Available online www.jsaer.com

***Journal of Scientific and Engineering Research*, 2024, 11(1):290-299**

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

High Significant Anamoly Detection Analysis by Using Azure Core Workload Insights

Kartheek Pamarthi

Email ID: Kartheek.pamarthi@gmail.com

Abstract Workload insights offered by Azure Core include time-series data with many metric units. There are several issues with the time-series data that can be explained by mistakes in the metric name, resource region, dimensions, and the associated dimension value. One of the most critical tasks for Azure Core is to provide the user with a dashboard that allows them to easily see any errors or irregularities that need to be addressed. The number of anomalies that are reported has to be quite substantial and in a limited quantity; for example, an hourly rate of five to twenty anomalies should be recorded. In any time-series forecasting model, the anomalies that have been reported will have a major impact on user perception and a high coefficient of reconstruction error. Therefore, it is our responsibility to automatically recognise "high significant anomalies" and the information that is connected with them for the purpose of user perception.

Keywords Significant Anamoly detection, Azure core, Insights.

Introduction

Microsoft has a suite of artificial intelligence-powered tools and services that are housed on the Azure cloud platform. One of these tools is called Azure Anomaly Detector. For the purpose of analysing time-series data and locating abnormalities, deviations, or outliers within the data, it makes use of sophisticated machine learning methods. These irregularities may be indicative of unique patterns, unexpected spikes or declines, or deviations from the norm, all of which may point to the existence of possible problems, fraudulent activity, or development of new trends. The capability of Azure Anomaly Detector to automatically adjust to various data patterns and circumstances is one of its most important characteristics. This capacity enables it to be extremely versatile and appropriate for a wide variety of situations and applications. In order to continually learn from data streams and increase the accuracy of anomaly detection over time, it makes use of supervised and unsupervised learning, statistical modelling, and probabilistic algorithms.

Multiple customer escalations occurred as a result of sluggish root cause analysis (RCA) because there was no automated mechanism in place to make recommendations for speed optimisation. For the purpose of addressing the pain spots that fall under three different categories of pain points, we have provided a solution. Customers are able to check for problems within their workload and receive ideas on how to tackle those problems in a single location. As a component of the comprehensive solution, we are actively working to ensure that the workload of our customers remains the primary emphasis. Our proposed feature is the ability to detect unusually high or low workloads that can be generated by Azure services or components. Figure 1 summarises the high- and low-level significant anomalies discovered in a time series sample extracted from Azure Core workload data. The red-outlined high significant anomalies are very rare occurrences in the data, easily identifiable by users, and expected to cause a large reconstruction error in time-series forecasting models. Frequent events with low significant anomalies have a low reconstruction error in time-series forecasting models, are difficult to detect by users, and occur frequently overall. These anomalies are highlighted in green.

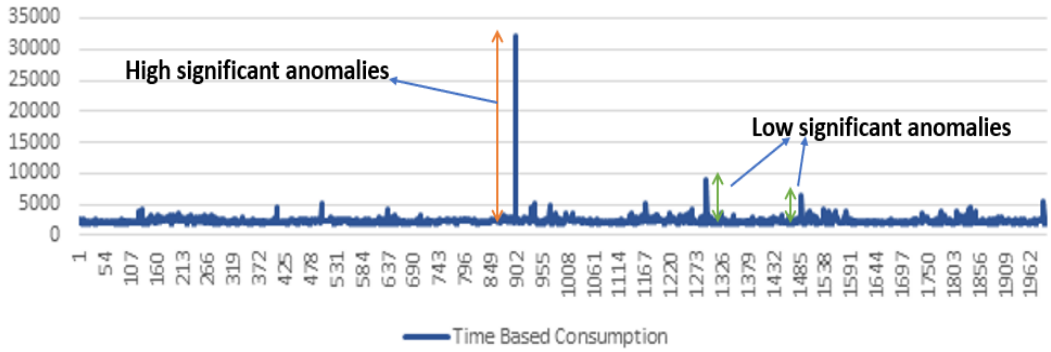

Figure 1: A time series graph displaying outliers with high and low significance levels

