Artificial intelligence and big data driven IS security management solution with applications in higher education organizations

Vladislavs Minkevics
Riga Technical University
Kalku 1, Riga, Latvia
Vladislavs.Minkevics@rtu.lv

Janis Kampars
Riga Technical University
Kalku 1, Riga, Latvia
Janis.Kampars@rtu.lv

Abstract — **This paper presents the architecture of a modular big-data-based information system (IS) security management system (ISMS) and elaborates one of its modules – artificial intelligence driven NetFlow data analysis (NFAI) module. The ISMS is used in production at Riga Technical University and can be adapted for use in other organizations. The proposed platform is based on mostly free and open-source tools and allows to prevent or minimize the consequences of malware's activity with little impact on the employee's privacy. The presented NFAI detection module provides detection of malware activity by extracting features from NetFlow data within a 10-minute interval and feeding it into several trained classifiers. ISMS does not rely solely on NFAI module alone, it uses an ensemble of modules and algorithms to increase the accuracy of the malware detection. The presented IS security management system can be employed in real-time environment and its NFAI detection module allows to identify an infected device as soon as it starts to communicate with the botnet (a logical collection of Internet-connected devices such as computers, smartphones or IoT devices whose security have been breached and control ceded to a third party) command and control centre to obtain new commands. The presented NFAI module has been validated in the production environment and identified infected devices which were not detected by antivirus software nor by firewall or Intrusion Detection System.**

Keywords — IS security, big data, malware, NetFlow, artificial intelligence.

I. INTRODUCTION

Nowadays, securing information systems has become a challenge like never before. Failing in this endeavour may lead to serious consequences. Lately many security breaches went viral, like Solarwinds attack [1] and Microsoft Exchange security flaws [2]. Consequences of such attacks may also affect public authorities, even the police [3]. Usually these consequences are the result of not paying attention to patches released by vendors [4], but in case of Solarwinds there is another possible reason – built in password [5]. The problem with security nowadays, apart of visible security breaches, is invisible attacks and data exfiltrations usually done by botnet members. According to report [6] Mirai botnet [7], drove the largest DDoS attack ever recorded in 2016 on the French hosting company OVH peaking at 1Tbps and mostly targeting IoT devices, like home routers and IP cameras. Mirai distributed denial of service (DDoS) worm remained an active threat and, with 16% of the attacks, was the third most common IoT threat in 2018. According to the same report [6], malicious URLs are growing by 3.4% a year.

Based on Accenture 2020 Cyber Threatscape Report [8], senior decision makers should be kept abreast of the rapid and constant evolution in adversary tradecraft to support network defenders with the resources and business and technical mitigations required to adapt and stay ahead. High number of unpatched vulnerabilities is one of the most important problems, especially in environments where different operating systems are used. According to this report, malicious actors are using off-shelf tools and exploits to penetrate networks and exfiltrate data. Therefore, identification of such activity by means of automation is a vital element in today's information security management field.

In this paper, we are focusing on methods to detect botnet using supervised machine learning algorithms widely used in previous studies. The focus of this article is on the artificial intelligence (AI) driven Netflow data analysis (NFAI) module. Module extracts significant NetFlow features and uses machine learning algorithms to detect malware. Current research often is limited to analysis of synthetic data, which have significantly different properties than real data. For this research we use real data from the production environment in Riga Technical University (RTU). The input data for the NFAI originates from the devices used by RTU students, employees, and guest researchers. This provides a realistic evaluation of the proposed NFAI module.

The importance of the NFAI is motivated by the fact that modern malware is trying to hide itself by using SSL encryption, commonly used ports and by employing other smart evasion techniques, especially in the case of botnets. The NFAI module identifies infected devices based on their abnormal behaviour patterns. Our experiments show that this approach is effective even when common ports and encryption are used by the malware. Usually, botnet originated traffic differs from normal traffic in size, number of packets sent, and sent/receive ratio, which allows to identify the infected devices.

