Journal of Natural Sciences and Engineering, Vol. 3, Issue 1(2021)
DOI number: 10.14706/JONSAE2021320

Using Exploratory Data Analysis and Big Data Analytics for Detecting Anomalies in Cloud Computing

Ibrahim Muzaferija[1], Zerina Mašetić[1]
[1]International Burch University, Sarajevo, Bosnia and Herzegovina
ibrahim.muzaferija@stu.ibu.edu.ba
zerina.masetic@ibu.edu.ba

Abstract – While leveraging cloud computing for large-scale distributed applications allows seamless scaling, many companies struggle following up with the amount of data generated in terms of efficient processing and anomaly detection, which is a necessary part of the management of modern applications. As the record of user behavior, weblogs surely become the research item related to anomaly detection. Many anomaly detection methods based on automated log analysis have been proposed. However, not in the context of big data applications where anomalous behavior needs to be detected in understanding phases prior to modeling a system for such use. Big Data Analytics often ignores anomalous point due to high volume of data. To address this problem, we propose a complemented methodology for Big Data Analytics – the Exploratory Data Analysis, which assists in gaining insight into data relationships without the classical hypothesis modeling. In that way, we can gain better understanding of the patterns and spot anomalies. Results show that Exploratory Data Analysis facilitates anomaly detection and the CRISP-DM Business Understanding phase, making it one of the key steps in the Data Understanding phase.

Keywords - Cloud Computing, Big Data, Data Mining, Anomaly Detection

1. Introduction

With constant growth and advancements of the Internet, there are more systems connected to other connected systems, constantly generating and exchanging data. That data is referred to as Big Data and is constantly targeted by cyber-attacks as it contains sensitive and valuable information. The term "big data" refers to data that is so large, complex, or rapid that it's not possible to process using traditional computing and data management tools. Big Data provides opportunities to improve research, operational efficiency, and decision-support applications with increased value for digital applications [1]. At the same time, Big Data represents the challenges to store, transport, process, mine, and serve the data. Data that is high in volume, velocity, variety, and veracity must be processed with advanced analytical tools and algorithms to reveal meaningful information and provide value.

Cloud computing represents the use of distributed and shared resources such as computing, storage, networking, and analytical software, and provides fundamental support to address the challenges of Big

Data. Cloud computing serves both as a technological enabler and producer of big data [1].

Anomalies represent unusual or behaviors that deviate from the normal. In efforts to increase cloud computing reliability, anomaly detection poses a frequent problem in threat detection and identification, as reported by Cloud Security Alliance (CSA) [2] which represents the world's leading organization dedicated to securing cloud computing environments, conducts annual research with an aim to raise awareness of threats, risks, and vulnerabilities in the cloud environment. In their latest (2019) report [3], CSA re-examined the risks with cloud security and took a new approach, examining the problems in configuration and authentication, rather than the traditional focus on vulnerabilities and malware, highlighting the following threats:

1. Data Breaches
2. Misconfiguration and inadequate change control
3. Lack of cloud security architecture and strategy
4. Insufficient identity, credential, access, and key management
5. Account hijacking
6. Insider threat
7. Insecure interfaces and APIs
8. Weak control plane
9. Metastructure and applistructure failures
10. Limited cloud usage visibility
11. Abuse and nefarious use of cloud services

In this research, we aim to address the threats which can be traced in user logs (numbered 1, 4, 5, 6, 8, 9 and 11) by utilizing Big Data Analytics and Exploratory Data Analysis in order to discover anomalies and contribute to increase of security in Cloud Computing applications.

2. Literature Review

Anomaly detection in the cloud infrastructure and big data environment has been the topic of many research studies in the literature. Since the first introduction of cloud infrastructure in 2006 [4], cloud computing has greatly impacted the industries. The rapid development of Internet and Big Data technologies has resulted in increased service development on cloud computing, such as online banking services, electronic news services, government information systems, mobile services, etc. These systems handle sensitive and confidential data, making the anomaly detection mechanisms one of its core security requirements.

In the review paper by Arif Sari [4], [5], different techniques and mechanisms used in the detection of anomalous activities within the cloud environment are described: threshold detection, statistical analysis, rule-based measures, data mining, and machine learning. We aim to apply statistical techniques and EDA (Exploratory Data Analysis) in order to discover anomalies.

In the “Big Data processing for Anomaly Detection” survey [6], Ariyaluran et al. present the details of the comparative analysis and the relationship of three different domains, which are anomaly detection, machine-learning algorithms, and real-time big data processing. This paper aims to contribute to complemented techniques for anomaly detection. Once anomalies are detected, we can utilize Machine Learning and real-time anomaly detection for future improvements.

In their research, Dalal and Rele [6], [7] emphasize the steps in creating effective and reliable mechanisms for threat detection. They highlight the importance of the first CRISP-DM (Cross Industry Standardized Process for Data Mining) phase named “Develop Business Understanding”, where reasons for defects and answers for maintenance are taken into consideration. They discuss the phase “Analyze Data and Data Dependencies” where the aim is to analyze, combine, and compare the data with the present situation, without proposing EDA as a baseline for data understanding. Our work aims to employ EDA in order to complement the methodology.

Also, they highlight the step named “Engage with Subject Matter Experts (SME’s)” for better dataset examination and analysis of the anomaly situation, along with a grouping of the threat factors. By employing these methods, we aim to set transparent expectations and bring out clarity to our results. In further research, we work closely with application development technical lead which serves as SME, and facilitates in clarification of log data, as well as threats, anomalies and our results

3. Methodology

The research is implemented using a portion of the CRISP-DM (Cross Industry Standardized Process for Data Mining) methodology [8], which represents the common standards used by data scientists and data mining experts in order to build analytical and machine learning models. Prior to analytical and machine learning model creation, we need to construct a clean dataset of user behavior with anomalies labeled for future modeling. To do so, in this research we focus on the first three phases: Business Understanding, Data Understanding, and Data Preparation, as highlighted with red color in the figure below. Modeling and subsequent phases are researched in our extended study of anomaly detection in cloud computing.

