第 V 部

ネットワークトラフィック統計情報の収集と解析

第 5 部
ネットワークトラフィック統計情報の収集と解析

第 1 章　MAWI WG について

MAWI（Measurement and Analysis on the WIDE Internet）ワーキンググループは、トラフィックデータの収集と解析を研究対象とした活動を行なっている。

MAWI WG では WIDE プロジェクトの特徴を活かした研究をするため、「広域」「多地点」「長期的」の三つの項目に重点を置いたトラフィックの計測・解析を行っている。広域バックボーンでのデータ収集はバックボーンを持っている WIDE だからできる事である。分散管理されるインターネットの状態を把握するためには、多地点で観測したデータを照らし合わせることが欠かせない。また、長期的にデータを収集し蓄積するために、ワーキンググループとしての継続的な活動が役に立つ。

計測技術はほとんどの研究分野で必要となるため、MAWI ワーキンググループは WIDE 内の他のワーキンググループと連係をとりながら活動をしている。また、グローバルなインターネットの挙動を把握するために、海外の組織とも積極的に協調して研究活動をしている。

第 2 章　MAWI WG 2010 年度の活動概要

今年度の報告書では、第 3 章で、計測に関する国際協調について報告する。WIDE では、CAIDA/CASFI とフランスの CNRS との間で計測に関する包括的な共同研究を行なっていて、それぞれの組織と複数のテーマについて共同研究を進め、定期的なワークショップの開催や研究者交換を行なっている。

次に、第 4 章において、MAWI で公開してきているトラフィックデータをもとに、種々の異常検出手法のベンチマークを行い結果を公開する MAWILab の活動について報告する。

第 3 章　計測に関する 2010 年度国際協調活動報告

3.1 はじめに

WIDE プロジェクトは多くの国際協調活動を行なっているが、近年は計測研究の重要性が増している。これは、インターネット研究において、グローバルなレベルでその挙動を把握する必要性と難しさが認識されてきたためである。

WIDE では、現在、CAIDA（the Cooperative Association for Internet Data Analysis）との間で計測に関する共同研究を行なっている。また、フランスの CNRS（The Centre National de la Recherche Scientifique）と新たに 2 年間の共同研究を開始したので報告する。

3.2 CAIDA および CASFI との共同研究

CAIDA と WIDE プロジェクトは、2003 年度から計測に関する包括的な共同研究を行なっている。2008 年からは韓国の CASFI チームも交えたワークショップを行っていて、今年度は 2010 年 4 月に CAIDA-WIDE-CASFI 計測ワークショップを大阪で開催した。CAIDA から 1 名、WIDE から 15 名、CASFI から 10 名が参加し、それぞれのチームからの発表に加え、今後の共同研究に関して議論を行った。

- 第 3 回 CAIDA-WIDE-CASFI 計測ワークショップ

  2010 年 4 月 24–25 日　慶應義塾大学大阪リバーサイドキャンパス

- インターネット計測デーの実施

  CAIDA との共同研究の一貫として 2007 年から行っているインターネット計測デーを 4 月中旬に実施した。WIDE ではトランジット回線の 83 時間のパケットトレース等を収集し公開した。

3.3 CNRS との共同研究

WIDE とフランスの大学連合である CNRS は、2009 年に終了した 3 年間の共同研究に引続き、2010–2011 年の 2 年間の計測とセンサーネットに関する共同研究を始めた。

本共同研究では、次世代ネットワークのための再現可能なデータセントリックベンチマークインフラストラクチャおよびその方法論に関する研究を行う。データセットとして、インターネットトラフィックならびにセンサネットワークトラフィックに着目する。

本研究の目的の一番目は、異常検出やアプリケーションタイプの同定等のネットワークトラフィック解析手法を定量的に相互比較可能なベンチマークフレームワークを確立することである。これにより、解析手法ごとの特性を理解・比較することが容易となる。第二にデータセット（インターネットバックボーントラフィックおよびセンサネットワークデータ）に対して、上記フレームワークを用いて各種解析手法を適用することで、正解データセットを構築し、その結果を研究コミュニティに公開することである。データセットとしては、日米インターネットバックボーンの日々の定点観測データである MAWI データベース（mawi.wide.ad.jp、2001–2010、40 GB）、および、国内数十拠点で日常的に気象データ収集を行っている Live E! データベース（www.live-e.org、2008–2010、10 GB）である。これらのデータを用いて、トラフィックタイプの同定、異常検出、時間発展等に関する解析を行った結果を用いて、データベースに対してラベル付けを行う。

日本側では、画像処理に基づく異常検出アルゴリズム、日仏で開発したマルチスケールガンマモデルに基づく異常検出アルゴリズム、主成分分析に基づく異常検出アルゴリズムの性能を MAWI トラフィックデータセットを用いて評価し、手動による異常トラフィックならびにアプリケーションタイプのラベリングを行う。これにより、完全な正解データセットではないものの、比較可能な部分正解データセットを構築する。また、Live-E! センサデータのデータベース整備を進め、多変量・マルチスケールデータの抽象化やデータ圧縮を考慮したデータ収集・配信アーキテクチャの設計と実装を行う。