This work is structured as follows. Section II provides a review of the related work in the field of botnet detection in network traffic. Section III presents the IS security management approach in Riga Technical University and further motivates the development of the NFAI module. Section IV develops and evaluates the NFAI module in the production environment. Section V concludes and provides directions for future research.

II. RELATED WORK

There are several studies concentrating on the IP Flow analysis. IP Flow data is valuable because it provides information about connections, TCP/IP flags, and data amount sent and received, which provides enough information to identify suspicious activity of possibly compromised devices.

978-3-903176-36-2 ©2021 IFIP

Farooq et al. [9] state that organizations have now realized that traditional monitoring complemented by Machine Learning (ML) based Threat Hunting will be a necessary part of any IS security monitoring system. Furthermore, ML based analytics can learn normal behavioural baselines, which leads to a smaller number of false-positive security alerts. In addition, ML analytics are best suited to analyze large volumes of security events and feed deviations from normal baselines into proactive threat hunting processes as indicators or leads of potential malicious activity. Authors suggest using Semi-supervised (one-class classification) algorithms for cyber security monitoring purposes.

Ganame et al. [10] state that Signature-based malware detection systems struggles with the identification of zero-day malware, as by design such systems rely on known malware patterns. Researchers evaluated AlienVault, McAfee and FireEye approach to detect unknown malware and stated that most of them perform poorly on unknown malware. In contrast, behavioural systems can potentially detect such malware which has led to organizations realizing that effective and versatile ML based Threat Hunting will be a necessary part of any IS security monitoring portfolio. Authors also stated that unlike sandboxing solutions, the ML behaviour system does not need to extract the binaries of malware to detect an infection. The behavioural analysis of the network flow using ML is sufficient for it to make decisions.

According to Jirsik et al. [11], a holistic view of the network (macro view) is provided by the traditional network monitoring applications. However, to develop a comprehensible overview of the network, more in-depth information (micro view) is needed, such as information about individual hosts and their actions. The combination of macro and micro views gives security analysts the ability to observe the overall status, as well as the status of any specific elements in the network, and thus to acquire an in-depth comprehension of the network. The authors proposed a stream-based network data analysis approach, where the IP flows are processed and analysed in data streams immediately after an IP flow is observed. Authors stated that it aids traditional monitoring with the ability to run analytical queries that are evaluated in real time ensuring high throughput, low latency and good scalability. They demonstrate the possibilities of the presented workflow for real-time analysis of IP flows. Apache Spark was selected as the data stream processing framework for its high IP flow data throughput.

A research by Mousavi et al. [12] proposed a fully scalable big data framework which is based on the Hadoop platform and enables scaling each individual botnet detection component. They stated that the framework can be used with any botnet detection method - including statistical methods, machine learning methods, and graph-based methods. This is achieved by using Apache Kafka, Apache Spark and Nprobe. The highest achieved detection rate is 72% with a false positive rate of 2%. The proposed framework was implemented in a large ISP network with near 5Gbps traffic and experimental results confirm the scalability and usability of the proposed framework.

Research done by Pacheco et al. [13] unveiled several challenges in the traffic classification field. For instance, working with encrypted traffic was challenging since the selected model features should be available even in the absence of the packet contentment. Authors stated that the current publicly labelled data is scarce, which complicates comparing ML solutions. Researchers concluded that knowledge extraction with ML can help finding new or anomalous behaviour in the internet traffic. Furthermore, the implementation of such ML solutions remains an important task to achieve, due to different factors, mostly related to performance and adaptability of the chosen solutions. They also proposed using a dynamic feature selection to create adaptive models that use the most suitable features for specific objectives.

Lashkari et al. [14] stated that most of their analysed research have employed machine learning techniques in their approach for botnet detection. This popularity indicates that ML is the primary technique that modern botnet detection techniques are using to detect botnets.

Researchers also stated that many types of botnet topologies exist and each of these topologies needs a specific approach for detection.

