SAND2019-12948C

Time-series Data Analysis for Classification of Noisy and Incomplete Internet-of-things Datasets

Candace Diaz
Sandia National Labortories
Department of Energy
cpdiaz@sandia.gov

Michael S. Postol
U.S. Department of Defense
msposto@radium.ncsc.mil

Robert Simon, Drew Wicke
Department of Computer Science
George Mason University
Fairfax, Virginia
simon@gmu.edu, dwicke@masonlive.gmu.edu

***Abstract*—Detecting and classifying device types and their long term communication patterns and anomalies in massive, noisy and anonymized Internet-of-things (IoT) data sets is a challenging problem. Recent advances in computational approaches for Topological Data Analysis (TDA), including the technique of persistence homology, appear to offer tremendous possibles for understanding highly complex IoT data sets. This paper presents the results of our use of TDA to understand a data set captured over 9 months of hundreds of interacting IoT devices situated in multiple residential settings. The data set is noisy, incomplete and subject to multiple Pattern-of-Life (PoL) fluctuations. We treated the data set as a collection of multi-attribute time series and performed several types of IoT classification experiments. We compared our results to other single and multi-attribute techniques for time series analysis. The outcome was that, as compared to these other standard methods, TDA does particularly well for classifying incomplete, noisy and PoL dependent IoT data.**

***Index Terms*—Topological Data Analysis, Time Series, Internet of Things, Persistent Homology**

I. Introduction

By the end of 2018 spending on Internet-of-things (IoT) devices was expected to exceed $772 Billion [1]. Over the next few years predictions call for annual increases of 20% to 28% in the number of installed devices for the connected home, work, city, health and vehicular sectors [2]. The pace and size of these deployments raises unprecedented challenges in the areas of device and network management and system security. From a management perspective users and organizations will become increasingly hard-pressed to know which IoT devices they own or over a long period of time what constitutes normal and correctly functioning behavior. Closely related are security issues: IoT devices are often left unpatched, vendors use multiple and occasionally non-interoperable standards and protocols, and IoT Information Assurance practices are at a very early state. Finally, there are a slew of privacy issues. Large scale municipalities or service providers need to have an accurate picture of the number and types of IoT devices they service. At the same time, user information associated with specific IoT devices will be encrypted, so system information must be obtained by observing meta data as it flows over public networks.

This paper addresses some of the above challenges by using machine-learning time-series analysis to perform automatic IoT device classification based solely on network traffic patterns. We also provide a discussion of the mathematical underpinnings of our TDA, and provide an explanation for our parameter selection used in our analysis. Using a software package we developed called the Time Series Analysis Tool (TSAT), IoT network activity is represented using the symbolic approximation methods SAX and SFA. TSAT transforms the network activity of each IoT device to strings consisting of symbols from a small alphabet. The resulting strings are then analyzed using several different data mining methods, including a grammar based approach and topological data analysis (TDA). The goal is to on a per device basis identify its function and characterize normal behavior.

We analyzed a dataset consisting of traffic from 183 devices collected over a 9 month period. These devices we deployed in several configurations representing typical home and office settings. The data set was quite noisy and exhibited a wide range of Pattern-of-Life behaviors. For instance, communication patterns for cameras and environmental sensors is very different at 2am on a Sunday evening versus a normal Tuesday morning workday. Further, as might be expected in a realistic settings, the collection activities were sometimes either incomplete or simply failed. Finally, we only looked at metadata such as packet size or interarrival time. This is important since future IoT security and management techniques must deal with issues of privacy or payload encryption.

To our knowledge we are the first group that has used time series analyze for large scale, noisy and realized IoT dataset. Our results indicate that with the appropriate choice of symbolic approximation reduction and machine learning algorithms it is possible to achieve high levels of correct IoT classification based solely on network behavior. For instance, while TDA methods preformed quite poorly when the dataset consisted of traffic from a few days, it achieved surprising levels of classification accuracy over the entire 9 month period. This enables answers to managerial questions such as how many and what types of IoT devices do I have? It can also address security concerns such as being able to automatically identify anomalous behavior.

The paper proceeds as follows: Section II provides some background on IoT and time series machine learning, including topological data analysis We also discuss related work. Section III discusses our approach to time series IoT classification. Section III-A provides details about testbed, while Section IV

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

discusses our experimental results. Finally, V offers a summary and some observations.

II. Background and Related Work

This section provides some basic background on time series data mining, TDA and IoT analysis. We also discuss some work related to our research.

A. Time Series Data Mining Tools

Over the last few decades time series approaches to solve data mining problems, such as classification and anomaly detection, have been very popular. Collected communication network packet data forms a natural time series. As a result, researchers have used time series analysis for problems such as traffic classification [3] and DDoS detection [4]. One advantage of using time series for computer network analysis is the availability of a large number of symbolic representations and similarity measures. These can be used to achieve significant computational advantages. In our work we primarily use the highly popular SAX representation [5], which reduces each event into a small number of symbols. These symbol strings can then be used for classification via supervised learning techniques, and for anomaly detection [6], [7].

The TSAT tool, a timeseries analysis and mining tool being actively developed within the authors' purview - represents an effort to consolidate extant capabilities in modern experimental time series mining methodologies, such as detailed above. Although some of the techniques, such as SAX, do have interpretations which are reconciliable with modern statistical practice - some of the techniques are experimental, and in some cases the results obtained themselves motivate an inquiry into establishing a thesis for the cause of such fitness to a particular application.