フランス側では、日仏共同で開発を進めている Minimum spanning tree およびトラフィック特徴量に基づくトラフィック分類器に関する研究を進め、同様に MAWI データベースを用いて評価を行う。これらの手動によるデータ比較から、両国でどのような形でデータへの正解のラベル付けを行うかを検討する。さらに、複数拠点の Live-E! センササンプルデータを用いて、気象データ時系列群の時空間的特性をセンサデータ間の時空間相関、定常性、予測可能性に着目して研究を進める。

2010 年 3 月にフランス側研究代表者が来日した際、非公式な打ち合わせを行い、2010 年 10 月に東京にて第一回ミーティングを実施した。これらの打ち合わせは、主として、研究結果の進捗発表、問題点の共有、ならびに次の半年間の研究計画の設定/修正を目的としている。また、2010 年 9 月に日本側から東京大学江崎研究室の本舘君を 4 週間 ENS-Lyon に派遣し、トラフィック異常検出アルゴリズムの性能解析に関する共同研究を行った。

3.4 まとめ

インターネットの計測研究では、国際的な協調による広域なデータ収集、しかも長期に渡る地道な努力が重要である。今後は、これまでに築いた関係をベースに、さらに協調の幅を広げると同時に、具体的な成果を出す努力をしていく。

第 4 章　Benchmarking Anomaly Detectors: A Systematic Methodology using Real Internet Traffic

4.1 Introduction

Anomalies in Internet traffic penalize legitimate users from accessing optimal network resources. High-speed backbone traffic is particularly degraded, but analyzing such traffic is a complicated task due to the amount of data, the lack of payload data, the asymmetric routing and the use of sampling techniques. Consequently, anomaly detection has received a lot of attention in the last decade, and numerous detectors have been proposed.

In previous works, we proposed two tools to diagnose and detect anomalous traffic:

- A visualization tool has been proposed to

display, explore, and understand network traffic focusing on anomalies[57]. It displays traffic on different temporal and spatial (address and port) scales and lets users navigate network data by using a simple interface. Different graphical representations are used to highlight anomalies quickly, and textual packet information about corresponding plotted points are provided. The proposed tool provides good support for understanding Internet traffic behavior and for manually evaluating the effectiveness of anomaly detection method. The tool directly reads dump files and uses no intermediate database in daily operations. Also, several examples emphasizing specific patterns for various anomalies have been identified.

- Moreover, we proposed an anomaly detection method that uses a pattern recognition technique to identify anomalies in pictures representing traffic[56, 58]. The main advantage of this method is its ability to detect attacks involving mice flows. We evaluated the effectiveness of this method by analyzing six years of Internet traffic collected from a trans-Pacific link. We showed several examples of detected anomalies and compared our results with those of two other methods. The comparison indicated that the only anomalies detected by the proposed method are mainly malicious traffic with a few packets.

  Moreover, we investigated the relationship between the parameter set of the proposed method and the traffic characteristics[55]. This analysis highlighted that constantly achieving a high detection rate requires continuous adjustments to the parameters according to the traffic fluctuations. Therefore, an adaptive time interval mechanism was proposed to enhance the robustness of the detection method to traffic variations. This adaptive anomaly detection method was evaluated by comparing it to three other anomaly detectors using four years of real backbone traffic. The evaluation revealed that the proposed adaptive detection method was constantly outperforming the other methods in terms of the true positive and false positive rate.

4.1.1 Motivation

While designing the proposed tools, however, we faced significant difficulties in rigorously validating their efficiency. Indeed, evaluating anomaly detectors is challenging due to a lack of ground truth data and a rigorous methodology. Distinct evaluation methodologies are commonly admitted by the research community but they suffer from several drawbacks. Let us classify evaluations methodologies into two categories; those applied to real Internet traffic/anomalies and those applied to simulated traffic/anomalies.

With real anomalies, researchers evaluate anomaly detectors by manually checking the reported alarms[25, 41, 105], or by comparing them to those reported by other anomaly detectors[55, 105]. Sometimes researchers also construct ground truth data by manually inspecting the analyzed traffic[13]. However, these evaluations are hardly comparable, trustworthy, or reproducible, as they require significant human intervention and traffic traces are usually inaccessible due to privacy issues. Moreover, a common shortcoming of these evaluation methodologies is the omission of the false negative rate of the detector, in spite of the fact that this metric is a particularly good indicator for emphasizing the number of missed anomalies and the sensitivity of the detector to different kinds of anomalies.

Simulating anomalies is also a common way to evaluate an anomaly detector[140, 153]. In this case, the parameters of anomalies are tunable (e.g. intensity and time duration), helping researchers to measure the sensitivity of their detectors to particular kinds of anomalies. However, simulating traffic as diverse as it is on the Internet is notoriously difficult[52], especially for anomalous traffic. Consequently, the evaluation of a detector

with simulated anomalies is restricted to certain kinds of anomaly, and thus, is insufficient for measuring the detector performance[150].

4.1.2 Goal

Ideally, an anomaly detector has to be evaluated using ground truth data containing real and nonspecific traffic where a wide range of anomalies is located. This ground truth data should be publicly available to allow all researchers to access the same data set and compare their results. Furthermore, the data set should follow the evolution of the Internet traffic to include traffic from emerging applications and anomalies. There is, however, currently a lack of such crucial ground truth data, and therefore, designing it is our ambitious goal.