A manual filtering method is currently being used in the solution being deployed by the Azure Core team. This method produces a large amount of fault points and warnings, with abnormalities being discovered at a rate of thousands per hour. There are two steps to the proposed procedure that will lead to the desired outcome:
The first stage is a private preview for the company's employees, while the second stage is for actual customers.
Stage 1: The overall number of anomalies discovered every hour should be less than 150, and there should be a decrease in the number of false positives while keeping the ability to recognise true positives.
Stage 2: In human-in-the-loop validation, the only anomalies that are identified are those that are highly significant and human perceivable. These anomalies are those that have a high reconstruction error between the actual and projected values. As a result, the overall number of anomalies that are discovered should be somewhere between 5 and 20 every hour.
Methods like WaveNet (Oord et al. 2016), Temporal Fusion Transformers (TFT) (Lim et al. 2021) and DeepAR (Salinas et al. 2020) can be used to achieve the goals of Stage 1.
On the other hand, in order to go to Stage 2, we need to identify anomalies that are really significant and get rid of corner case anomalies in which the loss of reconstruction is insignificant. Although previous methods have hit a 'glass barrier' at about the 99.998% quantile, others, such Extreme Value Theory (EVT) (Siffer et al. 2017) [4], can still perform filtration in this high quantile range. When the hyper-parameter risk factor/quantile is fine-tuned, we may spot very unusual things.
These are the primary contributions that we have made to the paper, which are listed below: Using Extreme Value Theory atop time-series forecasting models, we came up with a method to find extremely significant and easily observable irregularities. Data from Azure Core workload insights was subjected to this procedure for the purpose of identifying highly serious problems.
We implemented a system that we called MARIO. The generalisation of our strategy is demonstrated by the fact that we were able to get effective outcomes on two benchmark time-series datasets, namely the Volatility dataset and the Electricity dataset. Our performance is superior to that of two well-known state-of-the-art approaches, namely TFT and DeepAR, when it comes to locating highly significant anomalies in time-series measurements.

Literature Review

The traditional Internet was susceptible to spoofing and sniffer attacks because to its foundation in transmission control protocol/internet protocol (TCP/IP). A larger amount of data was transmitted across the Internet as it evolved. Intellectual property and commercial details were among the critical pieces of information contained in this collection. Yet, attackers can readily steal data or alter packets during transmission because TCP and IP do not by default offer a technique for safeguarding communication channels, such encryption. The Internet has been fortified with protected communication channels like transport layer security (TLS) in response to security threats [5]. A notable shift from earlier times is the increasing share of encrypted traffic on the Internet.
The Internet is protected by encryption technologies, yet attackers can still launch cyberattacks across protected communication channels. According to research, the proportion of attacks conducted over encrypted channels increased steadily over the years, rising from 57% in 2020 to 80% in 2021 and then reaching over 85% in 2022 [6]. The method of deep packet inspection (DPI) employed to recover attack evidence from packets is now obsolete [7]. Taking an ecological stance is now untenable. There are now methods for identifying abnormalities in network

traffic, such as analysing plaintext data or relying on information included inside packets for anomaly detection [8]. Because it uses essential data, including the payload, in plaintext traffic, this anomaly detection approach cannot be applied to encrypted communication which uses key data in packets. This is due to the fact that crucial information is encrypted.

Anomaly detection over encrypted traffic can be conceptualised with the use of artificial intelligence technologies like deep learning and machine learning [9]. Examining anomaly detection over encrypted traffic is the focus of an ongoing AI-based research project. In their study on anomaly detection in encrypted traffic, the authors of [10] used traditional machine learning methods such as SVM and XGBoost. The research article [11] examined anomaly detection over encrypted communications using deep learning techniques such as CNN and LSTM. Contrarily, research on anomaly detection using artificial intelligence over encrypted traffic has not been thoroughly evaluated. Regardless, proactive research on anomaly identification over encrypted traffic based on AI is presently underway.

A systematic literature review, often known as an SLR, is a research approach that includes the process of collecting and evaluating the results and material of previous research on a particular subject in a manner that is both systematic and standardised. Research questions are developed, a literature search is carried out, data is extracted, quality is assessed, results are interpreted, and finally, a discussion is held as part of the SLR process.

We are able to determine the present degree of technology for the research issue by utilising this standardised analysis approach. Additionally, we are able to uncover knowledge gaps that have not yet been explored.

