Do we need a perfect ground-truth for benchmarking Internet traffic classifiers?

M. Rosário Oliveira*, João Neves*, Rui Valadas†, Paulo Salvador‡
*CEMAT and Departmento de Matemática, Instituto Superior Técnico, Universidade de Lisboa, Portugal
Email: (rosario.oliveira,joao.d.neves)@tecnico.ulisboa.pt
†DEEC and Instituto de Telecomunicações, Instituto Superior Técnico, Universidade de Lisboa, Portugal
Email: rui.valadas@tecnico.ulisboa.pt
‡DETI and Instituto de Telecomunicações, Universidade de Aveiro, Portugal, Email: salvador@ua.pt

***Abstract*—The classification of Internet traffic using supervised or semi-supervised statistical learning techniques, both for anomaly detection and identification of Internet applications, has been impaired by difficulties in obtaining a reliable ground-truth, required both to train the classifier and to evaluate its performance. A perfect ground-truth is increasingly difficult, or sometimes impossible, to obtain due to the growing percentage of cyphered traffic, the sophistication of network attacks, and the constant updates of Internet applications. In this paper, we study the impact of the ground-truth on training the classifier and estimating its performance measures. We show both theoretically and through simulation that ground-truth imperfections can severely bias the performance estimates. We then propose a latent class model that overcomes this problem by combining estimates of several classifiers over the same dataset. The model is evaluated using a high-quality dataset that includes the most representative Internet applications and network attacks. The results show that our latent class model produces very good performance estimates under mild levels of ground-truth imperfection, and can thus be used to correctly benchmark Internet traffic classifiers when only an imperfect ground-truth is available.**

***Index Terms*—Traffic Classification, Latent Class Models, Anomaly Detection, Identification of Internet Applications.**

I. Introduction

The classification of Internet traffic is a task of growing importance, which finds applications in several areas, such as anomaly detection and identification of Internet applications. Many classifiers have been proposed over the years, but the reliability of the *ground-truth* has been an ever-present issue. The ground-truth is a subset of traffic objects labeled with the class they belong to, needed both to train the classifier and to evaluate its performance. A *perfect* ground-truth, i.e. one where all traffic objects are correctly labeled, is increasingly difficult, or sometimes impossible, to obtain due to the growing percentage of cyphered traffic, the constant updates of Internet applications, and the sophistication of Internet attacks, which try to travel unnoticed mixed with the licit traffic. As it will be shown in this paper, estimating the performance of Internet traffic classifiers using an imperfect ground-truth can result in gross errors, undermining any attempt to benchmark classifiers, which is required to select the best one for specific classification problems.

The ground-truth problem is especially acute in anomaly detection, since it involves the identification of Internet attacks (e.g. Botnet attacks, polymorphic worm/virus spreading, DDoS, and flash crowds). In their critical reviews of the anomaly detection research area, Gates and Taylor first [1], and Sommer and Paxson latter [2], pointed-out that attack-free data is hard to find, and elected the difficulty in obtaining a perfect ground-truth as one of the most challenging problems in the area.

Several approaches have been used to build the ground-truth for anomaly detection, including manual inspection of real traces, mixing emulated attacks with attack-free real traffic, and generating fully emulated traces.

Labeling traffic objects through manual inspection is a time-consuming and complex process [3]–[5], combining hints from the analysis of targeted ports or IP addresses, visual inspection of time series, top-n queries on the data, and expert knowledge about malware from forums and threat expert reports. However, this method cannot assure that network attacks are fully identified, given their unpredictable nature.

A better control over the ground-truth is obtained when the network traffic is emulated. Kim and Reddy [6] compose the ground-truth using the NLANR traces, assumed attack-free, together with a mixture of real and simulated attacks. The real attacks are scan attacks which, according to the authors, are easily detected using flow-counting analysis. Thatte et al. [7] use background traffic from their university network, again assumed attack-free, mixed with attacks emulated using Iperf. Pascoal et al. [8] produced the background traffic in a highly protected laboratory, with a selected set of users accessing the Internet but constrained on the type of applications they could use and on the sites they could visit; the attacks were emulated using NMAP and other tools. Kind et al. [9] uses the MIT Lincoln Labs dataset [10], where both the background traffic and the network attacks are emulated, trying to replicate the behavior of an Air Force base network. The approaches using emulated traffic allow higher percentages of correctly labeled objects, but can be easily criticized for not mimicking real network conditions, since traffic is emulated, rather than just observed "from the real world". Moreover, it is always difficult to assure that real background traffic is free of attacks.

The difficulties in obtaining a perfect ground-truth for anomaly detection led some authors to advocate the use of computer simulation [11]. Unlike emulation, where the traffic

978-1-4799-8381-0/15/$31.00 ©2015 IEEE

is generated using real applications, simulation relies entirely on models of network mechanisms and traffic. However, the complexity of modeling the user behavior and the networking mechanisms that most impact the traffic, and in reverse-engineering several types of network attacks, render it hard to imitate real traffic through simulation [2], [12].

Obtaining a perfect ground-truth is not less of an issue for the identification of Internet applications. In this case, the ground-truth is usually obtained using deep packet inspection [13]–[23]. However, this technique is useless in the case of proprietary and encrypted traffic. Moreover, real traces may include Internet attacks which, as discussed above, are hard to identify. As in the case of anomaly detection, the ground-truth issue is considered one of the most challenging ones [24].

The ground-truth issue is closely related with the problem of evaluating diagnostic tests, which has been for long in the agenda of life sciences [25]–[27]. Diagnostic tests determine whether or not a patient (human or animal) has a given disease. Incorrect information on the accuracy of a diagnostic test can lead doctors into wrong treatments, possibly putting the patient's life at risk. This is a very sensitive issue! The correct evaluation of diagnostic tests requires a perfect reference, i.e. a diagnostic test that identifies the status of a patient (diseased or non-diseased) without mistakes, called *gold standard*. Just as in the case of Internet traffic classification, the gold standard is very difficult to obtain. None is available for many diseases and, even when there is one, its application can be very costly, take a long time, be technically complex, or involve several risks, discouraging its use in a massive or regular way. An attractive alternative is to use the best diagnostic test available as a reference. However, it is well-known in the life sciences community that this approach can unacceptably bias the performance estimates. Latent class models are the established choice.

In this paper we propose a latent class model to estimate the performance of Internet traffic classifiers when only an imperfect ground-truth is available. The model alleviates the requirements on the reliability of the ground-truth when benchmarking Internet traffic classifiers. Although related, this problem is broader than the one faced by diagnostic tests. In both cases, the goal is to find the best classifier (diagnostic test) in the absence of a perfect ground-truth (gold standard). However, gold standard imperfections only impact the accuracy of the performance measures, and not the outcome of the test. Contrarily, ground-truth imperfections also impact the construction (training) of the classifier, and are thus an additional and important source of estimation error. Our latent class model embraces the classifier training process, through the features extracted from data, in the estimation of the performance measures.

