用いた場合であっても、得られるサンプルドフローの特徴量は毎回異なる。すなわち、生成される分類器の性能にばらつきが生じることが想定される。このばらつきはサ

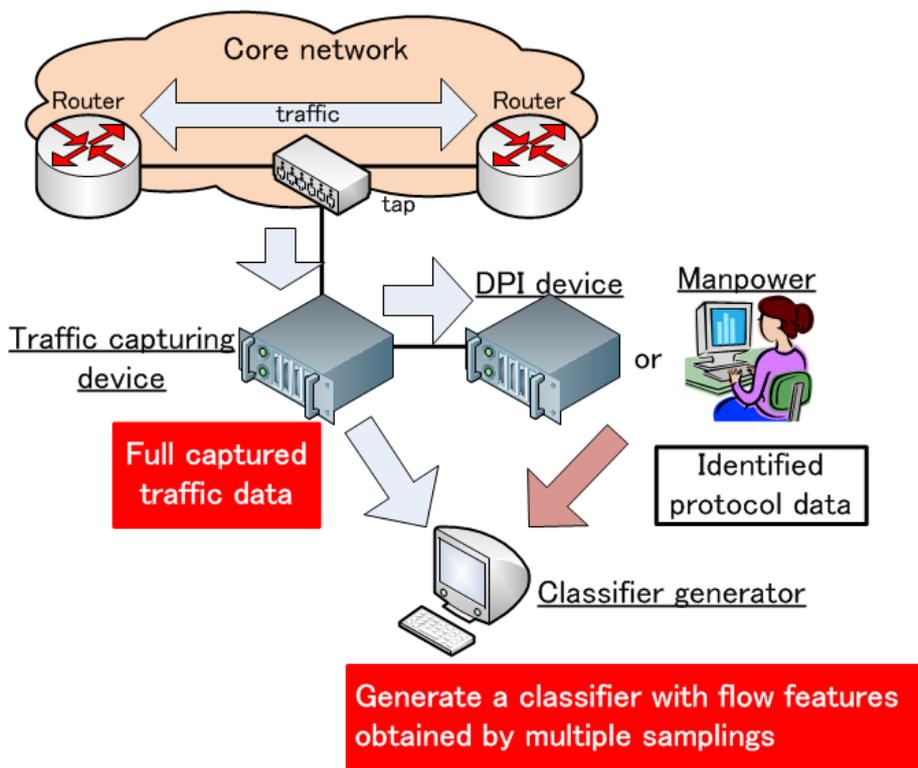


図 2 学習フェーズ

Fig. 2 Learning phase of our proposed method.

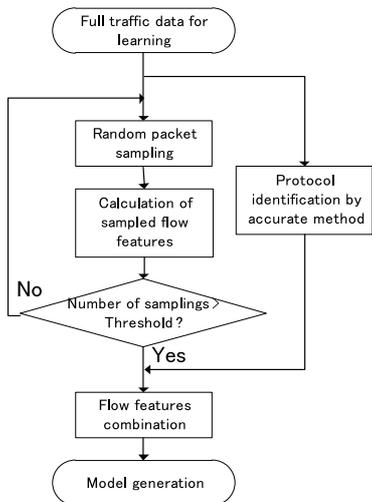


図 3 学習フェーズの処理フロー

Fig. 3 Processing flow of learning phase.

ンプリングにより学習に利用可能なフローデータが減るためであり、より多くのサンプルドフローの特徴量を学習させて分類器を生成することにより、ばらつきを抑えることが可能であると考えられる。そこで、提案する監視方法は、フルトラヒックに対して複数回サンプリングを適用し、得られるサンプルドフロー特徴量を統合することにより、より性能のばらつきが少ない分類器を生成する。この処理フローを図 3 に示す。あらかじめ定められた回数分サンプリングを繰り返し、得られる複数のフロー特徴量を統合さ

Number of samplings	Protocol	Packet size	Inter-arrival time	Duration	...
1st	HTTP	1000	12	8	...
	FTP	1400	8	4500	...
	Skype	100	3	180	...
2nd	SSL	1400	6	170	...
	Flash	1000	10	210	...
	PPLive	500	4	1890	...
Nth	HTTP	900	5	9	...
	BitTorrent	1400	15	4770	...
	ICMP	70	125	95	...

図 4 サンプルドフロー特徴量の統合例

Fig. 4 Example of combination of sampled flow features.