It can be concluded that nowadays analysis of encrypted data can be achieved only by means of extracting metadata. This metadata can be adapted and fed into a machine learning algorithm for classification. All reviewed articles stated that ML must be employed to achieve cybersecurity goals. Available research articles are often limited to analysis of synthetic data, which have significantly different properties than real data that we use in our research.

III. BACKGROUND

The development of Security Operation Center (SOC) type IS security management platform (ISSMP) in RTU started in 2014. Initially, it consisted of Suricata IDS [15] and Python scripts that helped the Chief security officer (CSO) to analyse IDS generated data. New Big Data based system has been built for security purposes to implement proactive and intelligent security measures.

Based on the best practices and industrial experience of managing large university network, a horizontally scalable and AI driven IS security management platform was designed and implemented, leading to the developments covered within this paper. The ISMS platform consists of preventive and detective capabilities and contains a vulnerability management component, which performs regular automated vulnerability scans and analytical report preparation for the responsible personnel: specific faculty's administrator and CSO.

ISMS is based on open-source products and is capable of handling IS security incidents in an efficient manner with respect to the privacy according to the EU General Data protection regulation [16]. The platform contains analysis and actions modules, which provide SOC capabilities (see Fig.1.).

Technology wise ISMS relies on well-established big data platforms like Apache Spark [17] and Apache Kafka [18].

The privacy goals of the ISMS are reached by avoiding linking the IP address with a user prior to detecting a suspicious activity. While constant lookups of user IP address or all users would reduce the latency of incident analysis, it has a negative effect on the privacy. Authors believe that the increase of the incident detection latency by a few seconds due to late IP-user lookup can be neglected in the light of significantly increased privacy.

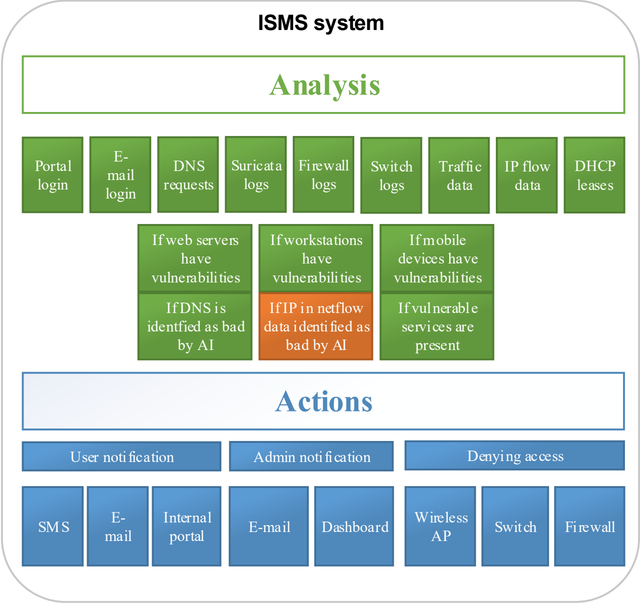

Fig. 1: Architecture of ISMS system.

Our previous work [19] describes ISMS elements in detail.

IV. NFAI MODULE IMPLEMENTATION AND EVALUATION

Development of the NFAI module includes following steps:

1) preparation of training dataset,
2) feature and classifier selection based on the results of related research and our experiments,
3) dimensionality reduction,
4) model evaluation in real-time production environment.

Training data was collected in RTU production environment for 3 months (from 28.09.2020-25.12.2020) by gathering NetFlow data. To collect data, we used fprobe tool [20] which collects network traffic data and emits it as NetFlow flows towards the specified collector nfcapd which is the part of nfdump [21] tool. Data preparation included:

1) nfdump tool with options “fmt:%ts %td %pr %sap %dap %flg %ipkt %ibyt %opkt %obyt %fl” to extract data in 10 minute interval and save result into text file.
2) Text file processing with Apache Spark [18] using python code:
   a. Creation of data frame from text file;
   b. Operations with data frame using embedded SQL functionality to extract all necessary features for further analysis (Table 1);
   c. After Apache Spark data processing, ready to use json file with data.