Our goal is to locate anomalies in the traffic from the MAWI archive[32], and make it available to researchers so that they can refer to it while evaluating their anomaly detection methods. The main advantages of the MAWI archive are that it is updated daily and it currently contains more than nine years of real publicly available Internet traffic data. However, manually labeling anomalies in such a large data set is certainly impractical, and therefore, the challenge we face is to accurately locate anomalies in an automated and therefore unsupervised manner. The numerous anomaly detectors that have recently been proposed in literature are the main support that will help us reach our goal. Therefore, we are selecting diverse anomaly detectors and combining their results to accurately locate anomalies in the MAWI archive. The synergy between detectors with different theoretical backgrounds allows a more accurate level of detection to be achieved. However, a key issue in combining such diverse detectors is that they report different granularities of the traffic that are difficult to rigorously compare.

Our main contribution is twofold. Firstly, we establish a reliable methodology, which is based on graph and community mining, that compares and combines the results from any anomaly detectors. The proposed method outperforms the combined detectors, and enables us to precisely locate twice more anomalies than the most accurate detector from our experiments. Secondly, we provide our results in the form of benchmark data representing an overview of the state of the art in anomaly detection. The database currently stands as more than nine years of traffic and it is growing along with the MAWI archive. Furthermore, our approach permits the enhancement of the database over time by integrating the results from emerging anomaly detectors. Thus, the proposed database is consistently updated with new traffic and anomaly detectors, and it is a valuable tool to assist researchers designing anomaly detectors.

4.2 Related work

Providing ground truth data to evaluate anomaly detectors is a challenge that has been addressed several times in the past. For example, the DARPA Intrusion Detection Evaluation Program[106] has been a great effort to provide labeled traffic to evaluate intrusion detection systems (IDS). It has been extensively studied, mainly through the KDD Cup 1999 data (KDD'99), and has been a profitable support for researchers. The main distinctions between this work and ours are the size of the network measured and the detectors to be evaluated. The DARPA Intrusion Detection Evaluation Program focuses on the evaluation of IDS and provides labeled LAN traffic where the packet payload is available and flows are complete. Whereas our work focuses on the evaluation of backbone traffic anomaly detectors and we provide labeled backbone traffic where the packet payload is not available, and the flows are incomplete and asymmetric. Furthermore, several critical drawbacks of the KDD'99 have been reported[125]. For example, the traffic data was captured in 1998 thus it contains no traffic from recent applications or anomalies. Therefore, this data must be carefully used

as it is not representative of real traffic[171] and does not contain recent anomalies.

We are conscious of the shortcomings of previous works and we have designed our data set to overcome such issues.

Our approach takes advantage of combination strategies in order to merge the results from several detectors. Although the combination of classifiers is a hot topic in the clustering community[101], only a few works have been conducted in the field of network anomaly detection. For example, Shanbhag and Wolf[160] have studied the combination of five rate-based detectors to accurately identify the real-time variance in traffic volume. They analyzed seven different combination strategies and emphasize that the best strategy improves the accuracy of the overall detectors. Our goal differs from theirs as they aim at detecting anomalies in real time by running several detectors in parallel. Thus, they restrict their study to a particular kind of computationally efficient anomaly detector (rate-based detector), whereas our approach takes advantage of diverse anomaly detectors.

Another recent study on the combination of anomaly detectors was conducted by Ashfaq et al.[8]. They propose a new combination strategy that takes into account the accuracy of the detectors; first, the accuracy of each detector is evaluated on a training data set, and then, the results of the detectors are combined regarding their accuracy. Their results emphasized the benefit of taking into account the detectors accuracies when combining them. Nevertheless, we avoid such methods as they involve a training step that increases the necessity of human intervention. Our approach focuses on unsupervised anomaly detectors that are combined with unsupervised combination strategies.

4.3 Proposed method

The method proposed in this article consists of four main steps that are executed for each traffic trace:

1) Several anomaly detectors analyze the traffic and all reported alarms are collected.
2) The similarities between the reported alarms are uncovered using a **similarity estimator** that groups similar alarms into communities.
3) Each community is investigated and classified by the **combiner**. Namely, the combiner decides if the community has to be reported or ignored depending on the overall outputs of the detectors.
4) The anomalies are located in the traffic.

4.3.1 Similarity estimator

Since the benefit of combining detectors relies on the diversity among the detectors ensemble, we combine various anomaly detectors based on different theoretical backgrounds. Nevertheless, these different anomaly detectors are inherently reporting traffic at different granularities (e.g. flow, host, or packet) that are difficult to systematically compare. For example, if a detector reports one alarm A_1 that is an host, IP_X, and another detector reports two alarms, B_1 and B_2, that are flows, $\langle IP_X, 80, IP_Y, 1234\rangle$ and $\langle IP_X, 80, IP_Z, 2345\rangle$. Then, A_1 is equivalent to B_1 and B_2, however, one cannot state that the three alarms are the same as B_1 and B_2 are obviously reporting distinct traffic. Therefore, a rigorous method precisely measuring the similarities of these three alarms is required.

The role of the similarity estimator presented in this section is to uncover the relations between the outputs of any kinds of anomaly detector. First, it reads the alarms reported by the detectors and the original traffic, and it extracts the traffic described by each alarm. Second, it constructs a graph that highlights the alarm similarities based on the traffic they have in common. Finally, similar alarms are identified by finding communities (i.e. dense connected components) in the graph. The following sections provide more detail on the mechanisms of the similarity estimator. The readers also referred to [53, 54] for more explanations.