One particular innovation we take time to cover in more detail than the more well known time series analysis methods detailed above is the application of a system of analysis methods, topological data analysis (TDA), in order to treat a multivariate time series of signaling characteristics, for each IoT device in the experiment, to derive a representative univariate time series which, for example, provides an unreasonably effective method for predicting indicators and warning (IW) signals of historical stock market crashes [8]. The technique is largely non-parametric, with the exception of choosing characteristic time and length scales to describe the finest and coarsest bounds to consider for describing the multitudes of proximal-connective associations discovered between collections of measurements of the multivariate samples for each device, over time. This explains the rationale for developing the technique in some detail in the proceeding section.

B. Topological Data Analysis for Time Series

We start by an appeal to intuition, since the mathematical foundation of TDA requires a level of development we cannot provide here. However, we make an effort to provide sufficient details for the mathematically sophisticated reader in order to reveal the essential structure of the problem and computations, and provide some examples to help build intuition for the technique. For a comprehensive foundation to the field of algebraic topology, we refer the reader to [9], [10]. In terms of comprehending the performance results, the reader will lose nothing by proceeding immediately to Section III after the following preamble.

Recent work in TDA for data mining and analysis of complex systems includes [8], [11], [12]. The work in [12], for instance, demonstrates an invariable increase in classification accuracy, over one of the state-of-the art techniques, for a human activity recognition problem when TDA is used for feature engineering. The work in [13] examined the entire TDA processing pipeline from the point of view of applying persistent homology, also to time-delay embeddings of univariate time series, similar to [12]; however the precise features formed from the topological information is slightly different. They also described a new data set for TDA analysis called TS-TOP. The technique used in the IoT experimental analysis instead involves the application of a rolling time window of a multivariate time series for each IoT device. The window is then used to construct an associated abstract simplicial complex, the fundamental data structure used to compute topological shape numbers (invariants) of the complex, describing cavities of various dimensions in the complex. We use the persistence landscape functional [11] to assess the change in morphology of the complex over time, as did [8]. Theory [14] shows that this functional is stable under small perturbations of the data. For volatile or chaotic time series - such as stock prices, EEG (human brainwaves), or turbulent fluid flow, the particular sequence of measurements taken over time is far less important to holistically understanding such a complex system than the overall shape of the attractor - a shape upon which the sequence of measurements tend to asymptotically, regardless of certain modalities peculiar to the system's normal fluctuations and reactions to stimuli in the short term. In our technique, we use TDA to estimate the long-term change of shape-connectivity characteristics of the various attractors describing the time series' of device signaling statistics, per device, and classified devices into related collections based on a univariate statistic describing the changes in the shape information over a long time scale (e.g. that of the experiment).

1) Homology for Simplicial Complexes : For complete rigor, we refer the reader to [10] for the prerequisite background. However, we do give sufficient details here to satiate anyone with a standard background in abstract algebra and set theory to largely understand the technique at hand.

A collection of sets X is an abstract **simplicial complex** if $\forall U \in X,\, V \subset U \Rightarrow V \in X$; in particular, if $V, W \in X$, then we require $V \cap W \in X$. The sets of X are called simplices, or faces of X. Any simplex contains $\varnothing$ as a face, and $\{\varnothing\}$ as well as $\varnothing$ are vacuously simplicial complexes of one and zero faces, respectively. The **dimension** of an individual simplex $U \in X$ is given by $\dim U = |U| - 1$. If the cardinality of X is finite, then X is a finite simplicial complex, and is finite dimensional if the maximal dimension of its faces is. Let $X^0 = \{x_0, \ldots x_N\}$ denote the vertices, or 0-dimensional faces of X;

it is common to abuse notation and consider $x_i \sim \{x_i\}$ for this purpose. The k**-skeleton** of X is determined as $X^k = \binom{X^0}{k+1} = \left\{\sigma \subset X^0 : |\sigma| = k+1\right\}$; e.g., the k-dimensional faces of X.

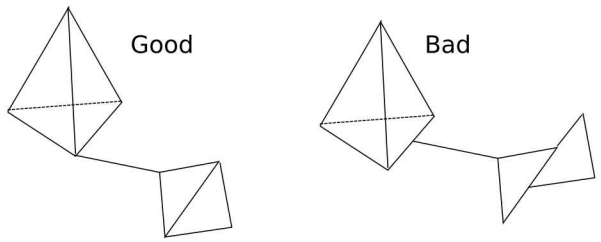

Fig. 1: Candidate simplicial complexes, of dimension ≤ 3.

Various constructions aide in determining the abstract simplicial complex from a metric space structure. The one we use is the **Vietoris-Rips** complex, illustrated in figure 2. Analogous to an MRI scan, it gives an adjustable parameter to explore multiple resolutions of the data. Given a set $X^0 \subset \mathbb{R}^n$ of points, let a parameter $r \in [0, \infty)$ be given. Denote by, for $x \in \mathbb{R}^n$, $B_r(x) = \{y \in \mathbb{R}^n \ni d(x,y) \leq r\}$ the ball of radius r about x, with respect to some metric $d\colon \mathbb{R}^n \times \mathbb{R}^n \to [0, \infty)$. In our work, we use the euclidean metric. The Vietoris-Rips, or simply Rips, complex $R_r(X^0)$ is then given as follows: $\sigma \subset X^0$ is a simplex of $R_r(X^0)$ if and only if $d(x,y) \leq r$, $\forall x, y \in \sigma$. Among other constructions, the Rips complex is convenient to calculate; but potentially introduces high dimensional artifacts. Since our focus will be on dimensions 2 and lower, this is of ancillary concern.