Journal of Natural Sciences and Engineering, Vol. 3, Issue 1(2021)
DOI number: 10.14706/JONSAE2021320

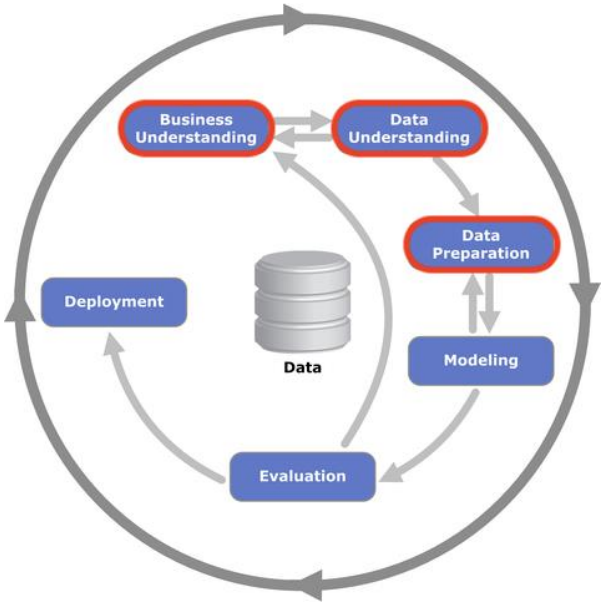

Figure 1. CRISP-DM workflow

In the Business Understanding phase, the goal is to determine business objectives, assess the situation from a business perspective, discuss with subject matter experts, determine data mining goals, and produce a project plan. In the Data Understanding, we collect and select raw data, describe and explore the data, consult with subject matter experts, and verify data quality. In the Data Preparation phase, which is often the most time-consuming phase, we select and clean the data, format data, and construct a clean dataset.

We approach the mentioned phases using Big Data Analytics and Exploratory Data Analysis (EDA). Big Data Analytics examines large amounts of data in a non-traditional manner, that is using distributed and shared resources to support the data quantity and complexity [8], [9]. Exploratory Data Analysis [10] is an approach to analyzing data in order to summarize their main characteristics and uncover the underlying structure using statistical and visual methods.

3.1. Data Collection and Selection

Cloud-based enterprise web application logs are produced by multiple servers and services, which are streamed to Elasticsearch [11] service, an open-source search, and analytics engine for all types of data. Elasticsearch is distributed, fast, and scalable, which makes it an ideal environment for big data ingestion, enrichment, storage, analysis, and visualization.

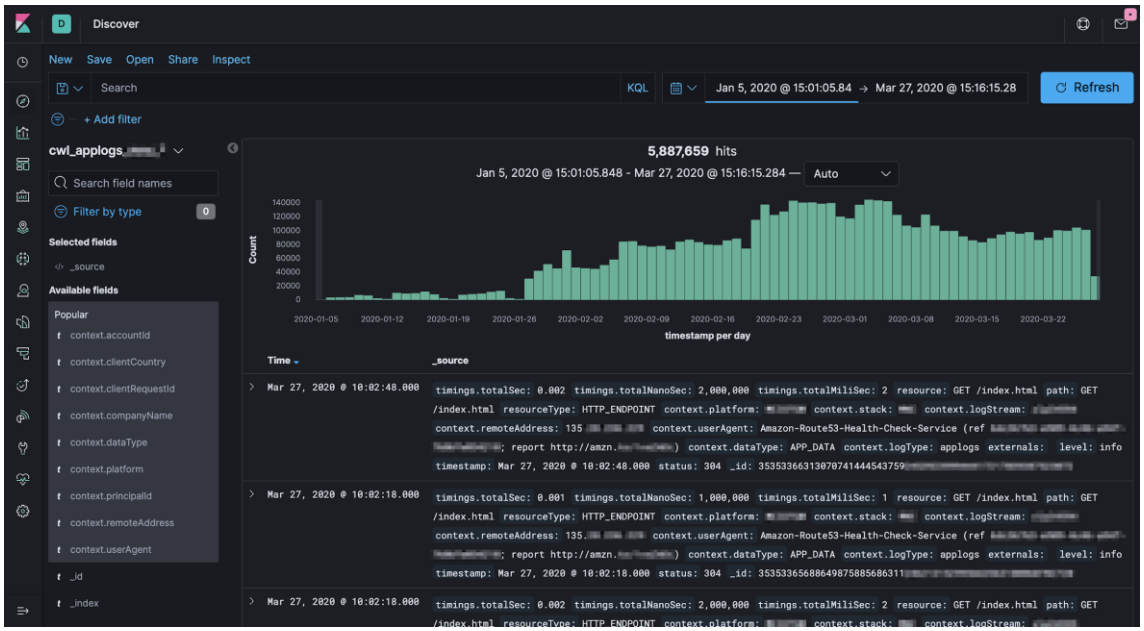

Figure 2. Raw data access from Kibana

Raw data is accessed by locally restoring the Elasticsearch cluster snapshot taken for a period of three months. The cluster contains around 20 GB of semi-structured data collected from different application services and levels, indexed by a timestamp. Application logs are mapped to 175 attributes and accessed using Kibana [12], the Elastic Stack service for data analysis and visualization.

Attribute selection is a part of the “Business understanding” and “Data understanding” phase, implemented together in consultations with application development technical lead, i.e., subject matter expert (which we’ll refer to as SME). The attributes describing the user’s application usage that were the most relevant for anomaly detection are selected for further analysis. The following table displays statistical information for selected attributes.