The paper is structured as follows. In section II we give some background on latent class models. In section III we study the impact of an imperfect ground-truth on the accuracy of the performance measures, when the behavior of the classifier with respect to the ground-truth is predefined. In section IV we extend this study to account for the classifier training process based on features extracted from data, and propose a latent class model that improves the estimation of the performance measures. Finally, in section V we present the main conclusions of our work.

II. The latent class model

The Latent Class Model (LCM) was proposed by Lazarsfeld and Henry [28], and has been widely used in social and behavioral sciences [25], [26], [28]. It assumes that, while a certain categorical variable is not observable (and because of that called *latent*), its effect can be indirectly expressed by other observable categorical random variables, named *manifest* variables. For example, we cannot observe directly the ambition of an individual, but there are questions that can be made in order to obtain indications of such characteristic.

Consider p manifest variables, $X_1, X_2, \ldots, X_p$, each with c_i categories, and assume a single latent variable Y, with k categories, called latent classes. The LCM parameters are defined as:

- $\pi_{sj}(i) = P(X_s = i|Y = j)$, the probability that X_s assigns an object to category i, when it belongs to the j-th latent class ($i = 1, 2, \ldots, c_i$, $j = 1, 2, \ldots, k$, $s = 1, 2, \ldots, p$);
- $p_j = P(Y = j)$, the probability of the j-th latent class ($j = 1, 2, \ldots, k$).

LCM assumes that the manifest variables are independent of each other for fixed values of the latent variable, i.e.,

$$\begin{aligned} P(\boldsymbol{X} = \boldsymbol{x}|Y = j) &= \prod_{i=1}^{p} P(X_i = x_i|Y = j) \\ &= \prod_{i=1}^{p} \pi_{ij}(x_i), \end{aligned} \tag{1}$$

where $\boldsymbol{X} = (X_1, X_2, \ldots, X_p)^t$, $\boldsymbol{x} = (x_1, x_2, \ldots, x_p)^t$, $x_i \in \{1, 2, \ldots, c_i\}$, $i = 1, \ldots, p$. This is called the Hypothesis of Conditional Independence (HCI).

Under HCI, the probability that an object with response vector $\boldsymbol{x}$ belongs to the j-th class is given by

$$\begin{aligned} d(j|\boldsymbol{x}) &= P(Y = j|\boldsymbol{X} = \boldsymbol{x}) \\ &= \frac{P(Y = j)\, P(\boldsymbol{X} = \boldsymbol{x}|Y = j)}{P(\boldsymbol{X} = \boldsymbol{x})} \\ &= \frac{p_j \prod_{i=1}^{p} \pi_{ij}(x_i)}{\sum_{m=1}^{k} p_m \prod_{i=1}^{p} \pi_{im}(x_i)}. \end{aligned}$$

The EM estimates of the model parameters, obtained based on n response vectors, $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$, are given by [27]:

$$\hat{p}_j = \frac{1}{n} \sum_{h=1}^{n} \hat{d}(j|\boldsymbol{x}_h), \tag{2}$$

$$\hat{\pi}_{ij}(x_i) = \frac{\sum_{h=1}^{n} \hat{d}(j|\boldsymbol{x}_h)\, x_{hi}(s)}{n\, \hat{p}_j}, \tag{3}$$

where $i = 1, \ldots, p$; $j = 1, \ldots, k$.

The LCM is naturally amenable to the performance evaluation of classifiers since the usual performance measures, such

as recall and precision, can be easily written as a function of the LCM parameters. The recall of class i is the probability that the classifier (whose predicted class is represented by X_s) assigns correctly an observation to class i, and is given by

$$Re(i) = P(X_s = i|Y = i) = \pi_{si}(i).$$

The precision of class i is the probability that an observation assigned to class i truly belongs to that class, and is given by

$$\begin{aligned} Pr(i) &= P(Y = i|X_s = i) \\ &= \frac{P(X_s = i|Y = i)p_i}{\sum_{m=1}^{k} P(X_s = i|Y = m)p_m} \\ &= \frac{\pi_{si}(i)p_i}{\sum_{m=1}^{k} \pi_{sm}(i)p_m}. \end{aligned} \quad (4)$$

The HCI can be considered unrealistic in some practical problems, e.g. in the case of two diagnostic tests that are based on the same biological principle. To overcome this limitation alternatives that incorporate local dependence have been proposed [26]. In this paper, the HCI has been studied using the methodologies described in [29].

Another potential problem with LCMs is the lack of correspondence between latent and user classes. This results from the categorical nature of latent variables, and is a common difficulty in unsupervised learning algorithms. In our case, this problem is easily solved by associating the latent classes to the user classes (the Internet traffic applications) that maximize the simple matching coefficient, based on the labels of the traffic objects available to train the classifiers.

III. Impact of ground-truth on the estimation process

In this section, we study the impact of an imperfect ground-truth on the performance measures, assuming that the behavior of the classifier with respect to the perfect ground truth, i.e., $P(X = i|Y = y), i, y = 1, \ldots, k$, is known a priori. Let Z be a random variable representing the class obtained with an imperfect ground-truth. Under these circumstances, the *imperfect* recall and precision for class i, $i = 1, \ldots, k$, are given by:

$$\begin{aligned} Re^I(i) &= P(X = i|Z = i), \\ Pr^I(i) &= P(Z = i|X = i). \end{aligned}$$

Assuming that Z and X are independent random variables given $\{Y = y\}$, $y = 1, \ldots, k$, these expressions can be written as

$$Re^I(i) = \frac{\sum_{y=1}^{k} Re_X(i, y)Re_Z(i, y)p_y}{\sum_{y=1}^{k} Re_Z(i, y)p_y}, \quad (5)$$

$$Pr^I(i) = \frac{\sum_{y=1}^{k} Re_X(i, y)Re_Z(i, y)p_y}{\sum_{y=1}^{k} Re_X(i, y)p_y}, \quad (6)$$

where $Re_A(a, b) = P(A = a|Y = b)$ and $Re_A(a) = P(A = a|Y = a)$. These expressions highlight the factors contributing to the imperfect performance measures: $Re_X(i, y) = P(X = i|Y = y)$ characterizes the classifier with respect to the perfect ground-truth, $Re_Z(i, y) = P(Z = i|Y = y)$ characterizes the imperfect ground-truth with respect to the perfect one, and p_y characterizes the perfect ground-truth.