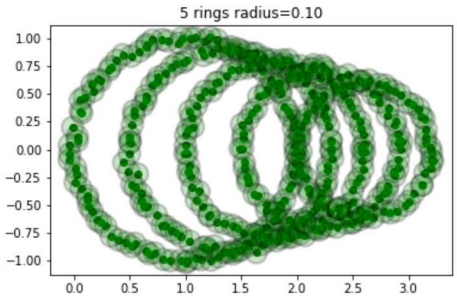

Fig. 2: Vietoris-Rips complex for fixed r.

Again, suppose that X is a complex with vertices $X^0 = V$. We show how to compute the shape data for the Rips complexes we will fashion from time series. By $C_p(X)$, denote a free abelian group (over $\mathbb{Z}_2$), called the p-dimensional chains; $c \in C_p(X)$ means $c = \sum_{\sigma \in X^p} \beta_\sigma \cdot \sigma$; $\beta_\sigma \in \{0,1\}$ mod 2. Intuitively, the coefficients β_σ simply tell us which faces of X_p are involved in c. The boundary operator $\partial\colon C_p \to C_{p-1}$ maps an individual simplex to the sum of its faces (its boundary in algebraic form) $\partial(\sigma) = \sum_{\tau \in \binom{\sigma}{p}} \tau$, extending ∂ linearly over $\mathbb{Z}_2$ chains yields $\partial\left(\sum_{\sigma \in X^p} \beta_\sigma \cdot \sigma\right) = \sum_{\sigma \in X^p} \beta_\sigma \cdot \partial(\sigma)$. Intuitively, $\partial(c)$ is the algebraic representation of the union of the boundaries of simplices which are represented by $\beta_\sigma = 1$. If a pair of simplices of dimension p share a common face, then their sum in the above will cancel. Now, $\partial\colon C_p(X) \to C_{p-1}(X)$ is a vector space transform (linear operator), so kernels and images make sense. Denote by p-cycles that subgroup $Z_p(X) \subset C_p(X)$ where $\partial(c) = 0$, $\forall c \in Z_p(X)$. These are the chains where the p-dimensional boundary of a simplex involved in c meets with other parts of other boundaries of various other p-chains; e.g., closed loops, bubbles and higher dimensional analogues. Denote by p-boundaries that subgroup $B_p(X) \subset C_p(X)$ such that $\forall c \in B_p(X)$, $\exists \tau \in C_{p+1}(X) \ni c = \partial(\tau)$. Intuitively, these represent the unions of boundaries of one-higher dimensional simplices, with intersecting pairs of faces canceling. Furthermore, if a cycle is not covered by the boundary of a one-higher dimensional set of simplices, then this becomes a representative cycle bounding a $(p+1)$-dimensional cavity in X. Algebraically, we need $B_p(X) \subseteq Z_p(X)$ so to make the quotient space $Z_p(X)/B_p(X)$, whose only non-zero elements the equivalence classes of cycles representing unique cavities of dimension $p+1$ within X; the **homology group** of dimension p. There are possibly many representative cycles for a cavity - a rubber band can loop around a paper towel roll in infinitely many ways, but each configuration can be slipped into the other without cutting the rubber band or turning it into a figure-eight. Suppose that $c \in C_{p+1}(X)$ is the chain consisting of one $(p+1)$-dimensional face with coefficient 1. Then $\partial^2(c) = \partial(\sum_{\tau \in \binom{c}{p}} \tau) = \sum_{\tau \in \binom{c}{p}} \sum_{\sigma \in \binom{\tau}{p-1}} \sigma$. Observe $|c - \sigma| = 2$, in the summand, so $\forall \sigma \in \binom{\tau}{p-1}$, $\exists u, v \in X^0 \ni \sigma \cup \{u, v\} = c$. Therefore, σ occurs twice for that τ, and thus every σ occurs an even number of times in the above sum for $\partial^2(c)$. Observe for an integer k, $(2k) \cdot \sigma = 0$ in $C_p(X)$, since $2k \sim 0 \mod 2$; and therefore $\partial^2(c) = 0$. Since X^p is a basis for $C_p(X)$, $\partial^2 = 0$. $B_p(X) \subseteq Z_p(X)$, and the homology group $H_p(X) = Z_p(X)/B_p(X)$ is well defined.

Now we combine the concepts of the Rips complex and homology to arrive quickly to a notion of persistent homology. For finite $V \subset \mathbb{R}^n$, observe that if $0 \leq r \leq s$ then $R_r(V) \subseteq R_s(V)$. A nested family of complexes $R_{t_0}(V) \subseteq R_{t_1}(V) \subseteq \cdots \subseteq R_{t_m}(V)$ is called a filtration. When $t_0 = 0$, the first complex in the filtration is just the discrete set of points in V. Topologically, the inclusion maps $R_r(V) \overset{\iota_{r,s}}{\hookrightarrow} R_s(V)$, when $r \leq s$, are continuous. [9, p. 111] shows that this induces a homomorphism $H_i(R_r(V)) \overset{H_i(\iota_{r,s})}{\to} H_i(R_s(V))$ in all dimensions i. The dimension of the image of $H_i(\iota_{r,s})$ counts the number of $(i+1)$-dimensional cavities that **persisted** in the Rips radius interval $[r, s]$. The **persistent homology** of these features in the filtration are represented by the images of these induced maps, and form the basis of the features we use for time series. For a non-zero homology class σ in $H_i(R_r(V))$, for any r, define the **birth time** to be $b_\sigma = \min\{r \in [0,\infty) : [\sigma] \in H_i(R_r(V)) \backslash [0]\}$ and the **death time** $d_\sigma = \max\{s \in [0,\infty) \cup \{+\infty\} : H_i \circ \iota_{b_\sigma, s}(\sigma) \neq 0\}$. The **persistence diagram** P_i, of the filtration's ith dimensional homology, is then given by integer points (b_σ, d_σ), along with multiplicity $\mu(b_\sigma, d_\sigma)$ to count how many unique homology classes in dimension i shared (b_σ, d_σ) (c.f. [8]).