4.3.1.1 Traffic extractor

The traffic extractor (called oracle in [54]) retrieves the traffic described by each alarm. Let an alarm be a set of traffic features that designates a particular set of flows identified by the detector, then, the traffic extractor records the association between the alarm and these flows.

4.3.1.2 Graph generator

The graph generator uses the traffic retrieved by the traffic extractor to build an undirected graph highlighting the similarities among all the alarms reported by the detectors. A **node** in this graph stands for an **alarm**, and there is an edge between two nodes if their associated traffic intersects. In addition, an **edge** is weighted with a similarity measure that quantifies the **traffic intersection** of the two alarms it connects. Therefore, the similarity measure enables to discriminate edges connecting dissimilar alarms having an irrelevant number of flows in common. We selected three similarity measures for our experiments, the Jaccard index, the Simpson index and a constant function. Since the Simpson index outperformed the two other metrics, and due to page limitation, only the Simpson index is discussed in this article. The Simpson index is defined as

$$S(E_1, E_2) = |E_1 \cap E_2| / \min(|E_1|, |E_2|)$$

where E_i is the traffic associated with alarm i. This metric ranges $[0, 1]$, where 0 means that the two traffic do not intersect, thus, the two alarms are dissimilar. Whereas, 1 means that they are identical or that one is included in the other.

4.3.1.3 Community mining

The weighted and undirected graph produced by the graph generator highlights the alarm similarities but identical alarms are left undetermined. Nevertheless, identical alarms are distinguishable in the graph as a set of strongly connected nodes, which is also called a community. Identifying the communities in a graph is a problem that has been extensively studied in the past[59]. Although numerous community mining algorithms have been proposed, our interest focuses on those designed for sparse graph since the generated graphs have disconnected nodes (e.g. a false positive alarm reported by one detector). In our experiments we selected a method based on the modularity; the Louvain algorithm[22]. This algorithm has the advantage of locally identifying the communities, thus allowing us to identify groups of a few alarms. Furthermore, this algorithm performs a fast and accurate analysis of the graph[59].

4.3.2 Combiner

The similarity estimator clusters similar alarms into communities, that is, each community represents distinct traffic reported by the detectors. The role of the proposed combiner is to decide whether each community corresponds to an anomalous traffic or not. Therefore, the combiner classifies the communities into two categories, *accepted* and *rejected*, respectively standing for the communities reported as anomalous or those ignored. The class of a community is determined by a combination strategy like those previously studied with machine learning or pattern classifiers[101].

4.3.2.1 Background: combining detectors

A combination strategy is generally categorized as a detector selection or an output fusion. On the one hand, detector selection consists of selecting the detector that is the most suitable for classifying an element (i.e. a community in our case) and makes the same decisions as the single selected detector. Since each element is analyzed by only one detector, this approach is usually a good candidate for performing a quick analysis. However, selecting an appropriate detector is in practice challenging. In particular, the sensitivity of detectors to traffic in network anomaly detection is misunderstood and prevents us from applying such techniques. On the other hand, output fusion makes no assumption on the detectors as it

inspects the results of all the detectors. The output of a detector is assimilated to a vote for a certain class, and the combination strategy refers to a voting procedure.

In order to emphasize the advantages of combining detectors with output fusion let us review perhaps the oldest and best-known strategy, the majority vote. It is a basic, but still powerful way, where the final decision is the simple majority of the detectors outputs (i.e. more than 50 percent of the outputs). The probability of making the correct decision with the majority vote depends on the probability of each detector for providing the correct output, that is:

$$P_{maj}(L) = \sum_{m=\lfloor L/2 \rfloor + 1}^{L} \binom{L}{m} p^m (1-p)^{L-m}$$

where L is the number of detectors and p is their accuracy. The result, also known as the Condorcet Jury Theorem, is as follows; if $p > 0.5$, then $P_{maj}(L)$ is monotonically increasing in L and $P_{maj}(L) \to 1$ as $L \to \infty$. If $p < 0.5$, then $P_{maj}(L)$ is monotonically decreasing in L and $P_{maj}(L) \to 0$ as $L \to \infty$. If $p = 0.5$, then $P_{maj}(L) = 0.5$ for any L. This theorem highlights the benefit of combining reasonable detectors (i.e. with an accuracy $p > 0.5$) over the use of a single detector.

4.3.2.2 Application to traffic anomaly detection

The output of an anomaly detector is a binary value that reports the traffic as anomalous or not. In this article, the candidate traffic are those described by the communities, and a detector votes to report the communities containing at least one of its alarms. Although this is sufficient enough to compute the majority vote, this binary value is too coarse to perform an accurate combination. Furthermore, the votes of the detectors may significantly vary with their parameter tunings. To prevent these difficulties our approach scores the confidence of the detector for each vote.