TABLE I: Classifier characteristics; $Re_1(x, y)$ on the left, $Re_2(x, y)$ at the center, and $Re_3(x, y)$ on the right.

		y = 1			y = 2			y = 3		
x	1	0.95	**0.85**	0.90	0.00	**0.05**	0.00	0.00	**0.05**	0.00
	2	0.05	**0.10**	0.10	0.90	**0.80**	0.85	0.25	**0.35**	0.10
	3	0.00	**0.05**	0.00	0.10	**0.15**	0.15	0.75	**0.65**	0.90

TABLE II: Imperfect ground-truth characteristics; $P(Z_1 = z|Y = y)$ on the left and $P(Z_2 = z|Y = y)$ on the right.

		y = 1		y = 2		y = 3	
z	1	0.95	**0.85**	0.00	**0.00**	0.00	**0.00**
	2	0.05	**0.15**	0.90	**0.80**	0.10	**0.20**
	3	0.00	**0.00**	0.10	**0.20**	0.90	**0.80**

To evaluate the impact of an imperfect ground-truth on the recall and precision, we carried out a simulation study. We considered three classes with probabilities $p_1 = p_2 = 0.4$ and $p_3 = 0.2$, three classifiers, X_1, X_2, and X_3, and two types of imperfect ground-truth, Z_1 and Z_2. The characteristics of each classifier are expressed in terms of $Re_s(x, y) = P(X_s = x|Y = y)$, $s = 1, 2, 3$ and were defined according to our own practice with algorithms C4.5, kNN, and Naive Bayes (NB); they are shown in Table I. The characteristics of the imperfect ground-truth are expressed by $P(Z_i = z|Y = y)$, $i = 1, 2$, and are summarized in Table II.

The simulation procedure consisted in generating first the true class Y of each object, with the probabilities defined above. Then, using the true class, we generated the imperfect class $Z_i|Y = y$ and the output $X_s|Y = y$ of each classifier using the conditional probabilities defined in tables I and II. This allows estimating the recall and precision in four cases: with perfect ground-truth (P), with imperfect ground-truth of types 1 and 2 (I_1 and I_2), and with the latent class model (LCM) where the three classifiers are combined together. In this simulation, samples are of size $n = 1000$. Each simulation was replicated 10000 times to obtain average recall and precision estimates. The results relative to classifier X_1 are shown in Figure 1. As expected, the recall and precision estimates obtained with the perfect ground-truth are very close to the theoretical values, which for recall are 0.95 (class 1), 0.90 (class 2), and 0.75 (class 3) (see Table I), and for precision are 1 (class 1), 0.837 (class 2), and 0.790 (class 3) (using equation (5)). Results clearly highlight the estimation errors associated with the use of an imperfect ground-truth. For example, the recall estimate of class 3 is 0.632 (type 1 imperfection) and 0.533 (type 2 imperfection); the theoretical recall is 0.75. Likewise, the precision of class 2 is 0.768 (type 1 imperfection) and 0.716 (type 2 imperfection); the theoretical

precision is 0.837. However, the LCM estimates are very close to the theoretical ones, showing that the model is able to cope well with ground-truth imperfections.

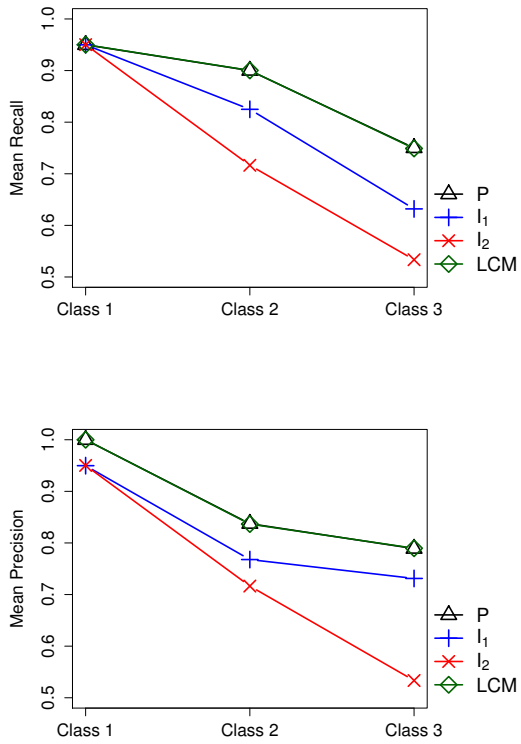

Fig. 1: Estimated recalls and precisions relative to X_1.

IV. Impact of Ground-Truth on the Classification and Estimation Processes

Up to now we ignored that an imperfect ground-truth also impacts the construction (training) of the classifier, which is an additional factor influencing the estimation of the performance measures. We address this problem in this section, both theoretically and through a simulation study based on real data.

Several cases can be considered, depending on whether or not perfect or imperfect ground-truths are used in the training and estimation tasks. Assuming that A-B denotes training with A and estimating the performance measures with B, we can consider the following cases: Perfect-Perfect (PP), Perfect-Imperfect (PI), Imperfect-Perfect (IP), and Imperfect-Imperfect (II). PP is the ideal scenario, that we will take as a reference. II is the scenario with more practical significance, since training and estimating the performance measures are both done with an imperfect ground-truth. The PI and IP scenarios are unrealistic, since if a perfect ground-truth is available then it could be used both for training and estimation. However, these scenarios allow assessing the isolated impact of the imperfect ground-truth on each task. As an alternative, our proposal is to use the LCM to estimate the performance measures. Thus, two more cases are considered: Perfect-LCM (PLCM) and Imperfect-LCM (ILCM).

A. Theoretical recall and precision

When only an imperfect ground-truth is available, the estimation of the performance measures is influenced by the classifier construction process. Classifiers are trained using the class labels assigned to each object and their characteristic features. If the class labels are incorrectly set (imperfect ground-truth) the classification rule will be affected.

As before, let Y represent the perfect ground-truth and Z the imperfect one. In this theoretical study we consider only two classes ($k = 2$) and assume that the features, summarized by random vector $\mathbf{W}$, follow a multivariate normal distribution with equal covariance matrices in each class. With these assumptions, the classification rule that minimizes the total probability of misclassification (TPM) is a linear function of $\mathbf{W}$, whose coefficients only depend on the expected mean vectors of the features in each class, $\boldsymbol{\mu}_j = E(\mathbf{W}|Y = j)$, $j = 1, 2$ and on the common covariance matrix, $\boldsymbol{\Sigma}$ [30]. Letting X^{P} represent the classification variable, the optimal classification rule is given by:

$$\begin{cases} \text{Assign } \mathbf{w} \text{ to } \{Y = 1\} \Leftrightarrow X^{\mathrm{P}} = 1, & \text{if } \boldsymbol{\alpha}^t \mathbf{w} \geq m \\ \text{Assign } \mathbf{w} \text{ to } \{Y = 2\} \Leftrightarrow X^{\mathrm{P}} = 2, & \text{otherwise} \end{cases} \quad (7)$$

where $\boldsymbol{\alpha} = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$, $m = \ln \frac{p_2}{p_1} + \frac{1}{2}\boldsymbol{\alpha}^t(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2)$, and p_i is the prior probability of each class.

For the optimal classifier described by (7), the recall with respect to the perfect ground-truth, $Re_{X^{\mathrm{P}}}(i) = Re^{\mathrm{PP}}(i)$, is given by [30]

$$Re^{\mathrm{PP}}(i) = \Phi\left(\frac{\Lambda}{2} + \frac{1}{\Lambda}\ln\frac{p_i}{1-p_i}\right), \quad i \in \{1, 2\} \quad (8)$$

where $\Lambda^2 = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^t \boldsymbol{\Sigma}^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$ is the square of the Mahalanobis distance between the two mean vectors and Φ represents the cumulative distribution function of the standard univariate normal distribution. It is through this parameter that the recall incorporates the characteristics of the features.