We use an alternative representation of a persistence diagram, invented by [11], to capture the topological information as a sequence of functionals, following [8]. Define $f_{(b_\sigma,d_\sigma)}(t)$ to be zero everywhere, except for its linear interpolants by line segments through the points $(0, b_\sigma), (d_\sigma + b_\sigma/2, d_\sigma - b_\sigma/2), (0, d_\sigma)$ The height of this tent function is how "strong" the feature σ is at time t. The longer it lives, the "stronger" the feature. The k**th persistent landscape** of the filtration is $\lambda_k(t) = k\text{-}\max\left\{f_{(b_\sigma,d_\sigma)}(t) : (b_\sigma, d_\sigma) \in P_i\right\}$, where the k-max is the kth largest value. For all the work here, $i = 1$, and so the homology classes we have in mind are 1-dimensional loops related to quasiperiodic behavior in timeseries.

In [11], a norm on the combined functionals $\lambda = \{\lambda_1, \lambda_2, \ldots\}$ is provided in terms of the standard L^p norm $||f||_p = (\int |f|^p d\mu)^{1/p}$ on a single functional. Note that there are only finitely many non-zero λ_k for a finite simplicial complex, since P_1 only then has finitely many points, and we will eventually exhaust P_1 at large enough k. This norm is given as follows, for $1 \leq p \leq \infty$: $||\lambda||_p^p = \sum_{k=1}^{\infty} ||\lambda_k||_p^p$, providing a Banach space structure on P_i, for a given filtration, enabling comparing different filtrations. The following figure shows the top 14 landscapes of a filtration associated with figure 2; note that the 4 largest holes in that figure are represented by the tallest peaks in the graph.

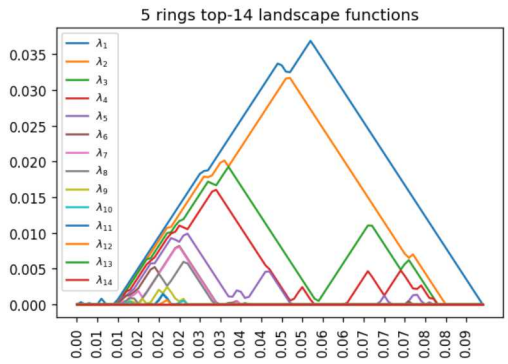

Fig. 3: Top 14 Landscapes for 5 Rings Example.

2) Landscapes of Multivariate Timeseries:

Now we are prepared to present the method we used in practice for detecting the timeseries of landscape norms for each device's multivariate measurements over time. Suppose that we are given a multivariate timeseries $\{\boldsymbol{x}_t \in \mathbb{R}^d : t = 0, 1, 2, \ldots\}$ for $d > 1$. Select a window size parameter w and a step-size $\delta t \in \mathbb{N}$, and also a maximum Rips radius r with which to grow the filtration of Rips complexes from 0 to r for persistent homology. For $i, t \in \mathbb{N}$, define $t_i = w + i \cdot \delta t$, a particle swarm window $Z_{t_i} = \{\boldsymbol{x}_{t_i - w}, \boldsymbol{x}_{t_i+1-w}, \ldots, \boldsymbol{x}_{t_i-1}\}$. We then form a Rips filtration for each time by $R_0(Z_{t_i}) \subseteq R_{r/m}(Z_{t_i}) \subseteq \cdots \subseteq R_r(Z_{t_i})$, where m is implicitly determined by the software. From the filtration of this particle swarm from the timeseries, we form the landscape L^2 norm $||\lambda_{t_i}||_2 = \left(\sum_{k=1}^{\infty} ||\lambda_{t_i,k}||_2^2\right)^{1/2}$, from the landscapes $\lambda_{t_i,k}$ corresponding to the filtration's P_1 diagram. We use Gaussian quadrature to compute the norms of λ_k on a grid size of 100 points by default. The final output of the algorithm is the sequence of values $||\lambda_{t_i}||_2$, $i = 0, 1, 2, \ldots, N$, which measure the changes in morphology of the time series' particle swarm over time. To give a bit of intuition about choosing the window size w and step-size δt, we appeal to an example based on the evolution of the dynamical system as follows: $\boldsymbol{x}_{t+1} = 2\boldsymbol{x}_t - \boldsymbol{x}_{t-1} + \frac{\delta t^2}{2N^2} \cdot (1 + t/N) \cdot \left(\frac{\boldsymbol{l}_t - \boldsymbol{x}_t}{||\boldsymbol{l}_t - \boldsymbol{x}_t||^3} + \frac{\boldsymbol{r}_t - \boldsymbol{x}_t}{||\boldsymbol{r}_t - \boldsymbol{x}_t||^3}\right)$, where N is the size of the time interval sampling. This is the second order evolution equation for a particle moving according to its own inertial forces and attraction to $1/r^2$ forces of equal and slowly growing magnitude at $\boldsymbol{l}_t = (-c_t, 0)$ and $\boldsymbol{r}_t = (c_t, 0)$, so that the overall centroid of the pair is at (0,0). Initially, $c_t = 0$, and we start the initial conditions $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ so that we have a particle first orbiting in a perfect circle around the initial centroid. For different methods of centroid control, note that the TDA method is able to detect the structural change.