Table 1. Selected data statistical information

Attribute name	Description	Data type	Range	Missing
timestamp	Timestamp	Date Time	[2020-01-05 21:17, 2020-03-26 21:06]	0.0 %
account_id	Account ID, unique company account identifier	Nominal	f6afd09c-****-****-****-c30a935ccc37, ...	8.87 %
client_country	User country	Nominal	BA, US, ...	9.53 %
company_name	Company Name	Nominal	Company A, Company B, ...	10.17 %
platform	Application platform	Nominal	BrowserMNC, BackendMNC, ...	0.0 %
principal_id	User email	Nominal	developer@**.com, ...	9.64 %
remote_address	User IP address	Nominal	[0.0.0.0. - 255.255.255.255]	9.12 %
user_agent	User-agent	Nominal	Mozilla/5.0 (Windows NT 10.0; Win64; x64) … , ...	0.0 %
error_message	Error message	Nominal	validation error, auth error, ...	99.96 %
message	Log message	Nominal	Profiling, FrontTimings, ...	0.18 %
level	Log level	Nominal	Info, error	0.0 %
path	Parameterized resource request	Nominal	PUT /customer/***/ticket/***, ...	99.78 %
resource	Request	Nominal	(GET) /invoices, ...	0.0 %
status_code	Response code	Nominal	200, 404, ...	10.17 %

Once the relevant data is selected, we utilize Elastic Stack service named Logstash [13] for collecting the data, that is, obtaining the initial dataset in CSV format for further work.

3.2. Data Cleansing and Engineering

In order to get an insight into data quality, graphical and statistical methods were used to detect anomalies, faults, outliers, missing values, etc. Moreover, we engineer new attributes in order to increase the interpretability or decrease data complexity. Exploratory Data Analysis assists understanding of relations between attributes and allows us to spot tendencies, as well as to identify the necessary cleaning steps we have to take.

First, we apply filters to remove log data from automated services, such as health-checks and other application services that don't reflect the user's interactions. Next, we remove attributes that contain a high fraction of missing values because the informational significance of attributes is inconsiderable.

Values of "status_code" attribute are mapped to the corresponding descriptions for better interpretability. We engineer new attributes: "resource_method", "resource_base" and "user_os". The "resource_method" and "resource_base" attributes are created from the values of the "resource" attribute by using regular expressions to extract the relevant information. The "user_os" attribute is created in a similar manner, extracting the relevant information using regular expressions from the "user agent" attribute. Creation of these attributes allows us to focus on the most relevant information and decrease the cardinality of original attributes.

3.3. Dataset Creation

The clean dataset contains 16 attributes describing the application usage, and 522,763 rows with a timestamp attribute range from 6th January to 26th March (81 days).

Data is imported to RapidMiner [14], a data science software platform that provides an integrated environment for data preparation, visualization, machine learning, text mining, and predictive analytics. It is open source and used for commercial applications, as well as for research, education, training, rapid prototyping.

In this phase, we continue with Exploratory Data Analysis in order to discover patterns beyond formal modeling or hypothesis testing tasks. Our aim is to utilize the business understanding to increase the understanding of data and relationships between attributes in order to spot anomalous trends.

As the application is B2B based, we analyze the company data first: company account histogram, statistics and distribution. Next, we analyze the behaviors of users in company and general context. By analyzing the "user" and "user domain" attribute, we spot trends in company context usage and behavior. Analysis of

application resource requests allows us to understand the usage in general context.

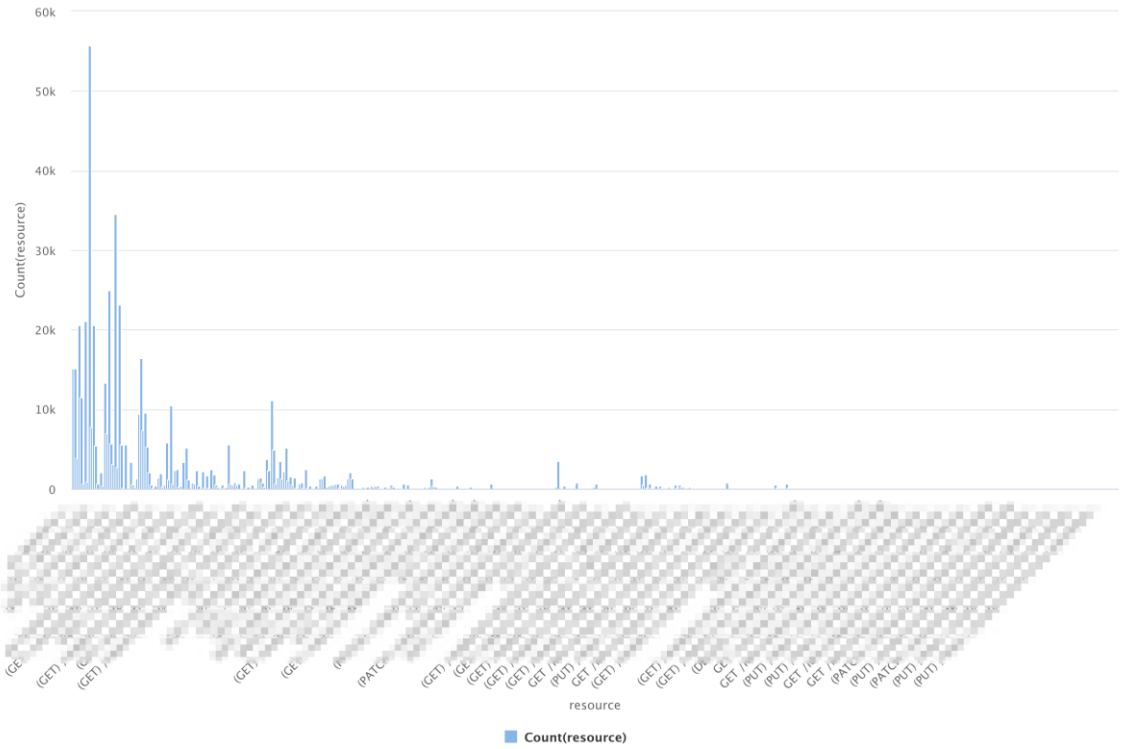

Figure 3. Counts of application resource requests

From the figure above, we can spot trends and further analyze the resource usage. The resource request represents a user action, thus are highly valuable for the context of anomaly detection. Moreover, granular analysis facilitates the business understanding as we gain deeper insight into user generated data.