For the same classifier, the precision with respect to the perfect ground-truth, $Pr^{\mathrm{PP}}(i) = P(Y = i|X^{\mathrm{P}} = i)$, can be written as a function of $Re^{\mathrm{PP}}(i)$ and p_i in the following way:

$$Pr^{\mathrm{PP}}(i) = \frac{Re^{\mathrm{PP}}(i)p_i}{Re^{\mathrm{PP}}(i)p_i + (1 - Re^{\mathrm{PP}}(j))\,p_j}, \; i \neq j. \quad (9)$$

In the PI case, the recall $Re^{\mathrm{PI}}(i)$ and precision $Pr^{\mathrm{PI}}(i)$, are defined as in (5) and (6), where $Re_{X^{\mathrm{P}}}(i, i) = Re^{\mathrm{PP}}(i)$ and $Re_{X^{\mathrm{P}}}(i, y) = 1 - Re^{\mathrm{PP}}(y)$, for $i \neq y$, $i, y \in \{1, 2\}$.

In the case of an imperfect ground-truth, $\mathbf{W}|Z = j$ will not be multivariate normal if $\mathbf{W}|Y = y$ is. In fact, $\mathbf{W}|Z = j$ is a mixture of k multivariate normal distributions, $\mathbf{W}|Y = y$, with weights $Pr_Z(y, j)$, $y = 1, 2$, where $Pr_A(y, a) = P(Y = y|A = a)$ and $Pr_A(a) = P(Y = a|A = a)$. Moreover, $\boldsymbol{\xi}_j = E(\mathbf{W}|Z = j) = \sum_{y=1}^{k} \boldsymbol{\mu}_y Pr_Z(y, j)$ and its covariance matrix depends on the class j. However, in practice it is usually assumed that $\mathbf{W}|Z = j$ is multivariate normal, which is approximately true if $Pr_Z(y, j) \simeq 0$, for all $y \neq j$.

In this case, letting X^{I} denote the classification variable, the optimal classification rule is given by:

$$\begin{cases} \text{Assign } \mathbf{w} \text{ to } \{Y=1\} \Leftrightarrow X^{\mathrm{I}}=1, & \text{if } \boldsymbol{\beta}^t\mathbf{w} \geq \gamma, \\ \text{Assign } \mathbf{w} \text{ to } \{Y=2\} \Leftrightarrow X^{\mathrm{I}}=2, & \text{otherwise,} \end{cases} \quad (10)$$

where $\boldsymbol{\beta} = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\xi}_1 - \boldsymbol{\xi}_2)$ and $\gamma = \ln\frac{P(Z=2)}{P(Z=1)} + \frac{1}{2}\boldsymbol{\beta}^t(\boldsymbol{\xi}_1 + \boldsymbol{\xi}_2)$. This classification rule is optimal, in the sense that it minimizes the TPM, under the assumption that $\mathbf{W}|Z = j$ is multivariate, with mean vector $\boldsymbol{\xi}_j$ and covariance matrix $\boldsymbol{\Sigma}$.

The recall and precision in the IP case can be obtained by replacing in (8) Λ^2 by Γ^2 and p_i by $g_i = P(Z = i) = \sum_{y=1}^{2} Re_Z(i,y)p_y$, and by replacing in (9) $Re^{\mathrm{PP}}(y)$ by $Re^{\mathrm{IP}}(y)$, $y \in \{1,2\}$, leading to:

$$Re^{\mathrm{IP}}(i) = \Phi\left(\frac{\Gamma}{2} + \frac{1}{\Gamma}\ln\frac{g_i}{1-g_i}\right), \quad i \in \{1,2\}. \quad (11)$$

Similarly, the recall and precision in the II case are given by:

$$\begin{aligned} Re^{\mathrm{II}}(i) &= Re^{\mathrm{IP}}(i)Re_Z(i)p_i/g_i \\ &\quad + (1 - Re^{\mathrm{IP}}(j))Re_Z(i,j)p_j/g_i, \quad (12) \end{aligned}$$

$$Pr^{\mathrm{II}}(i) = \frac{Re^{\mathrm{II}}(i)\sum_{y=1}^{2} Re_Z(i,y)p_y}{Re^{\mathrm{II}}(i)p_i + (1 - Re^{\mathrm{II}}(j))p_j}, \quad (13)$$

for $i \neq j$, and $i, j \in \{1,2\}$.

B. Theoretical recall deviation

The goal of this section is to study the deviations in the performance measures introduced by the degree of ground-truth's imperfection and the characteristics of the data features. Due to lack of space, we only show results related with the recall. Similar conclusions can be drawn from the precision results. We denote the recall deviation by $\Delta_{Re}^{\mathrm{AB}}(i) = Re^{\mathrm{AB}}(i) - Re^{\mathrm{PP}}(i)$, where AB means training with A and estimating the performance measures with B, and both A and B can represent a perfect ground-truth (P) or an imperfect one (I). The characteristics of the data features are expressed in terms of the separation between classes given by the Mahalanobis distances Λ or Γ, and the degree of ground-truth imperfection by the recall of the imperfect ground-truth $Re_Z(i)$. To obtain suitable numerical results, we consider that (i) the two classes are equally probable, i.e. $p_1 = p_2 = 1/2$, and that (ii) the degree of ground-truth imperfection is the same in the two classes, i.e. $Re_Z(1) = Re_Z(2)$. We will denote it by η. Under the above conditions, $g_1 = g_2 = 1/2$, $Re^{\mathrm{PP}}(1) = Re^{\mathrm{PP}}(2) = \Phi(\Lambda/2)$, and TPM is $(1 - \Phi(\Lambda/2)$. Moreover, $\Gamma = |2\eta - 1|\Lambda$. The results, which we discuss below, are shown in Figure 2.

In the studies related with η we are only interested in the range $\eta \in [0.5, 1]$. In fact, when there are only two classes and $Re_Z(1) = Re_Z(2)$, swapping the Z categories is equivalent to swapping η by $1 - \eta$. For example, if $\eta = 0$ and we swap Z, then a perfect ground-truth is obtained, which corresponds to $\eta = 1$. The most unfavorable situation occurs when $\eta = 1/2$, which corresponds to an imperfect ground-truth that assigns an object to a class by tossing a fair coin.

1) PI case: We start by studying the PI case, i.e. when the classifier is trained with the perfect ground-truth but the estimation of the performance measures is made with an imperfect one. The recall deviation is given by

$$\Delta_{Re}^{\mathrm{PI}}(i) = (1 - Re^{\mathrm{PP}}(1) - Re^{\mathrm{PP}}(2))\,(1 - Pr_Z(i))\,. \quad (14)$$

Interestingly, $\Delta_{Re}^{\mathrm{PI}}(i)$ can be written as the product of two factors, the first one depending only on the classifier and the second one on the ground-truth. The deviation will be negative, i.e. $Re^{\mathrm{PI}}(i) < Re^{\mathrm{PP}}(i)$ whenever $Re^{\mathrm{PP}}(1) + Re^{\mathrm{PP}}(2) > 1$. It can be easily seen that this condition corresponds to $\boldsymbol{\mu}_1 \neq \boldsymbol{\mu}_2$. Likewise, the deviation will be null, i.e. $Re^{\mathrm{PI}}(i) = Re^{\mathrm{PP}}(i)$, when $\boldsymbol{\mu}_1 = \boldsymbol{\mu}_2$. This result shows that, except for the trivial case where the two classes coincide, there will always be a penalty associated to the use of an imperfect ground-truth, even if the classifier is constructed using a perfect one.