Prepared data was divided into legitimate and malicious traffic based on the information provided by the firewall, intrusion detection system (IDS) and CSO interviews. Additionally, 10-minute data intervals were retrieved from the Apache Kafka topics up to 24 hours prior actual identification of device malicious activity by IDS or firewall. The collected data intervals were reviewed by the CSO and marked accordingly. In total, the training dataset consisted of 290 malicious and 4548 known good 10-minute intervals of NetFlow data.

The initial list of features is largely based on [22] and is given in Table 1.

TABLE 1. FEATURES USED TO DETECT MALICIOUS ACTIVITY

Feat. No.	Description	Example
F1	Total number of packets transferred by specific source address	3900.00
F2	Minimal number of packets per flow transferred by source address	2.00
F3	Maximal number of packets per flow transferred by source address	2701.00
F4	Average number of packets transferred by source address	36.79
F5	Packet variance divided by 100000 transferred by source address	0.68
F6	Total number of bytes transferred by source address	692791.00
F7	Minimal number of bytes per flow transferred by source address	80.00
F8	Maximal number of bytes per flow transferred by source address	426160.00
F9	Average number of bytes transferred by source address	6535.76
F10	Variance of bytes transferred by source address	16996.52
F11	Total number of flows transferred by source address	106.00
F12	Total number of packets transferred to specific source address	4680.00
F13	Minimal number of packets per flow transferred to source address	1.00
F14	Maximal number of packets per flow transferred to source address	3332.00
F15	Average number of packets transferred to source address	38.68
F16	Packet variance divided by 100000 transferred to source address	0.91
F17	Total number of bytes transferred to source address	3108903.00
F18	Minimal number of bytes per flow transferred to source address	40.00
F19	Maximal number of bytes per flow transferred to source address	2677172.00
F20	Average number of bytes transferred to source address	25693.41
F21	Variance of bytes transferred to source address	590984.36
F22	Total number of flows transferred to source address	121.00
F23	Total number of flows	227.00
F24	Percentage of flows transferred by source address	0.47
F25	Percentage of flows transferred to source address	0.53
F26	Number of unique IP addresses that source address connected to	27.00
F27	Number of unique IP addresses that connected to source address	29.00

F28	Number of distinct source ports used by source address	97.00
F29	Number of distinct destination ports used by source address	10.00
F30	Percentage of source ports higher than 1024 used by source address	1.00
F31	Percentage of source ports lower than 1024 used by source address	0.00
F32	Percentage of destination ports higher than 1024 used by source address	0.06
F33	Percentage of destination ports lower than 1024 used by source address	0.94
F34	Percentage of UDP protocol used by source address	0.23
F35	Percentage of TCP protocol used by source address	0.77
F36	Percentage of ICMP protocol used by source address	0.00
F37	Percentage of UDP protocol used to connect to source address	0.22
F38	Percentage of TCP protocol used to connect to source address	0.78
F39	Percentage of ICMP protocol used to connect to source address	0.00

Four classifiers were selected for NFAI: random forest (RFC), decision tree (DTC), neural network (NNC), and k-nearest neighbour (KNN). For every training set, 10fold cross validation was performed using *the sklearn* Python library.

Initial performance (see Table 2) of the classifiers was evaluated using all features from Table 1.