Table 1: Summary of the identified literature

Article	Dataset	Encryption Protocol	Feature Extraction	Feature Selection	Preprocessing	Classification Algorithm	Performance Metrics
[12]	ISCX VPN-NONVPN, USTC_2016, Self-collected data in the power system environment	SSL, SFTP, FTPS	-	Select only the first 784 bytes of the session to use as a feature	Length unification, Convert to two-dimensional data, Converted to 2D grayscale images	1D-CNN, 2D-CNN	Precision, Recall
[13]	NSL-KDD UNSW-NB15 CIC-IDS-2017	SSH TLS, SSH TLS, SSH,	Statistics-based feature extraction	-	Normalization (0,1), One-Hot Encoding	CNN, LSTM, GRU, CNN+GRU	Accuracy, Precision, Recall, FPR, F1-Score
[14]	Self-collection Slow DoS dataset	TLS	Directly implemented Conversation Processor	-	Normalization	Autoencoder	Accuracy, Precision, Recall, FPR, F1-Score
[15]	CTU-13	TLS	Log Information-Based Feature Extraction	Analysis of variance (ANOVA) method and mutual information (MI)	-	Random Forest, XGBoost, GNB	Accuracy, Precision, Recall, F1-score, FPR
[16]	CTU-Malware-	TLS	-	Select only the first	Length unification,	ResNet	Accuracy, Precision,

	Captures			784 bytes of the session to use as a feature	data cleaning		Recall, F1-score, MCC
[17]	Datacon 2020 Dataset	TLS	Log Information-Based Feature Extraction	-	-	Ensemble (RF, NB, TEXTCNN)	Recall, FPR
[18]	Mixed (ISCX VPN-nonVPN, CTU-13)	SSL/TLS	Statistics-based feature extraction	-	Data cleaning, Length unification, Converted to 2D grayscale images	Efficientnet	Accuracy, Precision, Recall, F1-score
[19]	CTU-13	TLS	Statistics-based feature extraction	-	-	SVM, 1D-CNN	Accuracy, Precision, Recall, F1-score, FPR
[20]	Mixed ISCX VPN-nonVPN, ISCX 2012 IDS)	SSL, HTTPS	-	-	Package Generation, Traffic Purification, Traffic Refiner, Length unification	1D-CNN, LSTM, SAE	Precision, Recall, F1-score

Leveraging Azure Anomaly Detector for Business Insights

Integrating Azure Anomaly Detector into existing data analytics workflows is relatively straightforward, thanks to its seamless integration with other Azure services and APIs. Here are some steps businesses can take to leverage Azure Anomaly Detector effectively:

- Data Preparation: Prepare the time-series data by cleaning, formatting, and structuring it appropriately for analysis. Ensure the data contains relevant features and attributes to help identify anomalies effectively.
- Model Training: Train the Azure Anomaly Detector model using historical data to learn normal patterns and behaviors. Choose an appropriate anomaly detection algorithm and fine-tune model parameters based on the specific requirements of the use case.
- Real-time Monitoring: Deploy the trained model to monitor real-time data streams and detect anomalies as they occur. Set up alerts or notifications to notify stakeholders whenever anomalies are detected, enabling timely intervention and decision-making.
- Continuous Improvement: In order to keep up with evolving trends and guarantee reliable anomaly detection over time, it is essential to constantly track the model's performance and retrain it with updated data at regular intervals. Incorporate feedback from stakeholders to refine the model and improve its effectiveness.

A. Applications Across Industries

Azure Anomaly Detector finds applications across various industries where detecting anomalies in data streams is critical for ensuring operational efficiency, detecting fraud, preventing equipment failures, optimizing resource utilization, and enhancing customer experiences. Let’s explore some industry-specific use cases:

- Finance and Banking:

In the finance sector, Azure Anomaly Detector can detect unusual patterns in financial transactions, identify fraudulent activities such as unauthorized transactions or account takeovers, and monitor stock market fluctuations. In order to reduce risks, safeguard client funds, and stay in compliance with regulations, financial institutions can implement real-time anomaly detection systems.

B. Manufacturing and IoT:

In manufacturing and IoT (Internet of Things) environments, Azure Anomaly Detector can analyze sensor data from machinery and equipment to identify anomalies that may indicate potential equipment failures or maintenance issues. Manufacturers can optimize production processes, reduce maintenance costs, and ensure smooth operations by predicting and preventing equipment downtime.