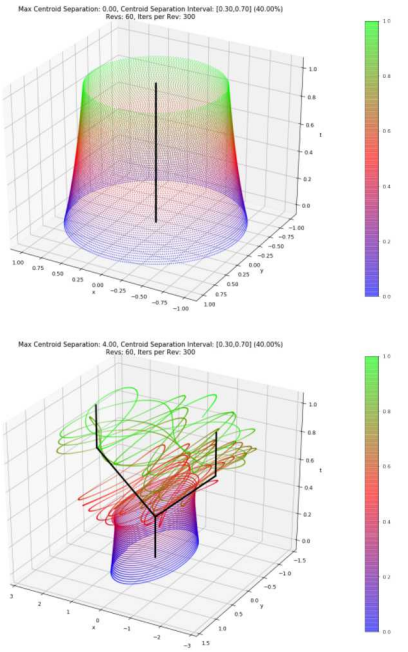

Fig. 4: Evolution of diverging attractors. Black trace indicates position of centroids.

Intuitively, each of the circular orbits near $t = 0$ generally have about 300 samples per period, at the beginning. The structure changes during $t \in [0.3, 0.7]$, which is a much longer time scale than an individual orbit (there would be about 60 orbits if the centroids indicated in black did not diverge), since the change thus occurs within about 24 orbits. Therefore, $w = 600$ gives is a large enough particle swarm to, on average, cover about two orbits' worth of data, but not

so large as to lose the 24 orbits, or so, of change information by absorbing too much information from the stable states near $t = 0, 1$. The change in the first figure's topology is seen to be zero, and the second figures is significant. For the step-size δt, we chose $\delta t = 25$ samples, since shifting the particle swarm of 600 points by that much causes the majority of points between consecutive particle swarms to be coincident, and therefore, below the characteristic time scale of the known modality shifts - and this speeds the calculation up by a factor of about 25.

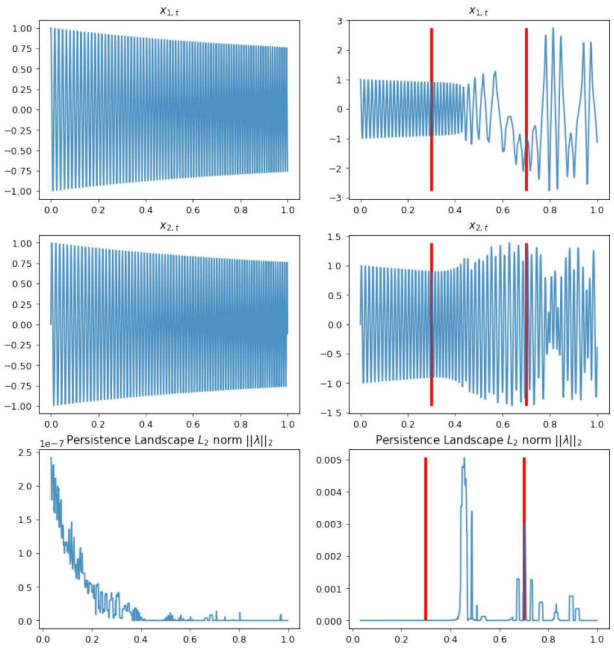

Fig. 5: TDA assessment of modality change for experiment in figure 4. Left is where there was no divergence of the attracting nodes and right is where there was divergence.

Clearly, there is a predictive signal in the change period with the TDA landscape norm series, indicated on both sides by the red vertical lines in the plot, for when the point attracting nodes diverge, and the orbits become more volatile. For the non-diverging attractor on the left, all the TDA landscape norms are at or below the threshold for floating point error within 0 - indicating essentially no change in morphology is being measured.

In practice, 40 hours (1 work week) was discovered to be the best size for the window w in the IoT data by a grid search, using classifier accuracy for devices as a proxy for fitness. Reverse engineering this through the example logic, one possible interpretation divined from the technique is that significant configuration and modality change in any particular class of IoT device happens over a characteristic time scale longer than a work week, which is an interesting observation. The step size was $\delta t = 1$, is prudent when the number of samples required to describe the shape of the overall attractor modality is unknown, at cost of extra computation.

3) TDA Software: [15] provided a very efficient method for computing persistent homology, a huge leap forward for progress in the field of computational topology. Dmitriy Morozov's Dionysus2 software ([16]) was used to perform the Rips and persistence computations in practice, and we thank him for the vigorous discussions. Our software computes the landscape functions, their norms, and handles the process of deriving the particle swarms from the time series and computing the persistent homology in parallel using Python.