Under the simplifying assumptions, equation (14) reduces to

$$\Delta_{Re}^{\mathrm{PI}}(i) = \left(1 - 2\Phi\left(\frac{\Lambda}{2}\right)\right)(1 - \eta)\,.$$

In Figures 2.(a) and 2.(b) we plot the recall deviation as a function of the class separation Λ and the imperfection degree η, respectively. The recall deviation decreases with Λ and increases with η. In particular, for a fixed η, if $\Lambda \to 0$ the recall deviation goes to zero, and if $\Lambda \to +\infty$ then $\Delta_{Re}^{\mathrm{PI}}(i) \to -(1 - \eta)$. Thus, if the two classes are very well separated the recall deviation is negative and its magnitude only depends on how good the ground-truth is, with no impact from the classifier construction process. For a fixed Λ, the recall deviation goes to zero when the $\eta \to 1$ and goes to $1/2 - \Phi(\Lambda/2)$ when $\eta \to 1/2$, which decreases in magnitude when the $\Lambda \to +\infty$.

2) IP case: The recall deviation in the IP case is

$$\begin{aligned} \Delta_{Re}^{\mathrm{IP}}(i) &= \Phi\left(\frac{\Gamma}{2} + \frac{1}{\Gamma}\ln\frac{g_i}{1-g_i}\right) \\ &\quad - \Phi\left(\frac{\Lambda}{2} + \frac{1}{\Lambda}\ln\frac{p_i}{1-p_i}\right). \quad (15) \end{aligned}$$

and, under the simplifying assumptions, reduces to

$$\Delta_{Re}^{\mathrm{IP}}(i) = \Phi\left(|2\eta - 1|\frac{\Lambda}{2}\right) - \Phi\left(\frac{\Lambda}{2}\right).$$

The behavior of $\Delta_{Re}^{\mathrm{IP}}(i)$ is similar to $\Delta_{Re}^{\mathrm{PI}}(i)$, except when η is fixed and $\Lambda \to +\infty$. As shown in Figure 2.c, except for $\eta = 1/2$, there is an inflection point and $\Delta_{Re}^{\mathrm{IP}}(i) \to 0$ as $\Lambda \to +\infty$. In fact, in the limit when the classes are extremely well separated constructing the classifier with an imperfect ground-truth does not have an impact on the recall, since the recall estimation is done using a perfect ground-truth.

3) II case: The case with most practical interest is when both the construction of the classifier and the estimation of the performance metrics is done with an imperfect ground-truth. In this case, the recall deviation is given by

$$\begin{aligned} \Delta_{Re}^{\mathrm{II}}(i) &= \Delta_{Re}^{\mathrm{PI}}(i) + \\ &\quad + \Delta_{Re}^{\mathrm{IP}}(i)Pr_Z(i) - \Delta_{Re}^{\mathrm{IP}}(j)(1 - Pr_Z(i)) \quad (16) \end{aligned}$$

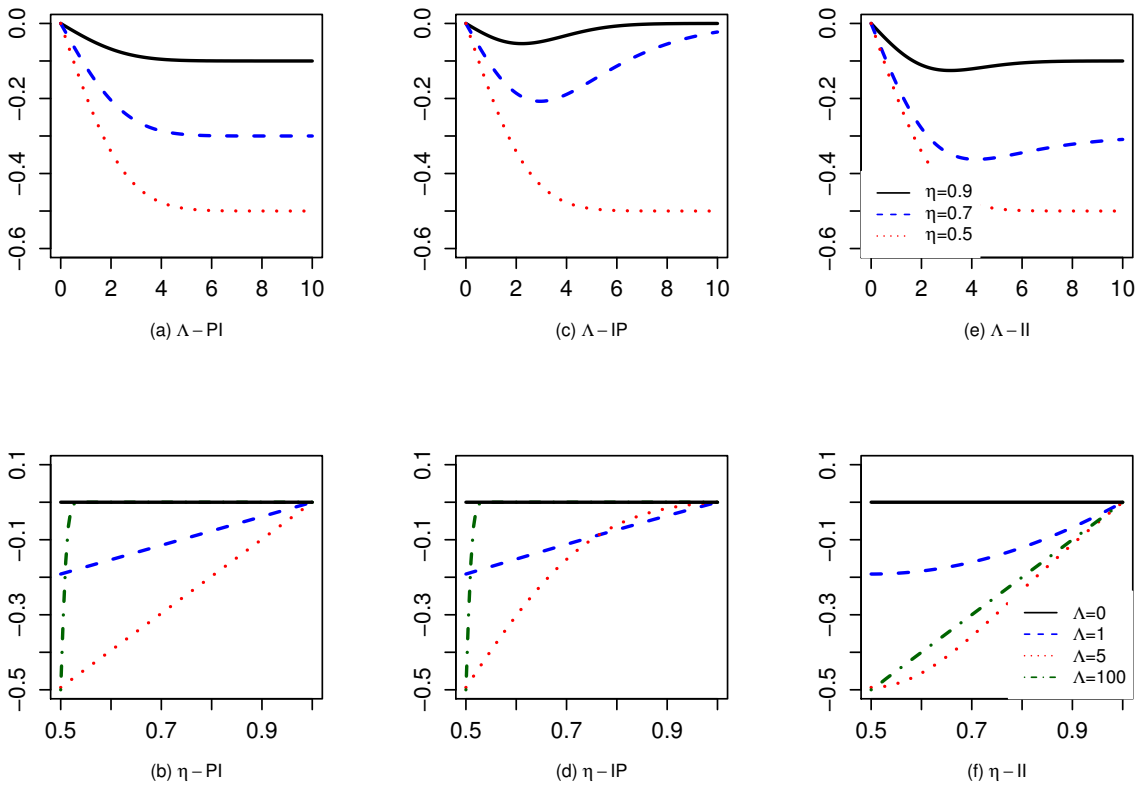

Fig. 2: Recall deviation as a function of the separation between classes, Λ, and degree of ground-truth imperfection, η, in the PI, IP, and II cases.