- Retail and E-commerce: In retail and e-commerce, Azure Anomaly Detector can analyze sales data to detect anomalies in purchasing patterns, identify trends, and predict demand fluctuations. This data can help stores improve customer service, tailor marketing to each individual, and streamline inventory management.
- Healthcare: In the healthcare industry, Azure Anomaly Detector can analyze patient data, such as vital signs or medical sensor readings, to detect anomalies indicating health complications or deviations from normal health conditions. Healthcare providers can use this information for early detection of diseases, patient monitoring, and improving clinical decision-making.

Methodology

System: Maintainability Availability Reliability Intelligence Ops (MARIO)

The Azure Core workload insights are managed by the MARIO application, which is a deployed application. This service, which makes use of AIOps-powered Insights in Azure Core, offers product solutions that simplify the process of managing client workloads, both for the customer and the support team.

A. MARIO Horizontal Features

Horizon 1 - Only concentrating on the telemetry of Azure, monitoring the workloads that are hosted on Azure. An application that interacts with customers or a procedure that runs on the back end are examples of workloads. Workloads are collections of resources and code that give value to a business.

End-to-End visibility of workload: Being able to see the big picture from start to finish of the entire task and seeing exactly where things went wrong or could go wrong is important. Over a period of time, the system need to be able to acquire knowledge on the workload.

Intelligent monitoring: The initial step in implementing Intelligent Operations is data collecting. For this data to be properly examined, it needs to be obtained from multiple sources and then linked to another data source. With these end-to-end insights accessible throughout the application stack, it would be simpler to ensure the reliability and high availability of business applications. The back-end infrastructure, customer behaviour, and performance are all part of this.

Detect issues proactively and remediate: Through the collection and examination of data, the process of making sophisticated automated choices. The utilisation of this data enables us to forecast likely future occurrences that may have an effect on availability and performance, and we can even take preventative measures to address those occurrences before they become a problem.

Faster root cause analysis: The study of the events, logs, and metrics generated by the tools can be automated, allowing for faster root cause analysis.

It does not rely on a single tool or data source but rather incorporate the insights that are gleaned from the complete tool chain.

Recommendations: Analysis of telemetry, logs, alarms, and events over time with the purpose of identifying trends and offering solutions to the observed issues and anomalies. Additionally, we can aim to provide the best practices that can be included into the project to forestall future issues by providing ideas for the most effective methods.

Horizon 2 - The following are examples of workloads that are hosted on Azure and have integration with open telemetry: The monitoring of workloads on Azure through the utilisation of additional telemetry from customer apps that are not associated with Azure. Have the capability to monitor workloads on both Azure and hybrid clouds. Telemetry workloads and monitor workloads are the two distinct types of data that we have utilised in the

implementation of our work together. Metrics for monitor workloads include count, duration, storage space, and user count, in contrast to the time-based measurement metrics for telemetry data.

B. System Details and Data Overview

This section provides an overview of the data structure, as well as high-level facts about the system and the data collection process.

Challenges and Opportunities with ML for Azure Core World

Accelerate Time to Value: As Microsoft teams utilise ML and AI to improve their products and services, the issue of time to value keeps coming up.

Although the initial model creation and training might not take much time (e.g., two weeks), teams face a significant burden when trying to operationalize at scale with MLOps, governance, and tracking. Economies of scale and quicker value realisation are necessary for a solution. We take raw workload data from the MARIO Service, analyse it, undertake feature engineering, build a model to identify outliers, and then make it public as an Azure MLaaS service.

Drive Adoption by Reducing Cognitive Load: Adoption and, by extension, the effect on businesses, are heavily influenced by the ways in which insights are presented in apps and experiences. Low utilisation and traction would result from providing too many faults/anomalies because consumers would be overwhelmed and unable to find out how to increase their productivity with these insights. Users require the most relevant data to be presented to them in a way that aligns with their processes so that they can make informed decisions. They should also include the processes into their workflow wherever feasible. A false alert will be marked and details like confidence scores, upper and lower series bounds, and anything else that could help find the source of the anomaly will be provided when an anomaly happens in an Azure core workload. By keeping themselves apprised of the poster's response (correction/false flag), the reviewer can do more reviews as needed and remain well-informed. Lastly, we check in on system-identified abnormalities and the steps used to address them on a regular basis.