せることにより、分類器生成に使用するフロー特徴量を作成することが特徴である。N 回サンプリングを行ったときのフロー特徴量の統合例を図 4 に示す。図 4 のとおり、提案手法では、各サンプリングの結果得られるフロー特徴量を単純に連結させて分類器生成用のフロー特徴量とする。なお、この方法では元々同一フローが異なるサンプルドフローとして複数観測されることがありうるが、これらは別々のサンプルドフローの特徴量として扱う。

3.2 推定フェーズ

図 5 に推定フェーズの概要を示す。学習フェーズにお

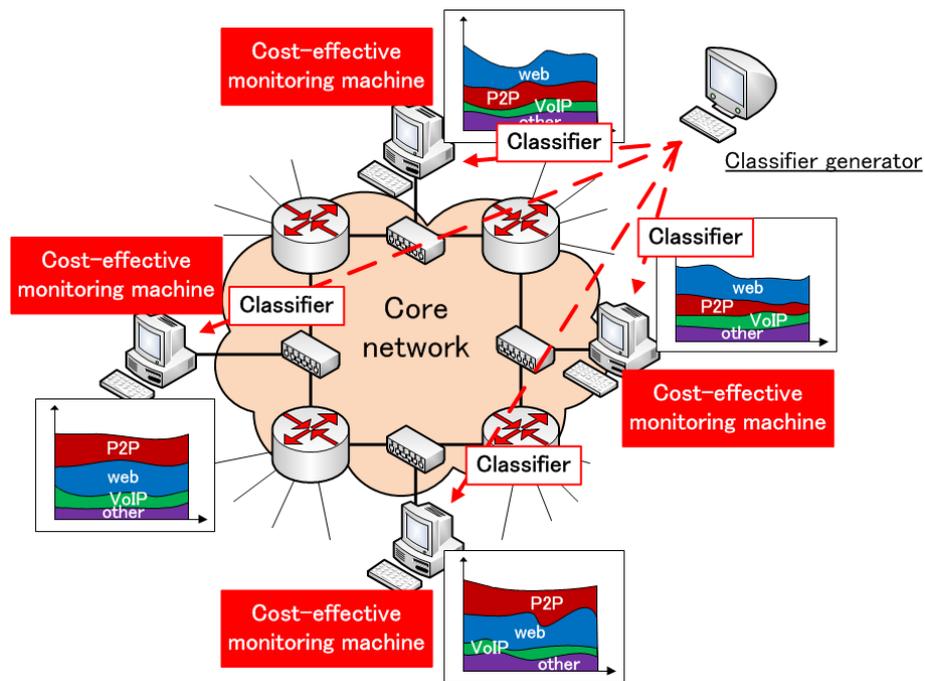


図 5 推定フェーズ

Fig. 5 Estimation phase of our proposed method.

ける分類器ジェネレータは、生成した分類器を多地点に設置されている監視装置 (Monitoring machine) に配布する。ここで、監視装置は汎用 PC で実現される。監視装置は、まず、タップ経由で入力される推定対象トラフィックを sFlow ソフトウェア [6] 等を用いてランダムパケットサンプリングし、さらにフロー特徴量に必要なヘッダのみをキャプチャする。キャプチャしたトラフィックを用いてフロー特徴量を算出し、算出されたフロー特徴量を分類器に入力することでプロトコルを推定する。プロトコル分類されたサンプルフローのバイト数を足し合わせ、プロトコルごとのバイト数を求める。そして、所望のアプリケーションごとのトラフィック量を算出するため、求めたプロトコルごとのバイト数にサンプリングレートの逆数を乗じ、プロトコルごとのトラフィック量を推定する。なお、以上の処理を基準時間ごとに行うことでリアルタイム性を実現することも可能である。すなわち、基準時間分のトラフィックすべてを対象とするわけではなく、前半でトラフィックをキャプチャし、後半でフロー特徴量を算出してプロトコルを推定し、全体として基準時間になるようにする。したがって、キャプチャするトラフィック量を抑えることによりリアルタイム処理が可能である。

4. 評価

4.1 条件

評価で用いるフロー特徴量に関して、先行研究 [7], [8], [9], [10] において様々なフロー特徴量が使用されているが、フロー特徴量を最も詳細に記述している文献 [7] を用いた。

文献 [7] では 250 種類の特徴量が定義されているが、プロトコル推定に関連のない、再送等の QoS に関連した特徴量やフローの方向 (ダウンロードまたはアップロード) を排除することにより、59 種類の特徴量を選択した。使用したフロー特徴量を表 1 に示す。

学習フェーズにおける各フローのプロトコル特定には、オープンソースの DPI エンジンである OpenDPI を用いた。本ツールは約 100 種類のプロトコル特定が可能である。2 章で述べたように、フルトラフィックが利用可能でかつプロトコル特有のシグネチャまたはパターンが既知の場合、ペイロードベース手法を用いることで正確なプロトコル特定が可能である。ペイロードベース手法は、プロトコル特有のシグネチャやパターンがあらかじめ解析されていないフローを未知 (unknown) と判断し、無理に誤って他のプロトコルとして特定しないため、ペイロードベース手法によって特定されたプロトコルは信頼性が高いといえる。したがって、OpenDPI により既知と判断されたフローのみを実験に使用した。

評価には、フルペイロードを含んでフルキャプチャされた 3 種類の公開トレース (ITOC [11], Laura [12], OP [13]) を用いた。パケット数、フロー数、OpenDPI によって識別されたプロトコル例を表 2 に示す。提案手法を評価するうえで学習用と推定用の 2 つのトレースが必要なため、各トレースを 2 つのサブトレースにフローを保ちながらフロー数が均等になるように分割し、一方を学習用に、他方を推定用にそれぞれ使用した。

教師あり学習のためのソフトウェアとして、オープンソー

表 1 教師あり学習によるトラフィック分類手法で使用するフロー特徴量

Table 1 Flow features used for traffic classification method by supervised learning.