Next, we analyze the application errors which are often one of the most informative attributes for the anomaly detection. Anomalies and cyber-attacks are often causing application errors, allowing us to quickly analyze error data and make distinctions between application anomalies, user anomalies and possible threats.

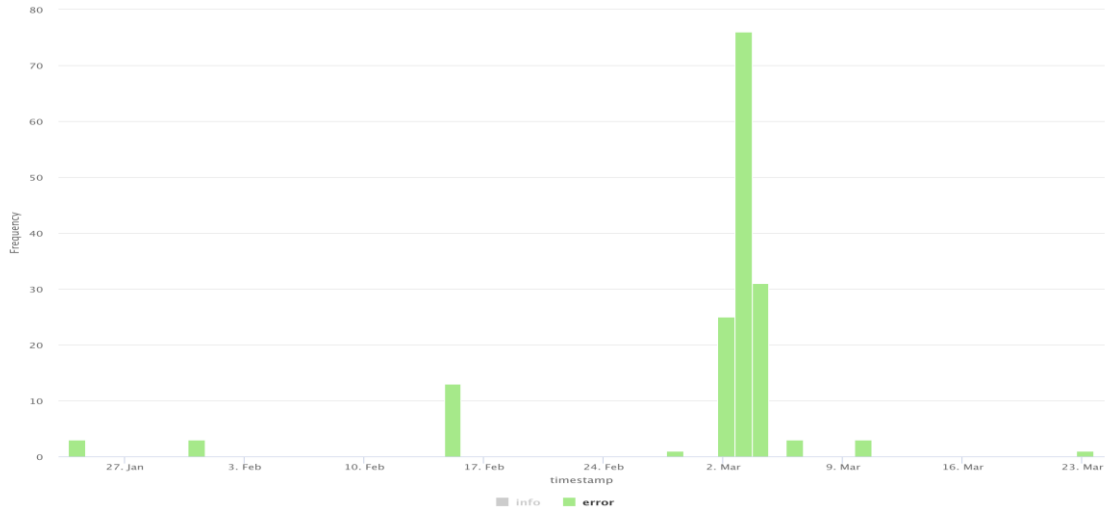

Figure 4. Application error logs histogram

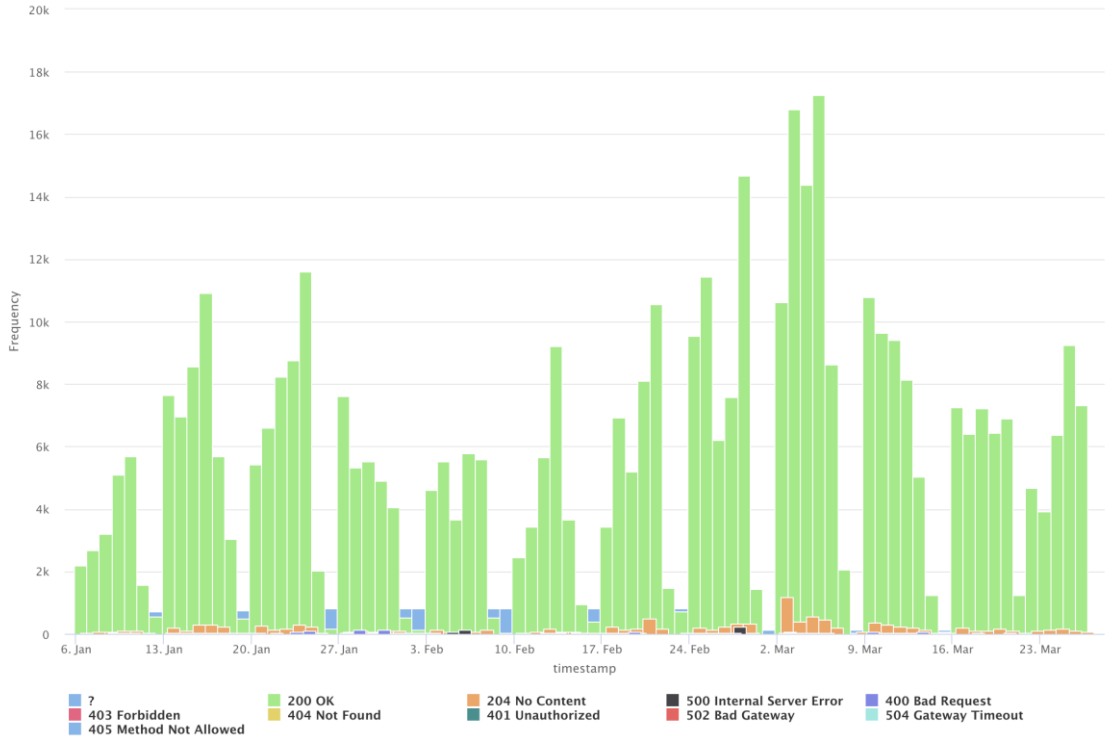

Figure 5. Application logs status codes histogram

Application status codes are highly correlated with application resource usage. By analyzing status codes, we gain insight into applications performance and usage trends. Anomalies are most visible when analyzing the status codes.