Reduce System fault by Catching Anomalies Early: With the help of a workload resource map, application system failures may be proactively discovered and relevant measures can be planned and executed in a timely manner.

Increase Transparency through Confidence Score: Users lose faith in and comfort with ML models when they lack transparency, which in turn slows down adoption. They require open and explainable systems where the suggestions can be made. From a review and auditing perspective, it is crucial that the model's risk mitigation efforts are transparent. We are incorporating confidence scores to propel this degree of certainty.

C. Data Collection by MARIO Service

- The customer has the option to use multiple deployment orchestration to deploy the infrastructure artefacts. Learning how resources interact with dependents during runtime is one of the primary aims of MARIO Service. In a hierarchical structure, all these artefacts are interconnected. At the very top is the application, or "front door," and the resources it makes use of are the "child nodes" below it. An application outage can be caused by a problem with a child node.
- For every customer application hosted by an Azure resource artefact, data is gathered. N-the total amount of resources in Azure. For example, AKS, Compute, Storage, SQLDB, Cosmos DB, and so on.
- For every Azure resource, there will be X-dimensional data for anomaly detection, which is the sum of M metrics (such as CPU utilisation) and N-dimensional data (such as clusters, servers, pods, databases, etc.).
- Depending on the Azure resource, metrics may be shared or specific.

D. Data Structure Overview

The MARIO service uses Azure Monitor insights to gather data from the tenant of the customer. An optimised time-series database stores all of the data collected in Azure Monitor. This database is designed to analyse data that is timestamped. A metric is a numerical value that, at a given instant in time, describes some feature of a system. They have a timestamp, a name, a value, and maybe some identifying labels; they are gathered at predetermined intervals. A number of methods can be used to aggregate measurements, compare them to one another, and examine patterns over time. Every collection of metrics is a time series that includes: The moment when the sum was tallied. The item with which the value is linked. A name-space that the metric uses as a category. A measure's descriptor. Simply said, its worth. The name/value pairs that accompany a metric's dimensions provide further information about the metric's value. An "Available disc space" metric, for instance, could have a "Drive" dimension with the values C and D. With Dimension, you may see the total amount of disc space on all drives or on a per-drive basis.

Each metric (M) for a specific resource or dimension needs to have its anomalies found utilising time series of values from the N-dimension. In order to upload metrics from Azure Monitor to Mario's storage endpoint, queries are executed. Every resource in a JSON file follows the same standard format. At the CGAADLSDEV endpoint, we have a pipeline that searches Mario's workspace for recently posted JSON files and parses the nested JSON data into a tabular delta table. Table 2 provides a summarised view of the data for a sample period and highlights the metrics, dimensions, resource categories, dimension values, and record counts for each meta-info.

Table 2: Parsed data summary for sample time-frame

Unique Resource Type	Unique Metrics	Unique Dimensions	Unique Dimension Values (Time-series)	Records
Microsoft.DocumentDB	18	13	1,302	13,914,350
Microsoft.Storage	7	6	168	4,215,037
Microsoft.Network	30	12	2,091	26,769,228
Microsoft.ContainerService	52	28	30,051	91,177,800
Microsoft.ServiceBus	19	2	33	3,000,996
Microsoft.Sql	26	1	2	406,443
Total	152	62	33,647	139,483,854

Result

- By using EVT over upgraded ADaaS, we were able to reduce the number of anomalies with substantial reconstruction loss from 149 to 8. The highest confidence score is likewise held by these outliers.
- As a result, on the MARIO dashboard, we will display high-priority or high-significant anomalies (in this case, 8 anomalies) together with the related metrics and resource information hourly for user perception.
- Users will have the option to view the remaining 141 anomalies as low priority or low significance if they so desire.
- EVT has not yet flagged them. In comparison to highly significant anomalies, these ones have lower confidence scores.

A. User-Study

- For future model enhancements, we collect user-study input or human-in-the-loop validation on the service's anomaly validity. The user-set consists of thousands of actual Azure Workload customers. Among these tasks are:
  - It is a component of the Supply Chain and its 'Device Care' task in the Cloud. ACSS, or Azure Centre for SAP Solutions, is a service that elevates SAP to the status of a top-tier workload on the Azure platform.
  - o- DPS: Digital Professional Services develops online infrastructure that helps customer success, industry solutions, and support groups speed up results for their business clients.