カテゴリー	数	詳細説明
ポート	2	送信元, 宛先
パケット到着間隔	17	最小, 25-, 50-, 75-パーセンタイル, 最大, 平均, 分散, 時系列の上位 10 周波数
IP パケットサイズ	7	最小, 25-, 50-, 75-パーセンタイル, 最大, 平均, 分散
セグメントサイズ	7	最小, 25-, 50-, 75-パーセンタイル, 最大, 平均, 分散
フロー	8	パケット数, 持続時間, データ転送時間, データスループット, バルク転送モードの遷移数/合計時間/比率, バルク転送・トランザクションモードの遷移数
TCP	8	PUSH パケット数, 要求 MSS, ゼロウィンドウサイズのパケット数, URG パケット数, URG バイト数, 初期ウィンドウのバイト数/パケット数, SYN と FIN のシーケンス番号差分
TCP ACK	7	ACK 数, データのない ACK 数, SACK 数, DSACK 数, SACK ブロック数, SACK の可否 SACK 情報付きの ACK 数
TCP オプション	3	ウィンドウスケーリングの可否, ウィンドウスケーリング値, タイムスタンプの可否

表 2 使用したトラフィックトレース

Table 2 Used traffic traces.

トレース名	パケット数	フロー数	識別されたプロトコル例
ITOC	2,104,650	178,485	SSL, ICMP, SMTP, HTTP
Laura	117,994	5,096	HTTP, FTP, Flash, MMS
OP	120,549	1,891	HTTP, ICMP, Bittorrent

スの Weka [14] を用いた。本ソフトウェアは学習フェーズと推定フェーズの両方で使用される。Weka では様々な教師ありの学習アルゴリズムが使用可能であるが、演算量が比較的低い AdaboostM1 と Bagging, J48, NaiveBayes の 4 種類等が実装されている。AdaboostM1 は、学習の仕方を学習するメタ学習として AdaboostM1 [15] を使用し、その基本となる分類器として decision stump [16] を用いる方法である。Bagging は、メタ学習として Bagging [17] を使用し、基本となる分類器として情報利得に基づいて決定木を生成する REPTree を用いる方法である。J48 は、C4.5 アルゴリズム [18] を用いて生成される決定木を用いる方法である。NaiveBayes は単純ベイズ分類器 [19] を用いる方法である。評価に使用する学習アルゴリズムを選択するために、事前評価実験を行った。フルトラフィックを 2 つに分割し、片方向で分類器を生成し、もう片方向のフローのプロトコルを推定した結果を表 3 に示す。表の値は、後述する Overall accuracy (式 (1) で定義) であり、推定精度を表している。本評価では、表 3 の結果をふまえ、Bagging を用いた。Bagging はベース分類器が高速決定木の REPTree で、学習用のフローを複数回サンプリングして得られる複

表 3 フルトラフィック使用時の様々な機械学習の Overall accuracy

Table 3 Overall accuracy of various machine learning methods using full traffic data.

学習 \ トレース	ITOC	Laura	OP
AdaboostM1	0.616	0.267	0.501
Bagging	0.998	0.819	0.971
J48	0.999	0.197	0.739
NaiveBayes	0.731	0.786	0.536

数の分類器を統合するメタ学習を行うものである。

サンプリングレートとして、1/10, 1/100 の 2 種類を用いた。また、ランダムサンプリングの影響により評価結果が異なるため、各サンプリングレートに対して 10 回の評価を実施した。

4.2 評価メトリック

評価メトリックとして Overall accuracy ならびに F-measure を用いた。Overall accuracy はトレース全体の推定性能を表すメトリックであり、F-measure は Precision と Recall の平均をとって計算され、プロトコル i ごとの推

定性能を表すメトリックである。それぞれ以下のように定義される。

$$Overall\ accuracy = \frac{\sum_{i \in \{\text{Known protocols}\}} TP_i}{\sum_{i \in \{\text{Known protocols}\}} TP_i + FP_i} \quad (1)$$

$$F\text{-measure}_i = \frac{2 \times Precision_i \times Recall_i}{Precision_i + Recall_i} \quad (2)$$

where

$$Precision_i = \frac{TP_i}{TP_i + FP_i}$$

$$Recall_i = \frac{TP_i}{TP_i + FN_i}$$

Precision はあるプロトコルとして推定したフローの中で正しく識別されたフローの割合を、*Recall* はあるプロトコルに属する全フローのうち正しく識別されたフローの割合を、それぞれ表しており、大きな値であるほど性能が高いことを意味する。これらのメトリックは、TP (True Positive), FP (False Positive), TN (True Negative), FN (False Negative) から算出されるが、本稿の研究目的はプロトコルごとのトラフィック量を把握することであるため、バイト数に基づいて以下のように定義した。

- TP_p : プロトコル p のフローが p と正しく推定されたフローの総バイト数
- FP_p : プロトコル p でないフローが p と誤って推定されたフローの総バイト数
- FN_p : プロトコル p のフローが p でないと誤って推定