C. IoT Analysis

Techniques for IoT device discovery include banner grabbing methods. The approach is to mine textual information, typically obtained at the application layer, for labels identifying IoT devices. One example is the work presented in [17] which used Nmap banner rules for device analysis.In [18]the authors propose an Acquisitional Rule-based Engine to automatically generate rules for IoT device discovery. The work presented in [19] uses multiple classifiers to distinguish between IoT and Non-IoT traffic. The authors looked at 9 IoT and 4 non-IoT devices, and based upon TCP sessions were able to accurately distinguish the two classes of devices.

Each captured IoT packet includes a large number of attributes, both directly observable and calculated. This means that IoT packet traces are multi-attribute time series. Examples of directly observable attributes include packet arrival time, packet length, and source and destination MAC address. Notice that these attributes are obtainable even in the case of encrypted traffic. Important examples of calculated data include the mean and variance of packet sizes or interarrival times for a particular "conversation," between a pair of MAC address. Data mining for multi-attribute time series is still an active research area [20]. One of the focuses of this paper is to understand how TDA can be used for multi-attribute IoT analysis.

III. IoT Analysis Testbed and Methodology

This section describes our experimental setup, our processing pipeline and the software we developed for this project.

A. IoT Testbed Setup

Figure 6 shows the proprietary testbed we for used our analysis. As can be seen, the IoT devices connected to each using a variety of wireless and wired technologies. Collectively, the devices communicated to the Internet and back-end cloud services provides through a gateway. The testbed consisted of multiple rooms and hall ways, designed to emulate how consumers would use IoT devices in residential and commercial areas. The rooms also had desks and workstations, and were used by workers associated with the project and other workers who were not. The purpose was to observe and experiment with IoT systems undergoing a normal pattern of life as may be seen in a typical office environment.

The engineers experimenting with the IoT testbed had workstations within the IoT enclave, and had access both to wired and wireless transmissions. Our packet traces were

Device Type	Number of Devices
Audio Speaker	8
Video Cameras	34
Doorbells	5
Fitbits	2
Game Controllers	1
IoT Hubs	39
Lights	13
Miscellaneous	8
3-D printer	1
Romba	1
Router	1
Environmental Sensors	23
Switches	5
Tablets	2
TVs	21
TV dongles	16
Weather Stations	3

TABLE I: Number of deployed IoT devices

obtained from a server that captured traffic between the IoT enclave the outside world. Thus, our experimental IoT data set consisted of traffic either being uploaded or downloaded to the cloud. Typical Internet destinations consisted of backend IoT cloud storage devices, media streaming services, etc.

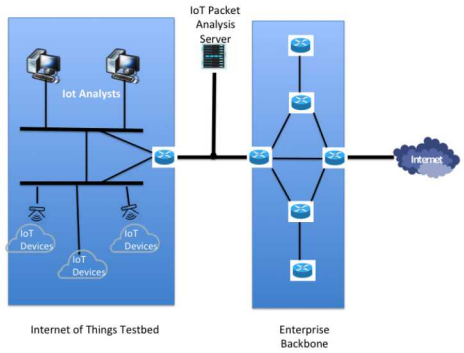

Fig. 6: Testbed

The lab contained a total of 183 IoT devices from a wide variety of manufacturers. Table I shows the number of different device types.

B. Experimental Approach and Processing Pipeline

All network traffic over a 9 month time period was collected. Each received packet was time-stamped. For the logs we analyzed, each time-stamped entry had source/destination MAC and IP addresses, and transport layer ports along with the device type for the payload. Notice that the transmitting device may or may not be the device that originated the payload. This enabled is to build a labeled training set. The captured log files enabled us to compute a number of derived characteristics, including, for a MAC or IP conversation, the total number of bytes sent over a time window, the mean and the variance of the interarrival time. The results of our experiments are presented in Section IV. As will be seen, we experimented with both single and multi-attribute time series.

In the case of a single attribute time series (regardless of the semantic meaning of the attributed) the data was converted into a SAX symbol. Labeled time series device files were then used to train a classifier. We used a variation of Representative Pattern Matching, essentially a string based classification method using clustering [6].

For multi-attribute time series we used both observed and derived attributes. We used two different approaches. The first was topological data analysis using the Dionysus package [16]. The second was WEASEL+MUSE [20]. The output of this step was a single attribute time series, which we then analyzed using the above method.

We have implemented an integrated analysis tool called TSAT, the Time Series Analysis. Training data is automatically cross validated, and F1 and MCC scores are displayed after testing.

IV. Evaluation

The goal of this work was to determine the effectiveness of device classification in the presence of noisy data, as this would represent the problems faced by managers and regulators of large scale IoT installations. We divided the IoT device world into one of three classes. The first were cameras, to include audio speakers, characterized by high-volume and burst data. Some cameras continuously broadcast and some are event activated, like as motion-trigged cameras and on-demand audio speakers. The second were sensors, such as environmental monitors. The third were called multi-purpose devices, such as tablets or certain cameras that allowed for two-way and streaming audio. Thus, each one of the devices shown in Table I were divided into one of three classes. Note that, depending on device type, some of the Video Cameras were mapped into the category ”camera” and some were mapped into the category ”multi-purpose.” Our goal was not to preprocess testing data, in the sense that all log files were analyzed.

A. Pattern of Life

Our analysis of the entire data set showed it to be noisy, in the sense of incomplete and inconsistent, for several reasons. The first was that due to factors such as system maintenance or equipment upgrade either no collections were performed or collections were incomplete. The second was due to device Pattern-of-Life (PoL). It was to be expected that usage for some types of IoT devices are impacted by the differences between an average work day and Thanksgiving break, but even within the same day there were significant variations.