Running a detector with several parameter sets and measuring the variances of its output

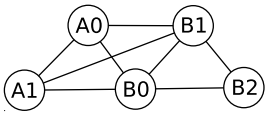


Fig. 4.1. Example of community c_{ex} composed of five alarms. Assuming that the input of the similarity estimator, X_i, consists of the output of three detectors $X = A, B, C$ with three different parameter sets $i = 0, 1, 2$, then the confidence scores are: $\varphi_A(c_{ex}) = 0.66$, $\varphi_B(c_{ex}) = 1.0$ and $\varphi_C(c_{ex}) = 0.0$.

quantifies its parameter sensitivity to vote for a community. Hereafter we refer to a certain detector with a specific parameter set as a **configuration**. The outputs of all the configurations are merged with the similarity estimator and the variance in the outputs is computed by inspecting each community. We define the **confidence score** φ of a detector d for a community c as:

$$\varphi_d(c) = \phi_d(c)/T_d$$

where T_d is the total number of configurations with the detector d and $\phi_d(c)$ is the number of these configurations that reports at least one alarm belonging to the community c. The confidence score is a continuous value that ranges $[0, 1]$, 0 representing that the detector ignores the community whereas 1 means that every configurations with the detector identify the community. For example, Fig. 4.1 is a community c_{ex} composed of five alarms. Assuming that the input of the similarity estimator, X_i, consists of the output of nine configurations corresponding to three detectors $X = A, B, C$ with three different parameter sets $i = 0, 1, 2$, then the confidence scores for this community are: $\varphi_A(c_{ex}) = 0.66$, $\varphi_B(c_{ex}) = 1.0$ and $\varphi_C(c_{ex}) = 0.0$.

4.3.2.3 Combination strategies

Average, Minimum, & Maximum In this section we present three simple combination strategies that differently aggregate the confidence scores of each community with a metric μ.

They classify a community c as accepted only if its aggregated value $\mu(c) > 0.5$.

Aggregating the confidences score of a community by averaging them allows us to take into account the decisions from all the detectors. Formally, for a community c and using L detectors, the average is defined as: $\mu(c) = \frac{1}{L}\sum_{i=1}^{L}\varphi_i(c)$. In the example shown in Fig. 4.1 the average of all the confidence scores equals 5/9, and thus, this combination strategy classifies the community c_{ex} as accepted.

A pessimistic decision is to classify a community as accepted only if all the detectors support this decision. Consequently, the ratio of false positive is substantially reduced at the cost of an increase in the ratio of true negative. This strategy consists in selecting the minimum confidence score. Formally, the decision made for the community c depends on its minimum confidence score, defined as: $\mu(c) = \min_i\{\varphi_i(c)\}$. In the example shown in Fig. 4.1, the minimum of all the confidence scores is 0, and thus, this combination strategy classifies the community c_{ex} as rejected.

Contrarily, an optimistic decision is to classify a community as accepted only if at least one detector supports this decision. Consequently, the ratio of true positive is substantially increased, but so is the ratio of false positive. This strategy consists in selecting the maximum confidence score. Formally, the decision made for the community c depends on its maximum confidence score, defined as: $\mu(c) = \max_i\{\varphi_i(c)\}$. In the example shown in Fig. 4.1, the maximum of all the confidence scores is 1, and thus, this combination strategy classifies the community c_{ex} as accepted.

Correspondence analysis: SCANN Correspondence analysis[17] is a multivariate statistical technique for analyzing multiway tables. It represents a data set in a lower-dimensional space based on singular value decomposition. Although its role is similar to the principal component analysis one, correspondence analysis is designed for categorical data.

Using correspondence analysis, Merz[127] proposes an unsupervised combination strategy called SCANN. This method stores all the decisions of the detectors in a table, so that each entry is a vector representing the decision of all detectors for a certain community. This table is reduced with correspondence analysis, thereby, the entries are then smaller vectors containing only the main features characterizing the detectors decisions. The benefit of this reduced table is to take into account only significant decisions. For instance, a particularly irrelevant detector is one constantly making the same decision; in the first table built by SCANN this detector decisions are constant values that are then ignored in the reduced table because they do not help for discriminating the communities.

Thus, the reduced table contains the characteristics of each community in a low-dimensional space. SCANN projects onto this low-dimensional space two reference points which are two representative communities unanimously elected by the detectors as accepted or rejected. At the end, the class of each community is determined by the closest representative community in the low-dimensional space.

However, since correspondence analysis is designed for categorical data, SCANN is unable to deal with the confidence scores previously proposed. Therefore, in order to still take advantage of different parameter tunings our implementation of SCANN consider the different configurations outputs (binary values) as its input.

4.4 Evaluation

4.4.1 Data set

The traffic we are labeling is from the MAWI (Measurement and Analysis on the WIDE Internet) archive samplepoints B and F[32]. This archive contains daily traces representing 15 minutes of traffic captured from a trans-Pacific link between Japan and the United States. The data is publicly available as packet payloads are omitted and IP addresses are anonymized. MAWI

started in January 2001, and thus, currently contains more than 9 years of traffic. Since 2001, the link has been updated three times, originally it was an 18 Mbps CAR on a 100 Mbps link, but it was updated to a full 100 Mbps link in 2006/07/01 and is currently a 150 Mbps link since June 2007. MAWI has enabled researchers to study Internet traffic characteristics[24, 66, 96], anomaly detectors[41, 55], and traffic classifiers[99].