されたフローの総バイト数

- TN_p : プロトコル p でないフローが p でないと正しく推定されたフローの総バイト数

5. 結果と考察

5.1 サンプリング

サンプリングを適用した実験結果について述べるとともに考察を行う。具体的には、以下の4種類のフロー特徴量を用いて学習した分類器を用いて推定した結果を比較した。

- Full : フルトラヒックより計算したフロー特徴量
- Sampling : サンプリングを1回適用して得られたフロー特徴量
- Sampling : サンプリングを10回適用して得られたフロー特徴量
- Sampling : サンプリングを100回適用して得られたフロー特徴量

5.1.1 Overall accuracy

図6に *Overall accuracy* の結果を示す。横軸と縦軸は、学習フェーズにおけるサンプリング回数と *Overall accuracy* (Sampling に対しては、*Overall accuracy* の平均値) を表している。Sampling (X) とは、サンプルドトラヒック (レート X) を学習し、サンプルドトラヒック (レート X) を推定した場合を表している。すなわち、学習と推定で同じサンプリングレートを用いる。また、Full (Y) とは、フルトラヒックを学習し、フルトラヒック (Y:Full) またはサンプルドトラヒック (レート Y) を推定した場合を

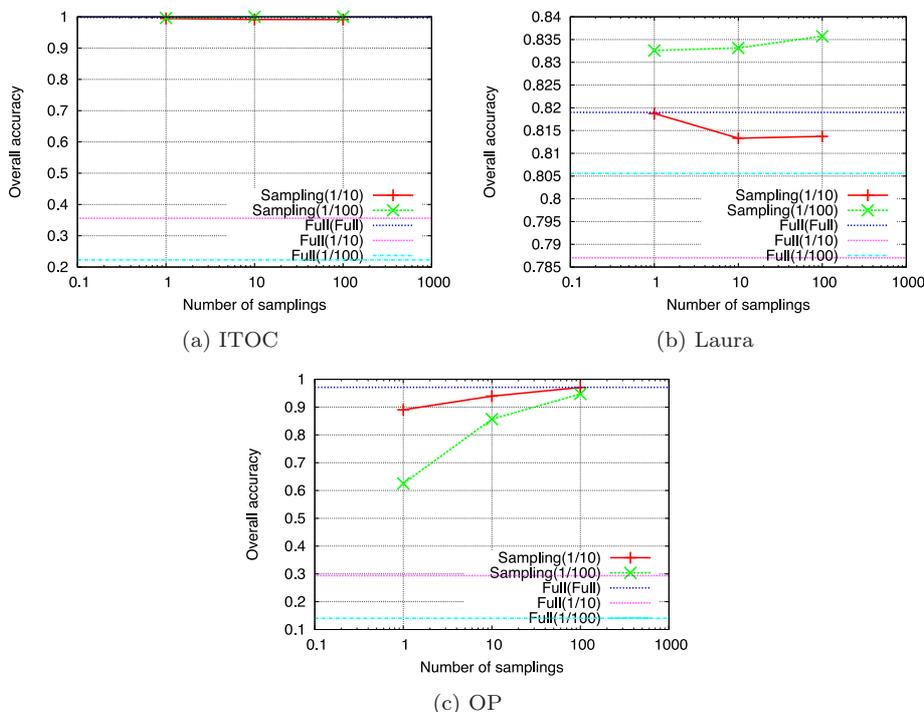


図6 Overall accuracy に基づく “Full” と “Sampling” の実験結果

Fig. 6 Experimental results of “Full” and “Sampling” patterns in terms of Overall accuracy.

表 4 情報利得に基づく推定トラフィックに有効なフロー特徴量の順位変化

Table 4 Change of ranking of flow features which is calculated using information gain.

推定用 トラフィックでの 上位 5 つ	学習用トラフィックでの順位					
	ITOC		Laura		OP	
	フル	サンプルド	フル	サンプルド	フル	サンプルド
1	10	1	22	2	2	2
2	19	2	21	1	1	1
3	3	3	3	3	4	5
4	5	4	6	5	9	4
5	4	6	4	4	5	3

表している。

まず, Full (Full) と Sampling (1/10 または 1/100) を比較すると, ITOC と Laura ではあまり違いが見られなかったが, OP ではサンプリングによる劣化が見られた. この差は, フルトラフィックを学習してフルトラフィックを推定する場合に対する, サンプリングによる劣化を表しており, トラフィックトレースの特性に応じてその程度が異なることが分かる. 次に, Full (1/10 または 1/100) と Sampling (1/10 または 1/100) を比較することにより, サンプルドフローを学習する提案手法の有効性を確認した. この理由として, 学習用サンプルドフローの特徴量が, フルトラフィックのフロー特徴量よりも, 推定用サンプルドフローの特徴量とより整合がとれていると考えられる. すなわち, 推定用サンプルドフローの分類に有効な特徴量とサンプルドフローを学習する際に有効であった特徴量とが類似していたためと考えられる. 使用した学習アルゴリズム Bagging は, 4.1 節で述べたとおり, 情報利得の大きな特徴量を用いて決定木を作成する. そこで, 1/10 サンプリングを適用した推定用トラフィックのフロー特徴量の情報利得を算出し, 上位 5 つのフロー特徴量を求め, その特徴量が, 学習用フルトラフィックまたは学習用サンプルドトラフィックの情報利得に基づくランキングにおいて, どう推移するかを調査した. 結果を表 4 に示す. 表より, フルトラフィックの順位よりもサンプルドトラフィックの順位の方が, 推定対象の上位 5 つと似通っていることが分かり, サンプルドフロー特徴量による学習の方が有効な特徴量を選択できていることが分かる. サンプルドトラフィックを学習するという点に関して, 文献 [5] は Sampled NetFlow を用い, 提案手法はフルトラフィックをサンプリングしたものを用いる. NetFlow から算出可能なフロー特徴量とフルトラフィックから算出可能なフロー特徴量とは異なるが, フロー特徴量自体は同様のものを用いるため, 両者の性能は同じになる.