Interestingly, it turns out that $\Delta_{Re}^{\mathrm{II}}(i)$ can be decomposed in the two previously cases, PI and IP. The first part, $\Delta_{Re}^{\mathrm{PI}}(i)$, accounts for the impact of the imperfect ground-truth on the estimation of the performance metrics, and the second one, which includes $\Delta_{Re}^{\mathrm{IP}}(i)$ and $\Delta_{Re}^{\mathrm{IP}}(j)$, accounts for the impact on the classifier construction. This part has two terms, which depend on $Pr_Z(i)$, the probability that an object assigned to class i by the imperfect ground-truth truly belongs to class i. In fact, when a classifier constructed using an imperfect ground-truth produces a classification error, it can either affect a correctly labeled object (the first term) or an incorrectly labeled one (second term). Thus, the second term corresponds to a sequence of two errors, a classification error and a labeling error, which ends up producing a correct decision, which explains why this term contributes to a decrease in the magnitude of the recall deviation.

Under the simplifying assumptions, the recall deviation is given by:

$$\Delta_{Re}^{\mathrm{II}}(i) = \Delta_{Re}^{\mathrm{PI}}(i) + (2\eta - 1)\,\Delta_{Re}^{\mathrm{IP}}(i).$$

The recall deviation is now higher than in the PI and IP cases, but the limiting values coincide with the PI case (Figures 2.e and 2.f). Thus, for a fixed η, $\Delta_{Re}^{\mathrm{II}}(i) \to 0$ if $\Lambda \to 0$ and $\Delta_{Re}^{\mathrm{II}}(i) \to -(1-\eta)$ if $\Lambda \to +\infty$. Moreover, for a fixed Λ, η, $\Delta_{Re}^{\mathrm{II}}(i) \to 0$ if $\eta \to 1$ and $\Delta_{Re}^{\mathrm{II}}(i) \to 1/2 - \Phi(\Lambda/2)$ if $\eta \to 1/2$.

C. Dataset, traffic object definition, and features

In order to obtain a perfect ground-truth we arranged a small private laboratory network, with a set of PCs interconnected through a switch, and Internet access provided through a router/firewall. There were 17 PCs, 10 for users producing licit traffic, one acting as a server, one for measurements, and 5 assigned to the attacks.

Multiple measures were adopted to assure that the licit traffic was free of attacks. We configured the PCs with a minimal Linux Ubuntu distribution running from live CDs to assure that no virus, worm, Trojan horse, or active Botnet were present. We also configured the forwarding table of the switch with static entries only, and disabled its learning capability, to prevent any local spoofing attack. Internet access is provided by a router/firewall (Cisco ASA 5510), which performs traffic inspection to allow the entrance of returning sessions only and contains access lists configured to prevent remote IP spoofing attacks.

The licit traffic is a mixture of video (over TCP), Web browsing (HTTP) and file sharing (BitTorrent), which are the predominant Internet applications. In order to generate this

traffic, we asked a set of 10 users to use these applications from inside our lab network following their normal behavior. The users were restricted to an (access) list of (safe) servers, configured in the router/firewall, to avoid the possibility of becoming infected while navigating. To produce video traffic users where given a (safe) playlist of YouTube videos. Moreover, while accessing YouTube videos and browsing, a browser adblock add-on was active, to avoid infection via advertisements. File sharing traffic were generated using the Linux Transmission BitTorrent client and providing users with a (safe) list of torrent files.

The attacks were produced only within the lab network and only to and from dedicated PCs. We concentrated on two broad classes of attacks, tightly related with Botnets, currently the main security threat in the Internet: port-scans and snapshots. Port-scans are usually the first activity of an infected Bot and are used to identify other targets vulnerable to infection and Botnet spreading. Snapshots, performed in the second phase of the attack, are a type of identity theft aimed at stealing personal information. In our setup, the port-scans were produced by NMAP, with one second interval between SYN probes. Snapshots were emulated by sending small files, from FTP clients installed at the attacking PCs to an FTP server. The files had 120 Kbytes, to simulate a small screen area of 335×180 pixels around the cursor, and were sent with inter-arrival times following an exponential distribution with mean 120 seconds, to simulate the user clicks.

The packet trace was collected by mirroring all traffic passing through the switch to a PC dedicated to measurements. The measurements were performed during three days, starting on June 18, 2014, for approximately 8 hours per day (some users were not always active). We captured the first 64 bytes of all packets.

The packets were aggregated in traffic objects that we call *datastreams*. A datastream aggregates all packets observed in a 5 minutes interval that have the same IP source address and one of the TCP port numbers equal. This object definition is not typical, but we claim it is very well suited to the identification of Internet applications (including Internet attacks). It aggregates on a single object all traffic of one application that enters or leaves one machine. In this way, the traffic of applications and attacks that open several TCP sessions, e.g. HTTP, BitTorrent, and port-scans, is placed in the same traffic object, which would not occur with the (more common) 5 tuple definition (source and destination IP address, source and destination port number, and protocol type). In this way, the datastreams are able to capture user and high-level application behavior, which in many cases has a better discriminating power.

The capture file was processed by tshark to extract the datastreams and obtain 5 traffic characteristics computed in 0.1 seconds intervals: number of upstream packets (PUp), number of downstream packets (PDw), number of upstream bytes (BUp), number of downstream bytes (BDw), and number of active TCP sessions (Ses). Then, for each characteristic we computed initially 8 summary statistics: minimum (min), 1^{st} quartile (Q_1), median (med), mean (m), 3^{rd} quartile (Q_3), maximum (max), standard deviation (sd) and median absolute deviation (MAD); this gives a total of 40 features.

TABLE III: Distribution of datastreams, per application.

Application	5% anomalies	20% anomalies
Video	123	104
File sharing	26	22
Web browsing	326	274
Port-scans	19	75
Snapshots	6	25

D. Simulation study

Using the collected datastreams we composed three scenarios, each comprising 500 datastreams, that differ on the number of classes and on the level of anomalous traffic: scenarios 1 and 2 have two classes and scenario 3 has five; scenario 1 has 5% anomalous traffic and scenarios 2 and 3 have 20%. Scenarios 1 and 2 illustrate an anomaly detection problem with light and heavy anomalous traffic, respectively, and scenario 3 illustrates an Internet application identification problem. Following several traffic reports and forecasts, we used a breakdown of 60% for video, 20% for Web browsing and 20% for file sharing, within the licit applications [31], [32]. Since these percentages refer to traffic volume, e.g. expressed in bytes, the number of datastreams per application was determined accounting for the average datastream size, which in our case was 28.4 MBytes for video, 45.2 MBytes for File sharing, and 3.57 Mbytes for Web browsing. We also considered a breakdown on the number of anomalous datastreams of 75% for port-scans and 25% of snapshots [33]. The distribution of the number of datastreams per application is shown in Table III.

In order to reduce the total number of features we have used the maxMIFS feature selection method [34] assuming only two classes. With 5% anomalous traffic, the following features were selected: Ses-max, BDw-m, PDw-MAD, PUp-m, and BUp-MAD. Likewise, with 20% anomalous traffic, the following ones were selected: Ses-max, BUp-MAD, BDw-m, BUp-sd, and BUp-max.

The imperfect ground-truth characteristics are shown in Table IV for two classes, and in Table V for five classes. In the first case, class 1 corresponds to licit traffic and class 2 to anomalous one. In the second case, class 1 corresponds to File sharing, class 2 to Video, class 3 to Web browsing, class 4 to Port-scans and class 5 to Snapshots. Note that there is significant confusion between Port-scans and HTTP. This has a practical meaning since Port-scans from the same source will be gathered in the same datastream, mimicking the behavior of persistent HTTP connections.