4.4.2 Anomaly detectors

We implemented four unsupervised anomaly detectors based on distinct statistical-analysis techniques. As they report traffic at different granularities, the proposed similarity estimator is necessary to compare their results. The confidence score for each detector is obtained by tuning them with three different parameter sets that correspond to an arbitrarily optimal, sensitive and conservative setting. Thus, in our experiments the input of the proposed method consists in 12 outputs standing for all the configurations (4 detectors using 3 parameter tunings). The main ideas of the four detectors are as follows.

(1) Principal component analysis (PCA) is an unsupervised technique highlighting the main features of the data. This is perhaps the most studied technique for anomaly detection in backbone traffic. It was first applied by Lakhina et al.[105], and it has received much attention in the last few years[151, 153]. The key idea underlying a PCA-based anomaly detector is the extraction of the main features defining a normal traffic behavior using PCA, then the distinct traffic is reported as anomalous. An inherent problem with PCA-based detectors is the retrieval of the original traffic features of the anomalous traffic[151]. In our experiments we overcame this difficulty by using random projection techniques (sketches)[95, 105]. This approach enables our PCA-based detector to report the source IP address of the identified anomalous traffic.

(2) Dewaele et al. introduced an anomaly detection method based on sketching and multi-resolution gamma modeling[41]. In a nutshell, the traffic is split into sketches and modeled using Gamma distribution. Traffic that is distant from an adaptively computed reference is reported as anomalous. The sketches are computed twice; the traffic is hashed on source addresses and destination addresses. Thus, this method reports source or destination IP addresses.

(3) The Hough transform is a pattern recognition technique that allows for the identification of a specific shape in a picture. This technique has been applied to several domains including anomaly detection of backbone traffic[55]. The approach proposed in [55] consists of first, monitoring the traffic in a 2-D scatter plot where the anomalous traffic appears as "lines", and second, identifies the anomalies with the Hough transform. The original data is retrieved from the identified plots, and the alarms reported by this method are aggregated sets of flows.

(4) The work presented in [25] detected the prominent changes in traffic by applying the Kullback-Leibler (KL) divergence to several kinds of histograms that monitor distinct traffic features. Furthermore, association rule mining allows for the extraction of the sets of traffic features that describes the anomalies detected by the histograms. Thus, the alarms reported by this anomaly detector are association rules, namely 4-tuples (source and destination IP addresses, source and destination port numbers) where elements can be omitted.

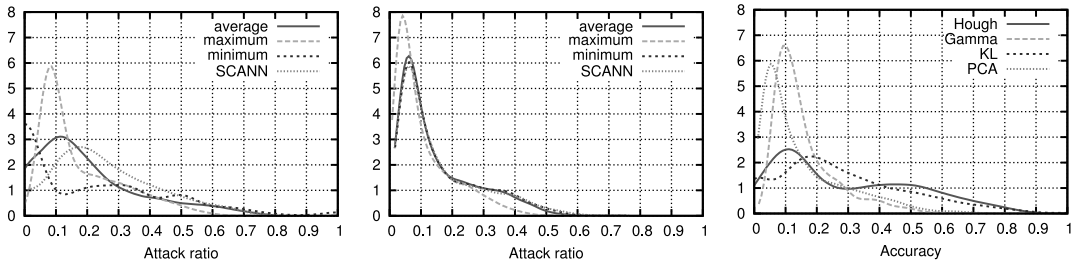
4.4.3 Attack ratio

In this work, measuring the accuracy of the four combination strategies is a contradictory task due to the lack of ground truth data. We bypass this issue by inspecting the results of the combiner with the heuristics of Table 4.1.

The heuristics label the communities reported by the similarity estimator into three groups: "Attack", "Special", and "Unknown". Since a relevant combination strategy is presumed to report the largest proportion of the communities labeled

Table 4.1. Heuristics labeling the traffic corresponding to a community into three categories ("Attack", "Special", and "Unknown"). These are originated from the anomalies previously reported[24, 55] and the manual inspection of MAWI.

Label	Category	Details
Attack	Sasser	Traffic on ports 1023/tcp, 5554/tcp or 9898/tcp
Attack	RPC	Traffic on port 135/tcp
Attack	SMB	Traffic on port 445/tcp
Attack	Ping	High ICMP traffic
Attack	Other attacks	Traffic with more than 7 packets and: SYN, RST or FIN flag $\geq$ 50% Or, http, ftp, ssh, dns traffic with SYN flag $\geq$ 30%
Attack	NetBIOS	Traffic on ports 137/udp or 139/tcp
Special	Http	Traffic on ports 80/tcp and 8080/tcp with less than 30% of SYN flag
Special	dns, ftp, ssh	Traffic on ports 20/tcp, 21/tcp, 22/tcp or 53/tcp&udp with less than 30% of SYN flag
Unknown	Unknown	Traffic that does not match other heuristics



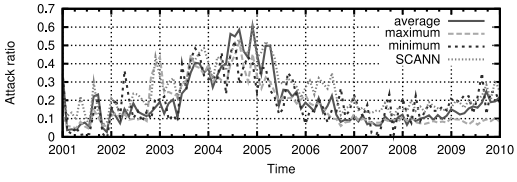
(a) PDF of attack ratio for accepted communities (large probability for high attack ratio is better)

(b) PDF of attack ratio for rejected communities (large probability for low attack ratio is better)

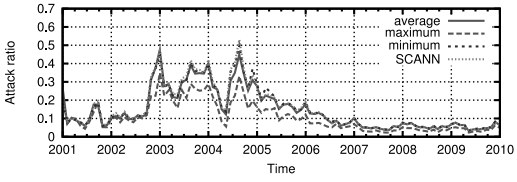
(c) PDF of attack ratio for detectors (large probability for high attack ratio is better)

Fig. 4.2. PDF of attack ratio for four combination strategies and four detectors evaluated on 9 years.