次に, サンプリング回数を比較すると (横軸方向), ITOC と Laura ではほとんど差が現れなかったものの, OP において 10 回または 100 回サンプリングを適用した学習は, 1 回サンプリングを適用した学習よりも性能が高くなっていることが分かる. すなわち, 学習サンプルを増やすことで

推定性能が向上する場合があることを確認した.

次に, 10 回行った結果得られる Overall accuracy の最大値と最小値の差をばらつきと定義し, サンプリング回数によるばらつきを評価した. 図 7 に結果を示す. 横軸と縦軸は, 学習フェーズにおけるサンプリング回数と Overall accuracy のばらつきである. 図 7 より, 10 回または 100 回サンプリングによる学習は 1 回サンプリングによる学習よりも安定しており, 特に 100 回サンプリングのときにばらつきが抑制できていることが分かり, 提案手法の有効性を確認した.

5.1.2 F-measure

トラフィック量が大きいプロトコルに着目して評価を行うために, フルトラフィックにおけるトラフィック量を計算し, 上位 5 番目までのプロトコルに着目して F-measure を評価した. 図 8 にサンプリングレートが 1/100 のときの結果を示す. 横軸と縦軸は, 学習フェーズにおけるサンプリング回数と 10 試行の F-measure の平均値である. 図 8 より, トラフィック量の多い様々なプロトコルに対して 10 回または 100 回サンプリングを適用することにより, 1 回サンプリングを適用した学習よりも性能が向上することを確認した. また, トラフィック量が最も多いプロトコル (ITOC は SSL, Laura と OP は HTTP) のグラフ形状が図 6 の Overall accuracy の結果と類似していることが分かり, Overall accuracy におけるトレース全体の推定性能は, トラフィック量の多いプロトコルを反映したものであると判断できる.

10 回行った結果得られる F-measure の最大値と最小値の差をばらつきと定義し, サンプリング回数によるばらつきを評価した. 図 9 に結果を示す. 横軸と縦軸は, 学習フェーズにおけるサンプリング回数と F-measure の変動であり, Overall accuracy と同様, 10 試行の最大値と最小値の差として定義した. 図 7 より, トラフィック量の多いプロトコルに対して 10 回または 100 回サンプリングによる学習は 1 回サンプリングによる学習よりも安定していることが確認できる. 一方で, OP の ICMP が, サンプリング回数が 10 回の際にばらつきが大きくなっていることが分かる. 図 8 において, 平均値が高いことや 1 回サンプリン

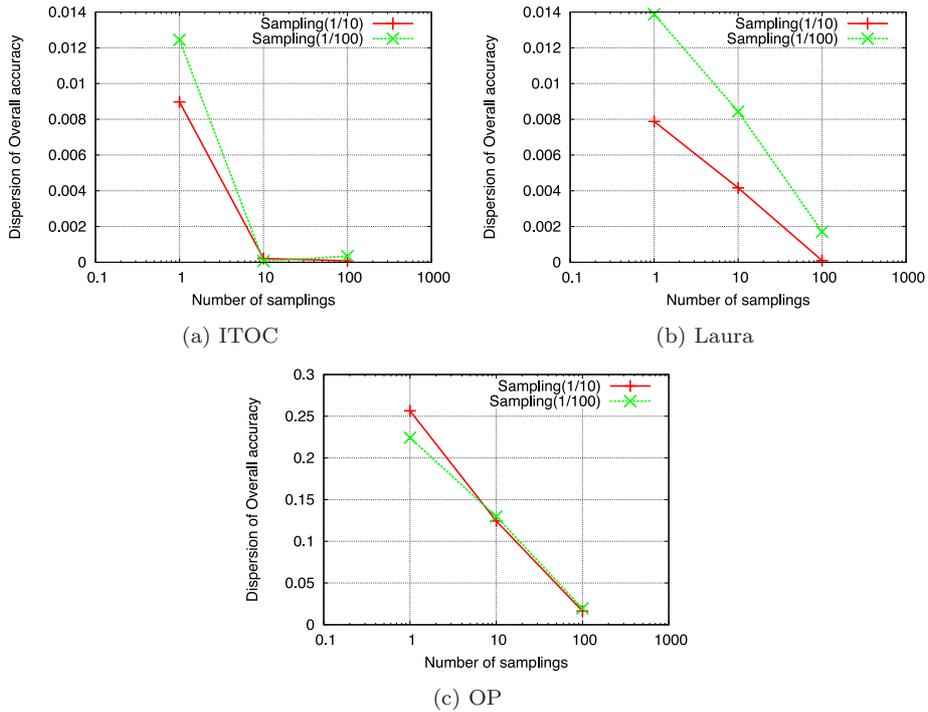


図 7 Overall accuracy に基づくサンプリング回数によるばらつき

Fig. 7 The variability depending on the number of samplings in terms of the dispersion of Overall accuracy.