Figures 7 and 8 illustrates this phenomenon. Figure 7 shows the number of packets transmitted over one hour between 12:11 and 13:11 pm on a workday. We captured a total of 349,677 packets from 161 network interfaces. Figure 8 shows the number of packets transmitted between 18:11 and 19:11 on the same day. A total of 67, 610 packets were transmitted from 142 network interfaces. This PoL repeated itself throughout the entire data set. We speculate that this type of noisy data

set well characterizes what data would look like in large scale IoT installations.

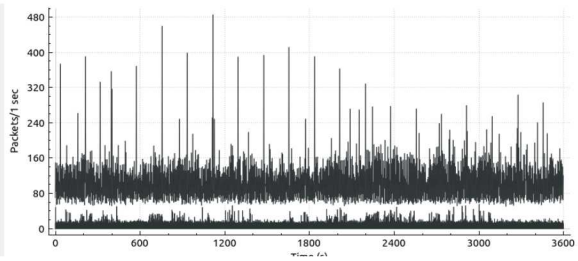

Fig. 7: Number of packets transmitted

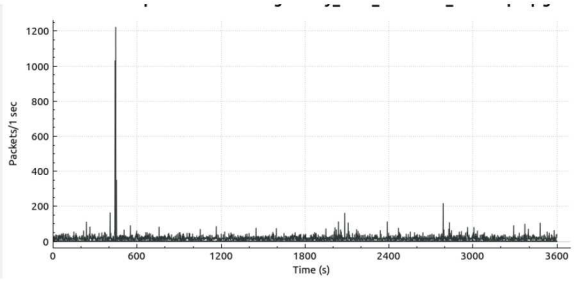

Fig. 8: Afterwork

B. Single Attribute Selection

As with all data mining problems attribute selection is a fundamental concern. We were interested in experimenting with both single attribute and multiple attributed time series. Each time series was a device "conversation", as identified by a transport-level endpoints. For single attributed time series we discovered that the best results were obtained by looking at the time number of bytes sent over a fixed time window. We found that setting the time window to one hour worked well. The next best single attribute was packet interarrival time.

For single attribute non-TDA classification, we tended to get the best results with the training dataset consisting of one day and the testing data set also for a single day. . Our initial assumption was that we would get the best results by normalizing the data in the form of min-scaling. Each data point X_{sc} is calculated as

$$X_{sc} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

where X is the entry in the time series, X_{min} is the minimum value in the time series and X_{max} is the maximum value of the time series. Table II shows the F1 per device class and the weighted score.

We then eliminated min-scaling and found, under most single day experiments, that we actually obtained better results. For instance, Table III shows the results for the same training and testing data set shown in Table II without min-scaling. Here we present both the F1 and MCC score per device class along with the overall weighted F1 and MCC score. Both

TABLE II: Univariate Time Series – Bytes Sent over a Single Day with MinScaling

Device Type	F1
Cameras	.567
Sensors	.83
Multi-purpose	.59
Weighted	.68

TABLE III: Univariate Time Series – Bytes Sent over a Single Day Without MinScaling

Device Type	F1
Cameras	.87
Sensors	.811
Multi-purpose	.789
Weighted	.828

Device Type	MCC
Cameras	.79
Sensors	.72
Multi-purpose	.72
Weighted	.75

experiments used the SAX data representation and the TSAT tool.

These results are relatively robust to Pattern-of-Life. For instance, for the data shown in the above tables the training day was a nationally observed holiday while the testing day was a routine work day.

C. Multiple Attribute Selection

We then explored the use of multiple attributes. Here we present the results from using the WEASEL+MUSE multi-attribute approach to TDA. WEASEL+MUSE uses feature extraction via Bag of Patterns, performs a statistical feature reduction and then does logistic regression for classification. We tried both the SAX data representation and Symbolic Fourier Approximation (SFA) data representation. For single day training and testing we found minor differences with the single attribute approach. For instance, for the data set presented in Tables II and III we achieved a weighted F1 score of .83.

When we examined either longer training or testing cases the above approaches rapidly collapsed, with MCC scores approaching 0. For this reason we experimented with TDA. We found that we required at least one month for either training or testing to get meaningful results. Our approach was to first make the time series stationary by using the log first order difference. We then passed it into the Dionysus library function, which outputs a univariate time series. For brevity we show the results with a sliding window of size 40, with no skipped samples, and using a L^2 norm as the distance function. Further, we found that, unlike for the results shown in Section IV-B, min-scaling outperformed no-min-scaling for the single attribute.

Table IV show the weighted F1 results for a variety of experiments with a one month training and a one month testing data set. As can be seen, some attribute combinations were no better than random while others were slightly better. TDA outperformed the WEASEL+MUSE approach when using the attributes of Experiment 1, with a weighted F1 of .28.