For really noteworthy outliers, we deduced a True Positive Rate of 98%. For small, non-significant abnormalities, our method was validated with a 4.92% False Positive Rate and a 4.3 percent False Negative Rate.

Table 3: Results on benchmark datasets

Dataset	Method	Quantile %	Anomaly Count	Anomaly %	SMAPE	Precision	Recall
	Enhanced ADaaS	99.998	26	0.29	23.98	0.88	0.82
Electricity	Enhanced ADaaS						
	+EVT	99.998	9	0.10	23.26	0.92	0.94
	Enhanced ADaaS	99.998	21	4.12	45.57	0.84	0.85
Volatility	Enhanced ADaaS						
	+EVT	99.998	8	1.57	45.14	0.945	0.95

B. Generalisation of Method

We conducted experiments on two widely used benchmark datasets to demonstrate the generalizability and practicality of our Enhanced ADaaS + EVT method:

Electricity: On an hourly basis, the data from the UCI Electricity Load Diagrams dataset is aggregated. This dataset contains the electricity consumption of 370 users. To identify outliers within the last day, we look at the previous seven days (or 168 hours).

Volatility: The OMI realised library contains 31 stock indexes' realised volatility levels and daily returns computed from intraday data. Using data collected over the last twelve months, we investigate anomaly detection in the recent seven days (five business days). Our experimental setup is identical to that described.

Anomalies discovered in the Electricity and Volatility datasets using the Enhanced ADaaS and Enhanced ADaaS + EVT methodologies are listed in Table 3, along with the percentage of each. The second approach, the Enhanced ADaaS + EVT, finds a negligible amount of outliers (0.1% in the Electricity dataset and 1.57% in the Volatility dataset), and its SMAPE value is almost identical to the first.

The table only shows SMAPE values for data points that were found to be abnormal. What this means for user perception is that we are getting better at spotting highly significant anomalies with large reconstruction errors in the model. According to the data in the table, the Enhanced ADaaS approach achieves very high values for measures such as Precision and Recall, which are greater than 0.8. This supports our claim that we can get a high rate of genuine positives while reducing the number of false negatives and positives. A rise in these metrics to levels over 0.92 and a recall value of 0.95 with the volatility dataset are observed when using the Enhanced ADaaS + EVT approach.

Table 4: Our method comparison results with state-of-art methods

Dataset	Method	Quantile%	Anomaly Count	Anomaly%	SMAPE	Precision	Recall
Electricity	TFT	99.998	197	2.22	23.49	0.533	0.445
	DeepAR	99.998	250	2.82	23.84	0.567	0.434
	Enhanced ADaaS +EVT	99.998	9	0.10	23.26	0.921	0.943
Volatility	TFT	99.998	112	21.96	45.31	0.458	0.469
	DeepAR	99.998	132	25.88	45.87	0.513	0.445
	Enhanced ADaaS +EVT	99.998	8	1.57	45.14	0.945	0.952

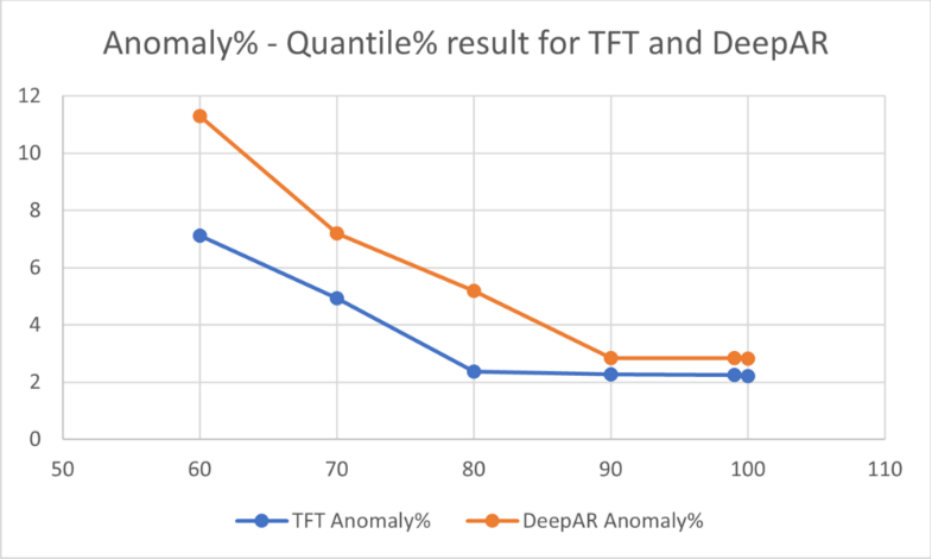

Figure 2: Comparison of TFT and DeepAR performance at different quantiles reveals a plateau.