We used four classifiers in our study: Optimal, NB, C4.5, and kNN ($k = 1$). The first one is taken as reference but, as discussed in section IV-A, it assumes that the features follow a multivariate normal distribution with equal covariance matrices in each class. The three last classifiers are among the most popular in the area of Internet traffic classification.

TABLE IV: Imperfect ground-truth characteristics for two classes, $P(Z = z|Y = y)$, $z, y = 1, 2$.

		y	
		1	2
z	1	0.95	0.10
	2	0.05	0.90

TABLE V: Imperfect ground-truth characteristics for five classes, $P(Z = z|Y = y)$, $z, y = 1, 2, ..., 5$.

		y				
		1	2	3	4	5
z	1	0.940	0.020	0.020	0.050	0.020
	2	0.020	0.930	0.030	0.025	0.020
	3	0.020	0.020	0.890	0.500	0.020
	4	0.010	0.015	0.030	0.300	0.090
	5	0.010	0.015	0.030	0.125	0.850

For all cases except PP, the simulation procedure starts by constructing the imperfect ground-truth according to the characteristics defined in tables IV and V. This step produces a dataset with 500 datastreams, with a percentage incorrectly labeled. Then, depending on the specific case, the classifier is trained with the perfect or imperfect-ground truth, and the recall and precision are estimated using the perfect ground-truth, the imperfect ground-truth, or the LCM. We perform 1000 replicas of each simulation, to obtain average recall and precision estimates.

In order to facilitate the interpretation of the results, we adopt as performance measure the Euclidean distance between vectors that summarize the average recall and precision of all classes obtained in each case, i.e.

$$(\widehat{Re(1)}, \widehat{Pr(1)}, \ldots, \widehat{Re(k)}, \widehat{Pr(k)})^t.$$

The distance is always calculated against the vector that corresponds to the PP case.

The results allow extracting conclusions about the resilience to ground-truth imperfections of the classification process (IP), of the estimation process (PI), and of both (ILCM). Remember that IP and PI presume that a perfect ground-truth is available, and are thus unrealistic from a practical perspective.

The first and most important conclusion is that ILCM decreases significantly the estimation errors when compared with II. This result is clear in all scenarios, and confirms the superiority of this methodology to benchmark Internet traffic classifiers.

In the scenarios with two classes, the ILCM error obtained kNN is slightly higher than the remaining ones and close to the IP error. This is because in kNN (with $k = 1$) the classification is based on the sole distance to the closest object, which renders this method very sensitive to training with an imperfect ground-truth. This result is not so pronounced in the case of five classes, because with two classes the percentage of objects in each class is more unbalanced. With two classes, the ILCM recall of class two is 0.705 and 0.850 for scenarios 1 and 2, much lower than that of class 1 which is 0.950 and 0.949, respectively.

Again in the scenarios with two classes, in can be seen that better results are obtained with 20% of anomalous traffic. This is simply due the fact that, on average, more information is available on the less numerous classes (the anomalies).

NB has high IP errors in all scenarios, which shows it is not very resilient to ground-truth imperfections. Indeed, NB ignores correlations of features within classes, which is an important aspect to account for.

Except for the Optimal classifier with 5% of anomalous traffic, the performance obtained with ILCM is better than that of IP. This is because ILCM uses information from all classification methods to obtain the estimates, while IP uses only information from a single one. This result clearly illustrates the strength of latent class models.

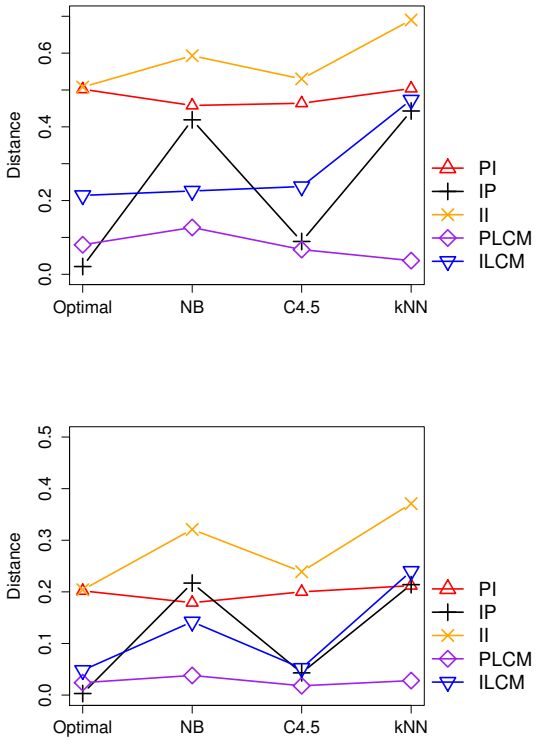

Fig. 3: Distance relative to PP case, two classes, 5% (upper) and 20% (lower) of anomalies.

V. Conclusions

Correctly benchmarking Internet traffic classifiers requires a perfect ground-truth, i.e. a set of traffic objects correctly labeled according to the class they belong to, which is needed both to train the classifier and to estimate its performance. However, a perfect ground-truth is difficult to obtain, due to the growing percentage of cyphered traffic, the sophistication of network attacks, and the constant updates of Internet applications. In this paper we showed, both theoretically and through simulation, that evaluating the performance of Internet

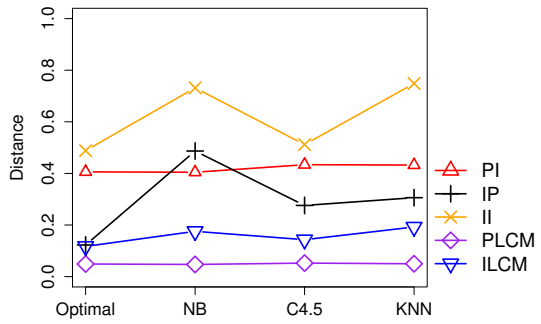

Fig. 4: Distance relative to PP case, five classes, 20% of anomalies.

traffic classifiers using an imperfect ground-truth can lead to large estimation errors. We then proposed a latent class model which overcomes this problem by combining estimates of several classifiers over the same dataset. Results obtained with a high-quality ground-truth, mixing real licit traffic with emulated attacks, showed that the proposed latent class model produces very good performance estimates under mild levels of ground-truth imperfection, and can be used to correctly benchmark Internet traffic classifiers.

REFERENCES

[1] C. Gates and C. Taylor, "Challenging the anomaly detection paradigm: A provocative discussion," in *Proc. of the 2006 Workshop on New Security Paradigms (NSPW '06), Dagstuhl, Germany, Sept. 19-22, 2006*, pp. 21–29.

[2] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in *Proc. of the 31st IEEE Symposium on Security and Privacy (SP'10), Berkeley/Oakland, California, USA, May 16-19, 2010*, pp. 305–316.