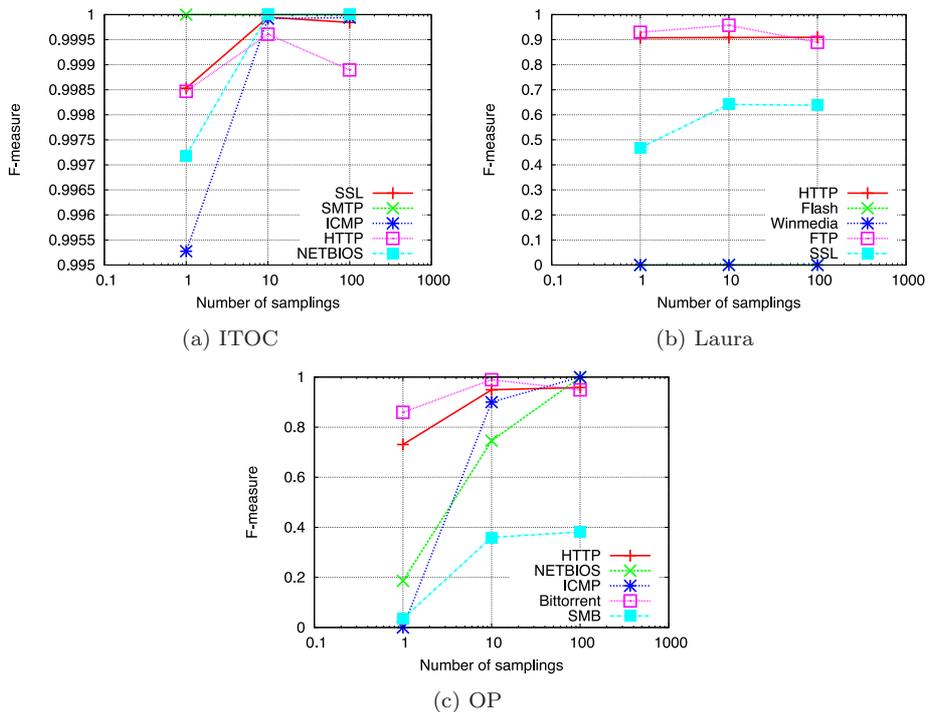


図 8 F-measure に基づく“Sampling”の実験結果 (サンプリングレート: 1/100)

Fig. 8 Experimental results of “Sampling” in terms of F-measure (sampling rate: 1/100).

グの平均値が低いことから、サンプリング回数が少ないために、偶然識別精度が低くなった試行が含まれた結果であると考えられる。サンプリング回数を 100 回に増やすことでばらつきが抑えられていることから、100 回程度のサン

プリングが必要であったことが分かる。以上、5.1.1 項の結果とあわせ、サンプルドフローを学習することで推定性能が向上すること、ならびに複数回サンプリングして得られるフロー特徴量を学習することで推定性能のばらつきが抑