Dataset creation is concluded with the creation of an “anomaly” attribute, which represents whether a specific application log instance is anomalous. The criteria for creation of such attribute are drawn from the discoveries of EDA and confirmed through the consultations with SME. By addressing the CRISP-DM

phases for Business Understanding, Data Understanding, and Data Preparation with the application of Exploratory Data Analysis, we are able to discover anomalies in application usage and user behavior.

4. Results and Discussion

As web application has busines-to-busines context, we approach the analysis of log data from a company perspective. We find that companies using the application can have their application usage segmented into three categories: heavy, medium, and light users, as shown below in the Figure 6. Heavy users are the companies responsible for application development and support. Medium users reflect the companies with frequent application usage, while light users represent the companies that are onboarding to application or in initial phases of application usage. Distinction of company users per their level of usage helps us create a better business understanding. Because of unbalanced level of application usage per company, we can expect an increased number of anomalies for heavy users, while companies with medium and light usage may have decreased the number of anomalies. Regarding the percentage of anomalies, it varies between companies with no specific pattern.

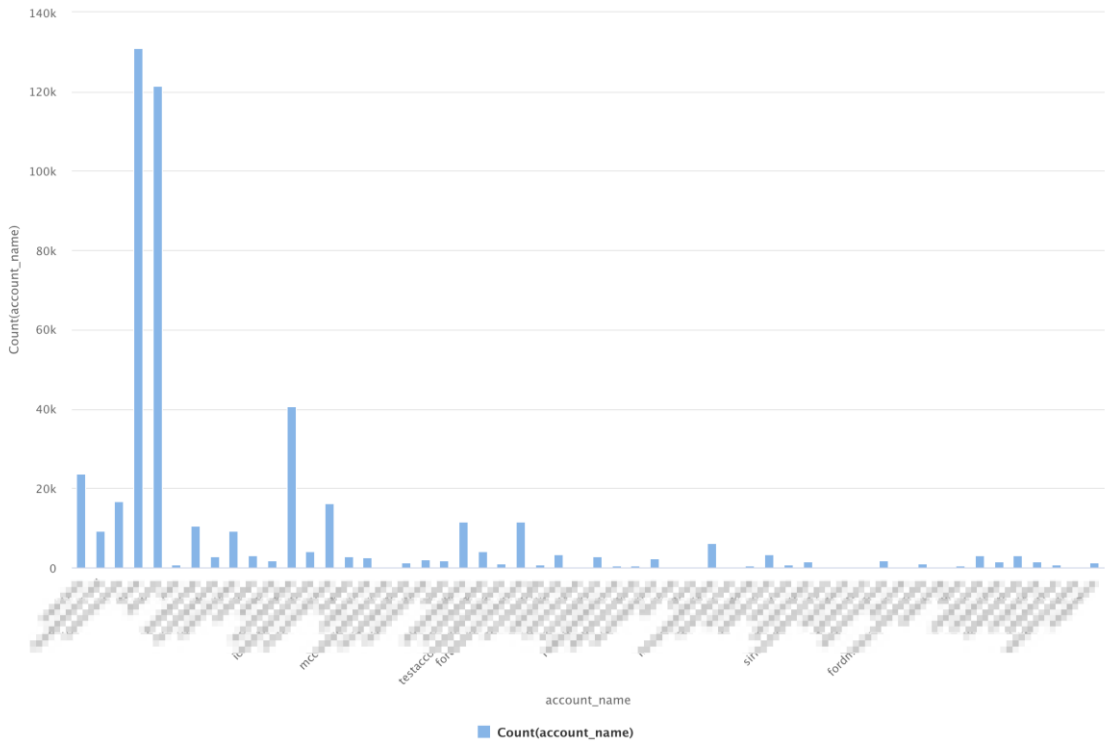

Figure 6. Application usage per company

When analyzing the histogram of application resource methods through the “resource_method” attribute, we find an anomalous request pattern, as shown below in the Figure 7. Consultations with SME yielded that resource request method anomaly corresponds to the service whose use has ceased, and the service behavior can be identified as anomaly.