[3] A. Lakhina, M. Crovella, and C. Diot, "Mining anomalies using traffic feature distributions," in *Proc. of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, Philadelphia, Pennsylvania, USA, Aug. 22-26, 2005*, pp. 217–228.

[4] H. Ringberg, A. Soule, J. Rexford, and C. Diot, "Sensitivity of PCA for traffic anomaly detection," *SIGMETRICS Performance Evaluation Review*, vol. 35, no. 1, pp. 109–120, June 2007.

[5] D. Brauckhoff, K. Salamatian, and M. May, "Applying PCA for traffic anomaly detection: problems and solutions," in *Proc. of INFOCOM'09, Rio de Janeiro, Brazil, April 19-25, 2009*, pp. 2866–2870.

[6] S. S. Kim and A. L. N. Reddy, "Statistical techniques for detecting traffic anomalies through packet header data," *IEEE/ACM Transactions on Networking*, vol. 16, no. 3, pp. 562–575, June 2007.

[7] G. Thatte, U. Mitra, and J. Heidemann, "Parametric methods for anomaly detection in aggregate traffic," *IEEE/ACM Transactions on Networking*, vol. 19, no. 2, pp. 512–525, April 2011.

[8] C. Pascoal, M. R. Oliveira, R. Valadas, P. Filzmoser, P. Salvador, and A. Pacheco, "Robust feature selection and robust PCA for Internet traffic anomaly detection," in *Proc. of INFOCOM'12, Orlando, Florida, USA, March 25-30, 2012*, pp. 1755–1763.

[9] A. Kind, M. Stoecklin, and X. Dimitropoulos, "Histogram-based traffic anomaly detection," *IEEE Transactions on Network and Service Management*, vol. 6, no. 2, pp. 110–121, June 2009.

[10] R. Lippmann, D. Fried, I. Graf, J. Haines, K. Kendall, D. McClung, D. Weber, S. Webster, D. Wyschogrod, R. Cunningham, and M. Zissman, "Evaluating intrusion detection systems: the 1998 DARPA off-line intrusion detection evaluation," in *Proc. of the DARPA Information Survivability Conference and Exposition (DISCEX '00), 25-27 Jan. 2000*, vol. 2, 2000, pp. 12–26.

[11] H. Ringberg, M. Roughan, and J. Rexford, "The need for simulation in evaluating anomaly detectors," *SIGCOMM Computer Communication Review*, vol. 38, no. 1, pp. 55–59, 2008.

[12] S. Floyd and V. Paxson, "Difficulties in simulating the Internet," *IEEE/ACM Transactions on Networking*, vol. 9, pp. 392–403, August 2001.

[13] A. Moore and D. Zuev, "Internet traffic classification using bayesian analysis techniques," *Performance Evaluation Review*, vol. 33, pp. 50–60, 2005.

[14] T. Karagiannis, K. Papagiannaki, and M. Faloutsos, "Blinc: multilevel traffic classification in the dark," *SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 4, 2005.

[15] L. Bernaille, R. Teixeira, and K. Salamatian, "Early application identification," in *Proc. of the 2006 ACM CoNEXT Conference (CoNEXT '06)*, 2006.

[16] D. Bonfiglio, M. Mellia, M. Meo, D. Rossi, and P. Tofanelli, "Revealing skype traffic: When randomness plays with you," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 4, pp. 37–48, August 2007.

[17] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson, "Offline/realtime traffic classification using semi-supervised learning," *Performance Evaluation*, vol. 64, no. 9-12, pp. 1194–1213, October 2007.

[18] T. Nguyen and G. Armitage, "A survey of techniques for Internet traffic classification using machine learning," *IEEE Communications Surveys and Tutorials*, vol. 10, no. 4, pp. 56–76, 2008.

[19] H. Kim, K. Claffy, M. Fomenkov, D. Barman, M. Faloutsos, and K. Lee, "Internet traffic classification demystified: myths, caveats, and the best practices," in *Proc. of the 2008 ACM CoNEXT Conference (CONEXT '08)*, 2008.

[20] M. Pietrzyk, J.-L. Costeux, T. En-Najjary, and G. Urvoy-Keller, "Challenging statistical classification for operational usage: the ADSL case," in *Proc. of the 9th ACM SIGCOMM Conference on Internet Measurement (IMC '09)*, 2009.

[21] W. Li, M. Canini, A. W. Moore, and R. Bolla, "Efficient application identification and the temporal and spatial stability of classification schema," *Computer Networks*, vol. 53, no. 6, pp. 790–809, 2009.

[22] G. Aceto, A. Dainotti, W. de Donato, and A. Pescape, "Portload: Taking the best of two worlds in traffic classification," in *Proc. of the INFOCOM'10 Workshops, March 2010*, pp. 1–5.

[23] Y. Jin, N. Duffield, J. Erman, P. Haffner, S. Sen, and Z.-L. Zhang, "A modular machine learning system for flow-level traffic classification in large networks," *ACM Transactions on Knowledge Discovery from Data*, vol. 6, no. 1, pp. 4:1–4:34, March 2012.

[24] A. Dainotti, A. Pescape, and K. Claffy, "Issues and future directions in traffic classification," *IEEE Network*, vol. 26, no. 1, pp. 35–40, January 2012.

[25] A. Baughman, K. Bisgard, M. Cortese, W. Thompson, G. Sanden, and P. Strebel, "Utility of composite reference standards and latent class analysis in evaluating the clinical accuracy of diagnostic tests for Pertussis," *Clinical Vaccine Immunology*, vol. 15, no. 1, pp. 106–114, 2008.

[26] P. Albert and L. Dodd, "A cautionary note on robustness of latent class models for estimating diagnostic error without a gold standard," *Biometrics*, vol. 60, pp. 427–435, 2004.

[27] D. Bartholomew, M. Knott, and I. Moustaki, *Latent Variable Models and Factor Analysis: A Unified Approach*. Wiley, 2011.

[28] P. Lazarsfeld and N. Henry, *Latent structure analysis*. Boston: Houghton Mifflin, 1968.

[29] A. Subtil, M. Oliveira, and L. Gonçalves, "Conditional dependence diagnostic in the latent class model: A simulation study," *Statistics & Probability Letters*, vol. 82, no. 7, pp. 1407–1412, 2012.

[30] R. Johnson and D. Wichern, *Applied Multivariate Statistical Analysis*, 6th ed. Prentice-Hall, Inc., 2007.

[31] "Cisco Visual Networking Index: Forecast and Methodology, 2013–2018," Tech. Rep., June 2014.

[32] "Sandvine, Global Internet Phenomena Report, 2013," Tech. Rep., 2013.

[33] K. Goseva-Popstojanova, G. Anastasovski, A. Dimitrijevikj, R. Pantev, and B. Miller, "Characterization and classification of malicious web traffic," *Computers & Security*, vol. 42, pp. 92 – 115, 2014.

[34] C. Pascoal, "Contributions to variable selection and robust anomaly detection in telecommunications," Ph.D. dissertation, Instituto Superior Técnico, Technical University of Lisbon, 2014.