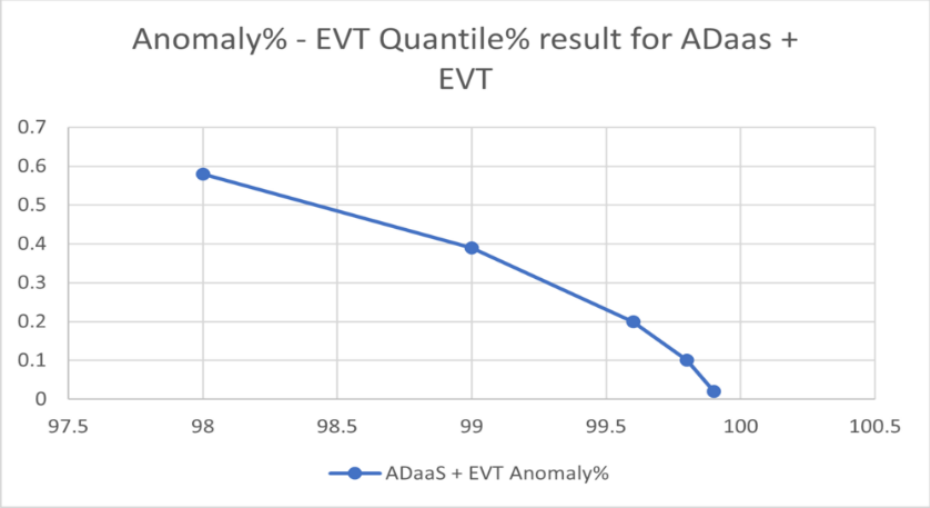

Figure 3: Plot showing EVT's growth performance with different quantiles.

As seen in Figure 2, algorithms like TFT and DeepAR have hit a performance ceiling at around 99.998% quantile value on the Electricity Dataset, where they have plateaued. The anomaly% (y-axis) remains unchanged regardless of the increase in the quantile% (x-axis). The 99.998% quantile has remained unchanged for our Enhanced ADaaS + EVT trial's hyper-parameter. However, we have another hyperparameter to tweak—the risk factor/quantile to employ in EVT—by taking use of EVT, which begins filtering in these high quantile ranges. Figure 3 shows that EVT works well in the high quantile range, and that the number of anomalies discovered falls without reaching a plateau as the EVT quantile% (x-axis) increases.

Conclusion

Using data from a variety of KPIs and resources, our system can automatically detect faults and anomalies in Azure Core workloads. Our analysis reveals a handful of outliers that stand out on the user dashboard due to their high reconstruction inaccuracy and their significance. To detect anomalies or defects in time-series data, we have a generic approach that, with little tweaking, works in many different contexts. As for the optimisation of time and resources, there are a few unanswered questions that we hope to resolve in due course. Prioritising data with availability metrics for products was our top priority. Azure Data Lake is where this data is kept, and we're helping with the training and inference by providing endpoint scoring. So far, we have trained on data from the past seven days and inferred from the last hour. We built the models in Azure ML Studio and performed data processing with Azure Databricks. We intend to incorporate all indicators from the workload insights data, which includes 42,000 time series for a whole week's worth of data, into the solution moving forward. We want to test our approach using the whole three months of data from the Azure Core workload. So that we may record various relationships between resources and measurements, we are aiming for a solution based on multi-variate anomaly detection. You can access our system solution through Microsoft Azure, and it is already in production.

References

[1]. Oord, A. v. d.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.; and Kavukcuoglu, K. 2016.Wavenet: A generative model for raw audio.arXiv preprint arXiv:1609.03499.

[2]. Lim, B.; and Zohren, S. 2021.Time-series forecasting with deep learning: a survey.Philosophical Transactions of the Royal Society A, 379(2194): 20200209.