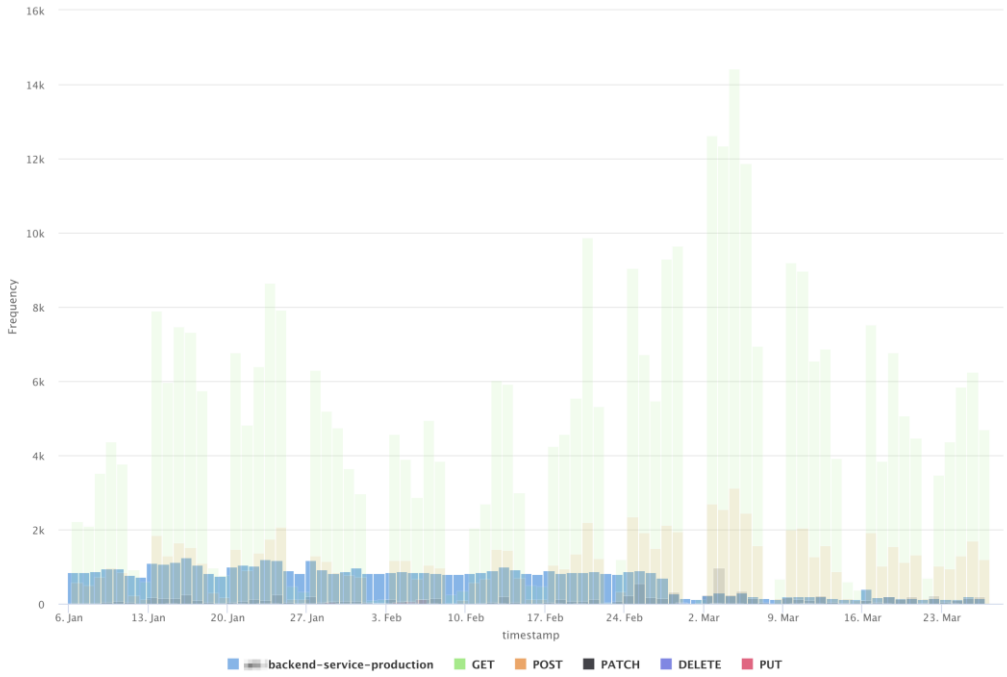

Figure 7. Application resource methods histogram anomaly

When analyzing individual users, we perform segmentation per company using the domain name in user email address. The histogram of user domains contributes to business understanding as we can spot user trends per each company. In the figure below, we present the user domain histogram focused on anomalous application usage of unknown domains. We discover that usage from unknown domains tends to be increased in the monthly peaks of application usage.

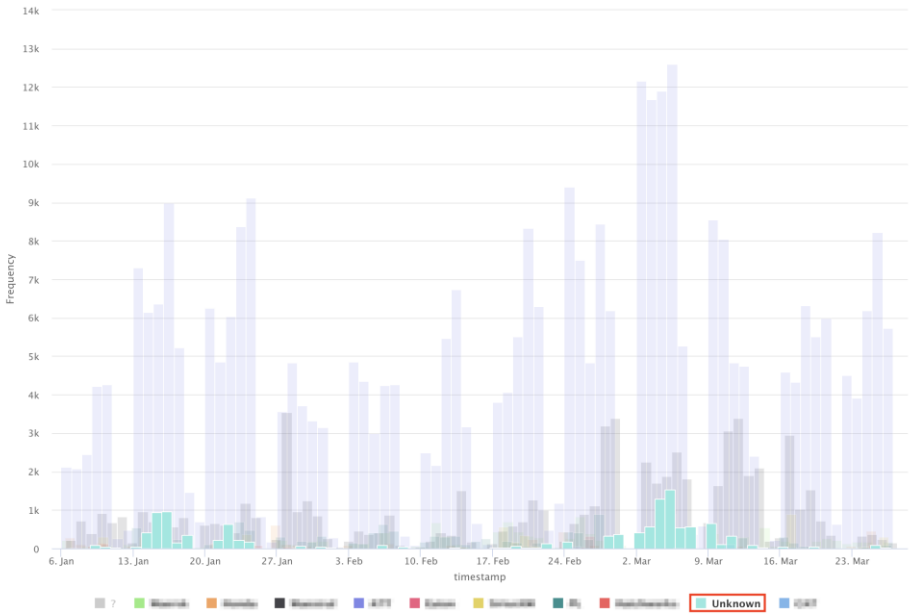

Figure 8. User domain histogram focused on unknown domains

Consultations with SME clarified that unknown domains such as “gmail.com”, “hotmail.com”, and “outlook.com” are used by quality assurance developers and were marked as such. This has further

decreased the number of visits from unknown domains. Moreover, consultations showed that users from unknown domains are companies in the trial phase, that is application demonstration phase, and are still eligible for anomaly detection. Application usage from other user domains is distributed as expected: two development companies take up the most traffic while others are medium and light users.

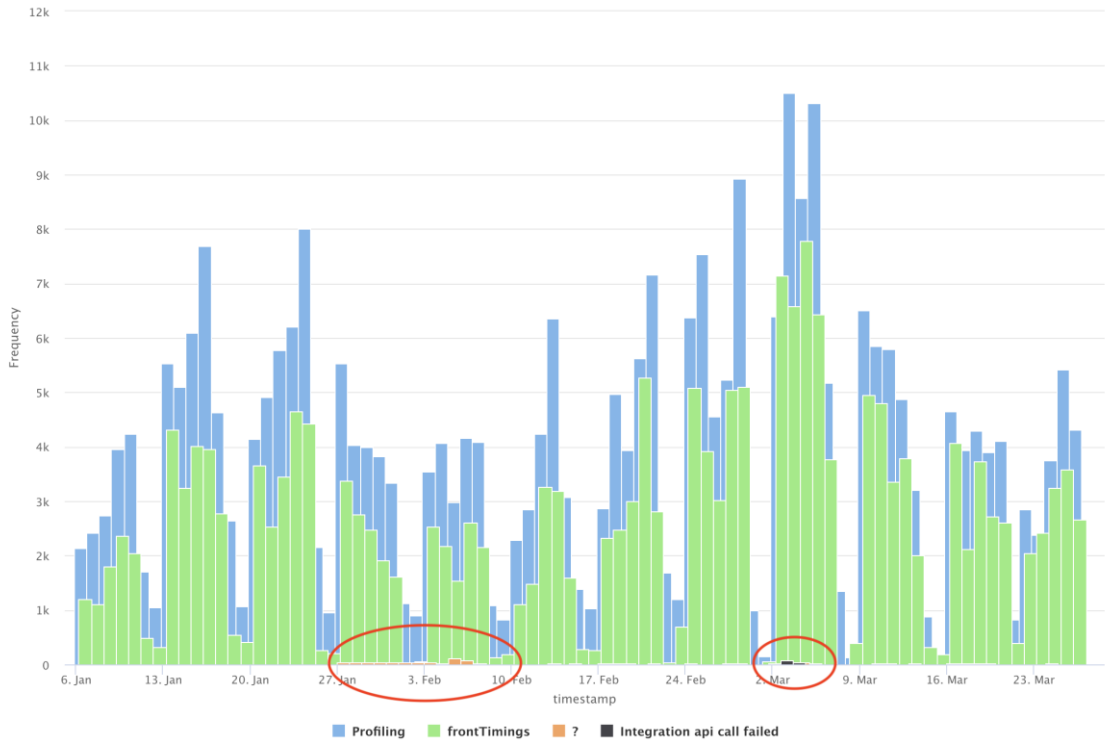

Figure 9. Log message histogram anomalies

In the figure above, we present an analysis result of log message histogram with revealed anomalies. We find that anomalies are caused by application development or, more specifically, integration attempts with other companies using the application.