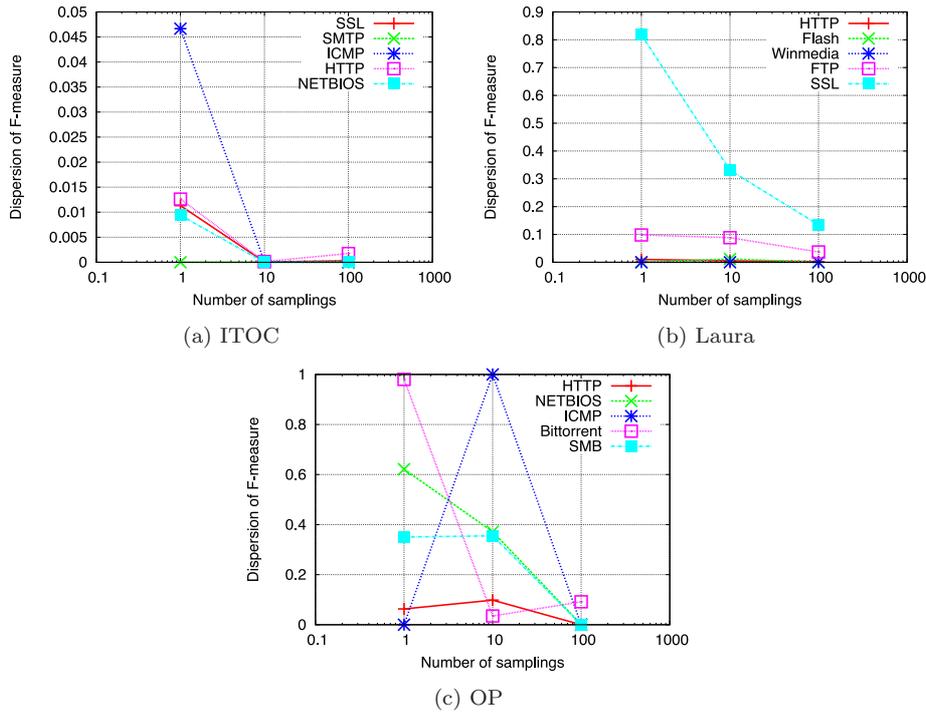


図 9 *F*-measure に基づくサンプリング回数によるばらつき (サンプリングレート : 1/100)

Fig. 9 The variability depending on the number of samplings in terms of the dispersion of *F*-measure (sampling rate: 1/100).

表 5 他トレースの推定結果

Table 5 Result of estimation using the other traffic trace.

学習トレース	推定トレース	Overall accuracy	BI	BR
ITOC	Laura	0.81	0.814	0.1
	OP	0.633	0.874	0.056
Laura	ITOC	0.452	0.976	11.2
	OP	0.631	0.885	0.633
OP	ITOC	0.117	0.999	14.3
	Laura	0.79	0.813	1.46

制されることが確認でき、提案手法の有効性が示された。

5.2 学習と推定の関係

多地点展開を想定し、異なる地点で学習した分類器を使って、異なる地点でのトラフィックのプロトコルを推定したときの調査を行う。3.1 節で述べたとおり、高精度なプロトコル推定を行うには、学習用トラフィックのプロトコル群が推定用トラフィックのプロトコル群を包含する必要がある。そこで、プロトコル群の包含関係とトラフィック量に対する推定精度を調査するため、3 種類のトラフィックトレースのうち 1 種類を使って分類器を生成し、残りの 2 種類のトラフィックトレースを推定した。結果を表 5 に示す。

表 5 において、BI は推定用トラフィックのうち学習用トラフィックのプロトコル群に包含されたトラフィック量の割合を、BR は包含されたプロトコル群における学習用トラフィック量と推定用トラフィック量の比率をそれぞれ表している。それぞれの定義は以下のとおりである。

$$BI = \frac{\sum_{i \in \{LP \cap EP\}} E_i}{\sum_{i \in \{EP\}} E_i} \quad (3)$$

$$BR = \frac{\sum_{i \in \{LP \cap EP\}} E_i}{\sum_{i \in \{LP \cap EP\}} L_i} \quad (4)$$

LP は学習用トラフィックのプロトコル集合を、EP は推定用トラフィックのプロトコル集合をそれぞれ表している。また、 L_i は学習用トラフィックのプロトコル i のトラフィック量を、 E_i は推定用トラフィックのプロトコル i のトラフィック量をそれぞれ表している。

表 5 において、各実験の BI は高いため、学習用トラフィックのプロトコル群は推定用トラフィックのプロトコル群をおおそ包含しているといえる。しかしながら、Laura または OP を学習して ITOC を推定する場合は、精度が悪くなっていることが分かる。そこで BR に着目すると値が大きいため、推定用トラフィック量に対して学習用トラフィック量が少ないことが分かる。

以上の結果より、多地点展開を想定した場合、推定用ト

ラヒックのプロトコル群が学習用トラヒックのプロトコル群に包含され、かつ学習用トラヒック量が推定用トラヒック量に比べて多い必要があると考察される。

5.3 設備コスト

学習フェーズは、学習用トラヒックを専用装置でフルキャプチャし、パケットペイロードを専用 DPI 装置または人的リソースにより精査するため、必要となる設備コストは高い。しかし、高コストな学習フェーズは分類器を生成するための処理であるため、1 地点でも可能である。一方、推定フェーズは、推定対象のトラヒックを汎用 PC を用いてパケットサンプリングしながらキャプチャするため、設備コストは低い。また、推定フェーズは実際の監視に相当し、監視が必要とされる多地点で実施される。すなわち、提案する監視方法は、監視を実施する地点数に比例して低コスト化が実現されることとなる。

6. おわりに

本稿では、大規模 ISP ネットワークにおけるプロトコルごとのトラヒック量を低コストに把握することを目的とし、パケットサンプリングと、教師あり学習によるフロー特徴量ベースのトラヒック分類手法を用いたトラヒック監視を提案した。学習フェーズにはトラヒックのフルキャプチャが必要なものの、多地点で実行する推定フェーズは、安価な汎用 PC により低コストに実現できる。また、フルトラヒックに対してパケットサンプリングを繰り返し、得られるフローを統合することで学習サンプル数を増やすという、新しいプロトコル推定手法を提案した。公開トラヒックトレースを用いて提案手法を評価し、有効性を確認した。

謝辞 日ごろご指導いただく KDDI 研究所中島所長ならびに長谷川執行役員に感謝する。

参考文献

[1] IANA: Port Numbers, IANA (online), available from <http://www.iana.org/assignments/port-numbers> (accessed 2011-05-12).