[3]. Salinas, D.; Flunkert, V.; Gasthaus, J.; and Januschowski, T. 2020.DeepAR: Probabilistic forecasting with autoregressive recurrent networks.International Journal of Forecasting, 36(3): 1181–1191.

[4]. Siffer, A.; Fouque, P.-A.; Termier, A.; and Largouet, C. 2017.Anomaly detection in streams with extreme value theory.In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1067–1075

[5]. Chen, L.; Gao, S.; Liu, B.; Lu, Z.; Jiang, Z. THS-IDPC: A three-stage hierarchical sampling method based on improved density peaks clustering algorithm for encrypted malicious traffic detection. J. Supercomput. 2020, 76, 7489–7518. [Google Scholar] [CrossRef]
[6]. Bakhshi, T.; Ghita, B. Anomaly detection in encrypted internet traffic using hybrid deep learning. Secur. Commun. Netw. 2021, 2021, 5363750. [Google Scholar] [CrossRef]
[7]. Moher, D.; Liberati, A.; Tetzlaff, J.; Altman, D.G.; Prisma Group. Preferred reporting items for systematic reviews and meta-analyses: The PRISMA statement. Ann. Intern. Med. 2009, 151, 264–269. [Google Scholar] [CrossRef] [PubMed]
[8]. Keele, S. Guidelines for Performing Systematic Literature Reviews in Software Engineering; Technical report, ver. 2.3 ebse technical report; School of Computer Science and Mathematics Keele University Keele: Staffs, UK, 2007. [Google Scholar]
[9]. Stratosphere Lab. CTU-Malware-Capture-Botnet. Available online: https://www.stratosphereips.org/datasets-malware (accessed on 31 December 2023).
[10]. Duncan, D.B. Malware Traffic Analysis. Available online: https://www.malware-traffic-analysis.net/ (accessed on 31 December 2023).
[11]. Chao, D. A Mining Policy based Malicious Encrypted Traffic Detection Scheme. In Proceedings of the 2020 9th International Conference on Computing and Pattern Recognition, Xiamen, China, 30 October–1 November 2020; pp. 130–135. [Google Scholar]
[12]. Chen, L.; Jiang, Y.; Kuang, X.; Xu, A. Deep learning detection method of encrypted malicious traffic for power grid. In Proceedings of the 2020 IEEE International Conference on Energy Internet (ICEI), Sydney, NSW, Australia, 24–28 August 2020; IEEE: New York, NY, USA, 2020; pp. 86–91. [Google Scholar]
[13]. UNB VPN-nonVPN Dataset (ISCXVPN2016). Available online: https://www.unb.ca/cic/datasets/vpn.html (accessed on 31 December 2023).
[14]. Yungshenglu USTC-TFC2016 Datset. Available online: https://github.com/yungshenglu/USTC-TFC2016 (accessed on 31 December 2023).
[15]. UNB NSL-KDD Dataset. Available online: https://www.unb.ca/cic/datasets/nsl.html (accessed on 31 December 2023).
[16]. UNSW Sydney. The UNSW-NB15 Dataset. Available online: https://research.unsw.edu.au/projects/unsw-nb15-dataset (accessed on 31 December 2023).
[17]. UNB Intrusion Detection Evaluation Dataset (CIC-IDS2017). Available online: https://www.unb.ca/cic/datasets/ids-2017.html (accessed on 31 December 2023).
[18]. Garcia, N.; Alcaniz, T.; González-Vidal, A.; Bernabe, J.B.; Rivera, D.; Skarmeta, A. Distributed real-time SlowDoS attacks detection over encrypted traffic using Artificial Intelligence. J. Netw. Comput. Appl. 2021, 173, 102871. [Google Scholar] [CrossRef]
[19]. Huo, Y.; Zhao, F.; Zhang, H.; Zhuang, S.; Sun, J. AS-DMF: A Lightweight Malware Encrypted Traffic Detection Method Based on Active Learning and Feature Selection. Wirel. Commun. Mob. Comput. Online 2022, 2022, 1556768. [Google Scholar] [CrossRef]
[20]. Stratosphere Lab. The CTU-13 Dataset. Available online: https://www.stratosphereips.org/datasets-ctu13 (accessed on 31 December 2023).