In the figure below, we present results from correlation analysis of the dataset. The correlation matrix shows increased correlation between attributes such as “platform” and “message”. These results help us to identify and discard highly correlated attributes and decrease the dataset complexity.

Attributes	accou...	coun...	platfo...	user	remote_add...	level	messa...	status_code	user_doma...	resource_meth...	resource_ba...	user_agent_os	timestamp
account_id	1	0.201	0.048	-0.033	0.021	-0.004	0.046	0.086	0.094	0.091	0.037	0.030	0.130
country	0.201	1	0.078	0.185	0.147	-0.006	0.102	0.121	0.336	0.221	0.081	0.012	0.025
platform	0.048	0.078	1	0.003	0.019	0.021	0.791	0.046	0.037	-0.141	0.030	0.045	-0.003
user	-0.033	0.185	0.003	1	0.009	-0.003	0.033	-0.017	0.233	0.088	-0.001	-0.093	0.318
remote_addres	0.021	0.147	0.019	0.009	1	0.012	0.031	0.017	0.057	0.058	0.022	0.146	0.096
level	-0.004	-0.006	0.021	-0.003	0.012	1	0.055	0.503	0.004	-0.009	0.067	0.011	0.008
message	0.046	0.102	0.791	0.033	0.031	0.055	1	0.039	0.056	0.247	0.013	0.045	-0.002
status_code	0.086	0.121	0.046	-0.017	0.017	0.503	0.039	1	0.044	0.089	0.150	0.048	-0.010
user_domain_name	0.094	0.336	0.037	0.233	0.057	0.004	0.056	0.044	1	0.109	0.079	0.060	-0.029
resource_method	0.091	0.221	-0.141	0.088	0.058	-0.009	0.247	0.089	0.109	1	-0.100	0.078	0.012
resource_base	0.037	0.081	0.030	-0.001	0.022	0.067	0.013	0.150	0.079	-0.100	1	0.068	0.026
user_agent_os	0.030	0.012	0.045	-0.093	0.146	0.011	0.045	0.048	0.060	0.078	0.068	1	-0.011
timestamp	0.130	0.025	-0.003	0.318	0.096	0.008	-0.002	-0.010	-0.029	0.012	0.026	-0.011	1

Figure 10. Correlation matrix

Correlation matrix also shows that attributes “status code” and “level” have a level of correlation. This indicates that application errors can be sourced from application status codes. In the figure below, status code histogram focused on error status code is depicted. We can spot the error trends together with identification of error sources.

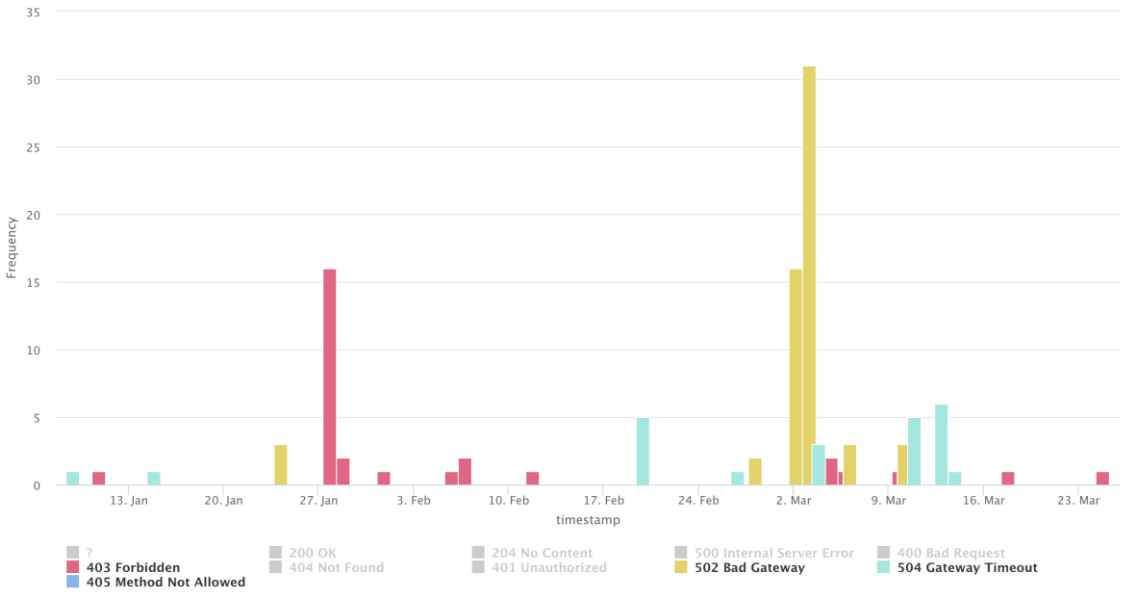

Figure 11. Status code histogram focused on error status codes

With application of EDA, the resulting anomalies are used in the creation of labeled dataset for anomaly detection purposes. The dataset can serve as a baseline for creating various analytical and machine learning anomaly detection models such as frequency threshold detection, supervised anomaly prediction, unsupervised anomaly detection, etc. In the Table 2, we present the final dataset statistical information.