"Attack", we define the **attack ratio** as the amount of communities labeled "Attack" divided by the total number of identified communities. The combination strategies are expected to also report numerous communities labeled "Special" or "Unknown", thus low attack ratio, as the proposed heuristics might label incorrectly several kinds of anomalies. Nevertheless, the attack ratio is a reliable indicator that helps us to identify the best combination strategy; i.e. the one accepting the highest ratio of communities labeled "Attack" (Fig. 4.2(a) and 4.3(a)) and rejecting the lowest ratio of communities labeled "Attack" (Fig. 4.2(b) and 4.3(b)).



(a) Accepted community attack ratio (higher value is better)



(b) Rejected community attack ratio (lower value is better)

Fig. 4.3. Attack ratio of four combining strategies for nine years of MAWI traffic.

4.4.4 Results

4.4.4.1 Comparison of combining strategies

This section evaluates the ability of the four combination strategies to label communities. The analyzed communities are produced by the similarity estimator with the alarms reported by the four detectors on nine years of MAWI traffic and using unidirectional flow as traffic granularity. These communities are classified by the combination strategies into two classes (i.e. accepted and rejected) and the attack ratio of both classes are computed for each day of the analyzed traffic. The results are presented in the form of probability density functions and time-based curves (Fig. 4.2 and 4.3).

Regarding accepted communities, the best combination strategy is SCANN as it features the largest probability for highest attack ratio (Fig. 4.2(a)). Nevertheless, the best combination strategy regarding rejected communities is the maximum because it is the one with the largest probability for lowest attack ratio (Fig. 4.2(a)). Since the prominent variance between the attack ratio probability of the accepted communities and the one of the rejected communities highlights the best combination strategy, our experiments support SCANN as the best strategy for discriminating the communities representing anomalous traffic.

The probability density functions of the four anomaly detectors attack ratio emphasizes that all detectors, except the KL-based one, have an average attack ratio that is inferior to SCANN (Fig. 4.2(c)). Although the KL-based detector attack ratio is close to that of SCANN, the thorough investigation of the SCANN output in Section 4.4.4.2 asserts that SCANN detected twice more traffic than the KL-based detector.

The time evolution of the attack ratio for each combination strategy is depicted in Figures 4.3(a) and 4.3(b). Although the SCANN algorithm is not constantly outperforming the other combination strategies, it never has the worst attack ratio. The low attack ratio of both the accepted and rejected communities from 2007 is due to the simple heuristics listed in Table 4.1 that mislabeled the numerous elephant flows from peer-to-peer traffic and other anomalies using random ports. Still, between 2007 and 2010, the efficiency of SCANN is noticeable as its attack ratio for accepted communities was 2 to 3 times higher than the one for rejected communities.

However, the increase in the attack ratio for rejected communities from 2003 to 2005 (Fig. 4.3(b)) highlights the particular traffic that is missed by the combination strategies. The release of the Blaster worm in August 2003 followed by the release of the Sasser worm in May 2004 were two of the main events reported during this time period. These two worms have substantially affected the main characteristics of the traffic and the four detectors were differently affected by this variance in traffic. The detectors reported numerous alarms that were not related to those of the other detectors, and consequently, the combiner failed in distinguishing several anomalous traffic. Nevertheless, this shortcoming of the combiner is inherently diminished by the combination of more detectors thus increasing the intersection of their outputs. Furthermore, we observed that selecting a single detector to analyze this traffic was also challenging, as the attack ratio of each detector critically fluctuated during this time period.

4.4.4.2 Inspecting the SCANN output

A manual inspection of the SCANN output reveals that several accepted communities contain only alarms from a single detector. Therefore, for the nine years of analyzed traffic, 8 accepted communities were identified by only the PCA-based detector, 325 accepted communities were identified by only the Gamma-based detector, 2467 accepted communities were identified by only the Hough-based detector, and 352 accepted communities were identified by only the KL-based detector. Meaning that 82% of the communities reported exclusively by the KL-based detector are

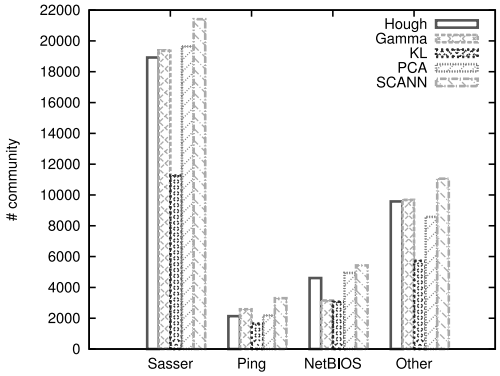


Fig. 4.4. Breakdown of communities accepted by SCANN and labeled "Attack" by heuristics.

accepted by SCANN. These communities highlight the advantage of SCANN over the average combination strategy. Whereas the average combination strategy inherently rejects all the communities reported by a single detector, SCANN performs a finer analysis that emphasizes the output from accurate detectors and allows for the acceptance of small communities identified exclusively by these detectors. Indeed, the SCANN algorithm factorizes the detectors decisions by disregarding the unnecessary ones, thus, SCANN ignores the output of the detectors that are making irrelevant decisions and emphasizes the other results. For example, in our experiments the PCA-based detector output was mainly separated from the outputs of the other detectors. Consequently, SCANN frequently disregarded the PCA-based detector and accepted only 8 of the numerous communities exclusively identified by this detector. Conversely, the Hough-based detector reports more relevant alarms as many are related to those from other detectors, and thus, SCANN selects 2467 communities reported by only this detector.