Using the same training data we then increased the size of the test data, and found a steady improvement. Figure V shows

TABLE IV: Multiple Attribute Selection for TDA

Experiment	Attributes	Weighted F1
1	TCP+UDP combined to create average throughput (AVG) and Interarrival time (IAT) time series	.51
2	TCP+UDP combined to create total throughput (TOT) and IAT time series	.31
3	TCP+UDP combined to create max throughput (MAX) and IAT time series	.41
4	TCP+UDP combined to create MAX, TOT, AVG, and IAT time series	.41
5	TCP only to create MAX, TOT, AVG, and IAT time series	.49
6	MAX of send and recv streams for UDP and MAX for recv for TCP along with TCP only to create MAX, TOT, AVG, and IAT time series.	.52

TABLE V: Topological Data Analysis - 8 months

Device Type	F1
Cameras	.687
Sensors	.835
Multi-purpose	.8
Weighted	.77

Device Type	MCC
Cameras	.535
Sensors	.765
Multi-purpose	.69
Weighted	.66

the results for one month training and 8 months testing for the attribute combination for Experiment 6. Given the amount of noise in the large dataset, we considered these results to be quite good. Our conclusion was that, for relatively small realistic IoT data sets, traditional time series approaches can work fine, and TDA requires a minimum amount of data to begin to work. However, given a large amount of data (multiple millions of packets) traditional methods do not work, and TDA performs quite well.

V. Conclusions

This paper described our time series analysis of a 9 month data set of network traffic sent and received by 183 IoT devices. The focus of our efforts was to perform IoT type classification. We first provided some basic background in time series analysis and the need to classify IoT traffic. We also provided a mathematical background for TDA and discussed how we chose our parameters for evaluation. Our results show that, while traditional time series techniques perform well for shorter periods of time, over a multi-month period TDA outperforms those other methods.

References

[1] BusinessWire, "Idc forecasts worldwide spending on the internet of things to reach $772 billion in 2018," 2017. [Online]. Available: https://www.businesswire.com/news/home/20171207005963/en/IDC-Forecasts-Worldwide-Spending-Internet-Things-Reach

[2] ZDNet, "Iot to drive growth in connected devices through 2022: Cisco," 2017. [Online]. Available: https://www.zdnet.com/article/iot-to-drive-growth-in-connected-devices-through-2022-cisco/

[3] Y. Cai, H. Tong, W. Fan, and P. Ji, *Fast Mining of a Network of Coevolving Time Series*, pp. 298–306. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/1.9781611974010.34

[4] S. M. Tabatabaie Nezhad, M. Nazari, and E. A. Gharavol, "A novel dos and ddos attacks detection algorithm using arima time series model and chaotic system in computer networks," *IEEE Communications Letters*, vol. 20, no. 4, pp. 700–703, April 2016.

[5] E. J. Keogh, J. Lin, and A. W. Fu, "HOT SAX: efficiently finding the most unusual time series subsequence," in *Proceedings of the 5th IEEE International Conference on Data Mining (ICDM 2005), 27-30 November 2005, Houston, Texas, USA*, 2005, pp. 226–233. [Online]. Available: https://doi.org/10.1109/ICDM.2005.79

[6] X. W. et al, "RPM: representative pattern mining for efficient time series classification," in *Proceedings of the 19th International Conference on Extending Database Technology, EDBT 2016, Bordeaux, France, March 15-16, 2016, Bordeaux, France, March 15-16, 2016.*, 2016, pp. 185–196.

[7] P. S. et al, "Grammarviz 2.0: A tool for grammar-based pattern discovery in time series," in *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2014, Nancy, France, September 15-19, 2014. Proceedings, Part III*, 2014, pp. 468–472.

[8] M. Gidea and Y. Katz, "Topological data analysis of financial time series: Landscapes of crashes," *Physica A: Statistical Mechanics and its Applications*, vol. 491, pp. 820 – 834, 2018.

[9] A. Hatcher, *Algebraic topology*. Cambridge: Cambridge University Press, 2002.

[10] E. H. Spanier, *Algebraic topology*. Springer Science & Business Media, 1989, vol. 55, no. 1.

[11] P. Bubenik, "Statistical topological data analysis using persistence landscapes," *Journal of Machine Learning Research*, vol. 16, pp. 77–102, 2015.

[12] Y. Umeda, "Time Series Classification via Topological Data Analysis," *Transactions of the Japanese Society for Artificial Intelligence*, vol. 32, no. 3, pp. D–G72, Jan 2017.

[13] L. M. Seversky, S. Davis, and M. Berger, "On time-series topological data analysis: New data and opportunities," in *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, June 2016, pp. 1014–1022.

[14] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer, "Stability of persistence diagrams," *Discrete & Computational Geometry*, vol. 37, no. 1, pp. 103–120, Jan 2007.

[15] A. Zomorodian and G. Carlsson, "Computing persistent homology," *Discrete & Computational Geometry*, vol. 33, no. 2, pp. 249–274, 2005.

[16] D. Morozov, "Dionysus," *Software available at http://www. mrzv. org/software/dionysus*, 2012.

[17] M. A. et al, "Understanding the mirai botnet," in *26th USENIX Security Symposium (USENIX Security 17)*. Vancouver, BC: USENIX Association, 2017, pp. 1093–1110.

[18] X. Feng, Q. Li, H. Wang, and L. Sun, "Acquisitional rule-based engine for discovering internet-of-things devices," in *27th USENIX Security Symposium (USENIX Security 18)*. Baltimore, MD: USENIX Association, 2018, pp. 327–341.

[19] Y. M. et al, "Profiliot: A machine learning approach for iot device identification based on network traffic analysis," in *Proceedings of the Symposium on Applied Computing*, ser. SAC '17. New York, NY, USA: ACM, 2017, pp. 506–509.

[20] P. Schäfer and U. Leser, "Multivariate time series classification with WEASEL+MUSE," *CoRR*, vol. abs/1711.11343, 2017.

Acknowledgement

This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government. Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energys National Nuclear Security Administration under contract DE-NA0003525. None of the work in this document was prepared with funding from a Sandia National Laboratories program.