Table 2. Dataset statistical information

Attribute name	Type	Missing	Least / Min	Most / Max	Range
timestamp	Date and time	0	*Jan 6, 2020 6:18 AM*	*Mar 26, 2020 9:06 PM*	80d 14h 48min
account_id	Nominal	3	*58710* (3)	*12345* (131,132)	*12345*, *c84c286[...]ffea5*, [52 more]
company_name	Nominal	3	*Company XYZ (3)*	*Company A (*131,132*)*	*Company A*, *Company B*, [52 more]
country	Nominal	3	*XX* (29)	*US* (399,465)	*US*, *BA*, *IN*, [12 more]
platform	Nominal	0	*Backend* (45%)	*Browser* (55%)	*Browser*, *Backend*
user	Nominal	6	*fk***@*.com* (4)	*fs***@*.com* (48,738)	*fs***@*.com*, *de***@*.com*, [209 more]
remote_address	Nominal	3	*184.*.*.22* (3)	*77.*.*.171* (41,561)	*77.*.*.171*, *144.*.*.229*, [302 more]
user_agent	Nominal	0	*Mozilla/[...]4.1* (3)	*Mozilla/[...]ri/537.36* (77,449)	*Mozilla/[...]36*, *Mozilla/[...].0*, [114 more]
error_msg	Nominal	467,225	*Getaddr[...].com* (1)	*ESOCKET[...]UT* (89)	*ESOCKET[...]UT*, *502*, [3 more]
level	Nominal	0	*error* (159)	*info* (467,225)	*Info*, *Error*
message	Nominal	0	*Integ[...]led* (159)	*Profiling* (264,851)	*Profiling*, *frontTimigs*, [1 more]
status_code	Nominal	93	*405 Method [...]ed* (1)	*200 OK* (453,461)	*200 OK*, *204 No Content*, [8 more]

resource_method	Nominal	0	*PUT* (97)	*GET* (373,123)	*GET, POST,* [3 more]
resource_base	Nominal	0	*produ[...]ile* (8)	*endpoints* (98,191)	*endpoints, customers,* [17 more]
user_domain	Nominal	6	*C**** (272)	*A**** (351,885)	*A***, M***,* [9 more]
user_agent_os	Nominal	0	*Unknown* (3)	*Windows* (411,762)	*Windows, OS X,* [2 more]
anomaly	Binominal	0	*True* (882)	*False* (466,502)	*False, True*

5. Conclusion

This study has shown that the use of Exploratory Data Analysis contributes to and complements the implementation of CRISP-DM methodology phases: business understanding, data understanding, and data preparation. Moreover, we demonstrate that Exploratory Data Analysis is efficient method for detecting anomalies in big data. Summarizing data characteristics and discovering underlying patterns for data and its distribution brings value for both data understanding and data preparation phase. We confirm the benefits of proven method from previous studies: consultations with SME play a crucial role in the business understanding phase and give a valuable contribution in data understanding phase Next, consultations in the data understanding and data preparation phase facilitates the workflow and can help us increase the data value.

Future efforts can be placed in implementation of subsequent CRISP-DM phases, that is, modeling, evaluation and deployment. Modeling data using Machine Learning techniques enables complex pattern discovery, as suitable for big data datasets, and further improves anomaly detection as underlying mathematical relationships can be leveraged. While this has been proven in majority of studies conducted in the field of anomaly detection and supervised machine learning, we propose a use of unsupervised machine learning for finding new anomalies that will enable a creation of extended labeled dataset - which can then be used for creation of supervised machine learning model for anomaly detection and prediction.

6. References

[1] "Big Data and cloud computing: innovation opportunities and challenges" [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/17538947.2016.1239771. [Accessed: 04-Sep-2020]

[2] "Cloud Security Alliance (CSA)" [Online]. Available: https://cloudsecurityalliance.org/. [Accessed: 04-Sep-2020]

[3] "Top Threats to Cloud Computing: Egregious." [Online]. Available: https://cloudsecurityalliance.org/artifacts/top-threats-to-cloud-computing-egregious-eleven/. [Accessed: 04-Sep-2020]

[4] "About AWS." [Online]. Available: https://aws.amazon.com/about-aws/. [Accessed: 04-Sep-2020]

[5] A. Sari, "A Review of Anomaly Detection Systems in Cloud Networks and Survey of Cloud Security Measures in Cloud Storage Applications," *Journal of Information Security*, vol. 6, no. 2, pp. 142–154, Mar. 2015.

[6] "Real-time big data processing for anomaly detection: A Survey," *Int. J. Inf. Manage.*, vol. 45, pp. 289–307, Apr. 2019.

[7] "Cyber Security: Threat Detection Model based on Machine learning Algorithm - IEEE Conference Publication." [Online]. Available: https://ieeexplore.ieee.org/document/8724096. [Accessed: 04-Sep-2020]

[8] "DMME: Data mining methodology for engineering applications – a holistic extension to the CRISP-DM model," *Procedia CIRP*, vol. 79, pp. 403–408, Jan. 2019.

[9] "A Reference Model for Big Data Analytics" [Online]. Available: https://www.researchgate.net/publication/327728739_A_Reference_Model_for_Big_Data_Analytics. [Accessed: 04-Sep-2020]

[10] "Exploratory data analysis" [Online]. Available: https://psycnet.apa.org/record/2011-23865-003. [Accessed: 04-Sep-2020]

[11] "Open Source Search: The Creators of Elasticsearch, ELK Stack & Kibana." [Online]. Available: https://www.elastic.co/. [Accessed: 04-Sep-2020]

[12] "Kibana." [Online]. Available: https://www.elastic.co/kibana. [Accessed: 04-Sep-2020]

[13] "Logstash." [Online]. Available: https://www.elastic.co/logstash. [Accessed: 04-Sep-2020]

[14] "RapidMiner." [Online]. Available: https://rapidminer.com/. [Accessed: 04-Sep-2020]