In our experiments the best detector was the KL-based one (Fig. 4.2(c)). Almost all the alarms from this detector were related to another alarm and are accepted by SCANN. However, about 50% of the communities accepted by SCANN and labeled "Attack" are not identified by the KL-based detector (Fig. 4.4). These communities are mainly reported by the three other detectors and they highlight the high false negative rate (i.e. anomalies missed) of the KL-based detector (Fig. 4.4).

4.4.5 MAWILab

The proposed method assists us in labeling the traffic from the MAWI archive. In accordance with our evaluation, we labeled the traffic using the SCANN combination strategy.

Our labeling of the MAWI traffic is publicly available in the form of a database named MAWILab[122]. This database assists researchers in measuring the detection rate of their anomaly detector. The results of the emerging detectors can be accurately compared to the labels of MAWILab by using a similarity estimator like the one presented in this work.

4.5 Discussion and future work

In addition to its accurate detection, the proposed method has several advantages that are presented in this section.

The graph-based similarity estimator proposed in Section 4.3.1 is a valuable support for systematically benchmarking a detector against other detectors that report traffic at a different granularity. Indeed, by clustering diverse detectors alarms into communities, it allows the automated inspection of numerous detectors outputs in a rigorous manner.

Also, the community rules obtained from the rule mining algorithm consist of concise descriptions of the traffic identified by the numerous alarms being merged into the communities. Therefore, an anomalous traffic reported by numerous similar alarms is annotated with a single label. Thus, the number of labels assigned to the MAWI archive is significantly inferior to the number of alarms reported by the four detectors, and the labels are intelligible to humans.

Following the expansion of the MAWI archive, MAWILab is updated daily to track the latest

trends in Internet traffic and upcoming anomalies. Furthermore, we will also take into account the results from emerging anomaly detectors, to improve the quality and variety of the labels over time. Indeed, by including new results from upcoming detectors the overlaps of the detectors outputs are emphasized and the accuracy of SCANN is improved. Therefore, MAWILab is constantly enhanced and represents a reference data set over time. In order to ease the evolution of MAWILab, we are planning to establish a collaborative system allowing researchers to easily contribute by submitting their anomaly detector or results.

We emphasize that our implementation has the advantage of handling manual annotations or annotations from traffic classifiers[99]. Indeed, the similarity estimator is able to deal with any traffic annotations[54] containing at least two timestamps and one traffic feature. This significant ability of our approach allows us to label traffic with an exhaustive taxonomy. For instance, by adding in the method input the annotations from a traffic classifier, our similarity estimator aggregates similar alarms and corresponding annotations in the same community. Afterwards, the combiner classifies the communities by ignoring the annotations, but the accepted communities are still reported with the extra information provided by the annotation.

The goal of this work is to locate traffic anomalies off-line, so we assume no constraint is placed on the execution time of our approach. Nevertheless, our experiments revealed that the current implementation requires only a few minutes to combine alarms with a 15-minute traffic trace, thus enabling for a real time analysis. However, the study of concurrently running anomaly detectors in real time is left for future work.

Furthermore, we are also interested in studying the sensitivities of the anomaly detectors to estimate the best candidates to combine and to evaluate the combination strategies based on detector selection.

4.6 Conclusions

We proposed a methodology that locates network traffic anomalies in the MAWI archive by comparing and combining the results from four anomaly detectors. Our approach consists of two main steps; first, a graph-based similarity estimator systematically uncovers the relations between the alarms reported by the detectors, second, a combiner classifies the similar alarms using a combination strategy. We evaluated the effectiveness of both steps using different traffic aggregations and combination strategies. Our experiments emphasized the benefit of combining detectors with SCANN, a strategy based on dimensionality reduction, as it ignored irrelevant alarms and detected twice more anomalous traffic than the more accurate combined detector.

The established methodology allows us to accurately detect anomalies in the MAWI archive and precisely assign concise labels. Our results are updated daily using the MAWI archive and are publicly available[122] to assist researchers in benchmarking their detectors. We encourage researchers to contribute to our system by submitting to us their results or detectors, so we can maintain a reliable labeling of the MAWI archive.

第 5 章　おわりに

インターネットの研究において、計測はますます重要視されてきていて、国際協調の機会も増している。そのような状況のなかで、WIDE の計測活動は、グローバルな視点を持った継続的な計測活動として国際的にも認知されてきている。2011 年度は、国際協調をさらに進め、より実りある研究に結びつける事を目標に置いている。また、計測データ保存用の分散ストレージの構築や、クラウド技術を用いた大規模データ解析にも取り組む予定である。