[2] Cisco Systems: Cisco SCE Series, Cisco Systems (online), available from <http://www.cisco.com/en/US/products/ps6151/index.html> (accessed 2011-05-12).

[3] ipoque: OpenDPI, ipoque (online), available from <http://www.opendpi.org/> (accessed 2011-05-12).

[4] Nguyen, T.T. and Armitage, G.: A Survey of Techniques for Internet Traffic Classification using Machine Learning, *IEEE Communications Surveys and Tutorials*, Vol.10, No.1-4, pp.56-76 (2008).

[5] Carela-Espanol, V., Barlet-Ros, P., Cabellos-Aparicio, A. and Sole-Pareta, J.: Analysis of the impact of sampling on NetFlow traffic classification, *Computer Networks*, Vol.55, No.5, pp.1083-1099 (2011).

[6] sFlow.org: sFlow Developer Tools, sFlow.org (online), available from <http://www.sflow.org/developers/tools.php> (accessed 2011-05-12).

[7] Moore, A., Zuev, D. and Crogan, M.: Discriminators for

use in flow-based classification, Technical Report RR-05-13, Department of Computer Science, Queen Mary, University of London, Mile End Road, London E1 4NS, UK (2005).

[8] Williams, N., Zander, S. and Armitage, G.J.: A Preliminary Performance Comparison of Five Machine Learning Algorithms for Practical IP Traffic Flow Classification, *Computer Communication Review*, Vol.36, No.5, pp.5-16 (2006).

[9] McGregor, A., Hall, M., Lorier, P. and Brunskill, J.: Flow Clustering Using Machine Learning Techniques, *Proc. 5th International Workshop on Passive and Active Network Measurement*, pp.205-214 (2004).

[10] Roughan, M., Sen, S., Spatscheck, O. and Duffield, N.G.: Class-of-Service Mapping for QoS: A Statistical Signature-based Approach to IP Traffic Classification, *Proc. 4th ACM SIGCOMM Conference on Internet Measurement*, pp.135-148 (2004).

[11] Conti, L.G., Cook, C.T., Moss, L.M., OConnor, M.T., Dean, M.E. and Backmon, M.C.: ITOC Research: CDX Datasets, ITOC (online), available from <http://www.itoc.usma.edu/research/dataset/> (accessed 2011-05-12).

[12] Chappell, L.: Laura's Lab Kit v.8, Protocol Analysis Institute Inc. (online), available from http://demeter.uni-regensburg.de/Lauras_Lab_Kit_v8/AutoPlay/trace_files_llk8/ (accessed 2011-05-12).

[13] OpenPacket.org: OpenPacket.org Capture Repository, OpenPacket.org (online), available from <https://www.openpacket.org/capture/list> (accessed 2011-05-12).

[14] The University of Waikato: Weka 3.6.3, The University of Waikato (online), available from <http://www.cs.waikato.ac.nz/ml/weka/> (accessed 2011-05-12).

[15] Freund, Y. and Schapire, R.E.: A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting, *Computer and System Sciences*, Vol.55, No.1, pp.119-139 (1997).

[16] Ai, W.I. and Langley, P.: Induction of One-Level Decision Trees, *Proc. 9th International Conference on Machine Learning*, pp.233-240 (1992).

[17] Breiman, L.: Bagging Predictors, *Machine Learning*, Vol.24, No.2, pp.123-140 (1996).

[18] Quinlan, J.R.: *C4.5: Programs for Machine Learning*, Morgan Kaufmann (1993).

[19] John, G.H. and Langley, P.: Estimating Continuous Distributions in Bayesian Classifiers, *Proc. 11th Conference on Uncertainty in Artificial Intelligence*, pp.338-345 (1995).



後藤 崇行 (正会員)

平成 19 年早稲田大学大学院国際情報通信研究科国際情報通信学専攻修士課程修了。同年 KDDI (株) 入社。ネットワーク計測技術の研究に従事。現在、(株) KDDI 研究所 IP 品質制御システムグループ研究員。



佐々木 力 (正会員)

平成 16 年東京工業大学大学院理工学研究科集積システム専攻修士課程修了。同年 KDDI (株) 入社。経路制御, マルチキャストの研究に従事。現在, (株) KDDI 研究所 IP 品質制御システムグループ研究主査。



立花 篤男 (正会員)

平成 14 年大阪大学大学院修士課程修了。同年 KDDI (株) 入社。以来, 研究所にて, IP ネットワーク計測・管理, 次世代インターネットの研究に従事。現在, (株) KDDI 研究所 IP 品質制御システムグループ研究主査。



阿野 茂浩 (正会員)

昭和 62 年早稲田大学理工学部電子通信学科卒業。平成元年同大学大学院修士課程修了。同年国際電信電話 (株) 入社。以来, 研究所にて, ATM 交換方式, IP ネットワーク管理・制御, 次世代インターネットの研究に従事。現在, (株) KDDI 研究所 NW 設計グループリーダー。平成 7 年度情報処理学会大会奨励賞, 平成 22 年度電子情報通信学会通信ソサイエティ優秀論文賞等各受賞。