TABLE 2. PERFORMANCE MEASURES

Classifier	Precision	Recall	F1 score	Accuracy
DTC	0.941	0.946	0.936	0.992
RFC	0.964	0.938	0.950	0.994
NNC	0.745	0.312	0.386	0.953
KNN	0.683	0.642	0.660	0.961

The results show that initially NNC and KNN classifiers are performing poorly, while DTC and RFC already provide relatively good results. Further dimensionality reduction was performed to determine the optimal set of features and improve the performance of the classifiers. For this purpose, extra-trees classifier from *sklearn* Python library was used. It implements a meta-estimator that fits several randomized decision trees based on the training dataset. The produced model provides feature importance information which can be further used for dimensionality reduction purposes. Feature importance threshold was selected in the range of [0.008-0.026]. For each selected threshold, feature importance was re-evaluated 100 times. Figure 2 shows the correlation between the threshold and different feature set variants provided by the extra trees classifier. To evaluate importance of features *feature_importances* class was used. Next 10-fold cross validation was performed using python *sklearn* library. The best performing feature set was chosen based on F1 Score, which according to Koo Ping Shung [24], is a good measure if a balance between precision and recall is needed. Figure 2 shows the highest mean values of 10-fold F1 Score for each classifier with most effective feature set determined by the extra-trees classifier.

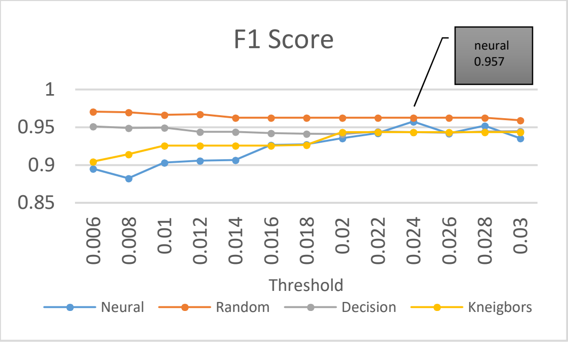

Fig. 2. Highest 10 fold F1 Score for each threshold

The final feature set consisted of the following features: *“F11,F24,F25,F26,F27,F28,F29,F32,F33,F34,F35,F38,F39”*

The result of 10fold cross validation for the chosen feature set is shown in Table 3.

TABLE 3. PERFORMANCE MEASURES OF ALL CLASSIFIERS FOR THE CHOSEN FEATURE SET

Classifier	Precision	Recall	F1 score	Accuracy
DTC	0.947	0.941	0.938	0.993
RFC	0.974	0.95	0.962	0.995
NNC	0.960	0.952	0.951	0.994
KNN	0.980	0.908	0.942	0.993

Table 3 shows that random forest and neural network are the most suitable classifiers with the selected feature set. To confirm that models are not overfitted and are able to perform well in a production environment with unseen datasets and possibly new kinds of botnet infections, all classifiers were evaluated in RTU production environment from February until April of year 2021.

TABLE 4. PRACTICAL RESULTS OF THE IMPLEMENTED NFAI MODULE

NFAI detection module performance for NATted IP	DTC (%)	RFC (%)	NNC (%)	KNN (%)
False Negative (NAT)	5.5	15.7	11.1	11.1
True Positive (NAT)	94.5	84.3	88.9	88.9
False Negative (PUBLIC)	43.2	47.7	45.4	45.4
True Positive (PUBLIC)	56.8	52.3	54.6	54.6

The results (see Table 4) show that the NFAI module can be successfully adapted for real-time NetFlow data analysis and DTC performed better over other classifiers used in the experiment for both NATted (network address translated) and public IP addresses. We gained these results by requesting feedback from users of possibly infected devices. Unfortunately, IP addresses available from internet (e.g., publicly accessible resources such as web servers) showed much worse results. This can be explained by the fact that NetFlow data originating from the NATted workstations was used for training set preparation. The other explanation is that public IP addresses are prone to port scanning which creates noise for classifiers. We calculated performance of all classifiers on 8 core workstation with 32GB of RAM: RFC - 2 minutes for decision making, KNN - 25 seconds, NNC - 7 seconds and DTC - 2 seconds.

V. CONCLUSIONS AND FUTURE WORK

The proposed NFAI module enabled to identify malware activity in the network. Nowadays NetFlow data is valuable source for improving overall IS security of organization. By analysing NetFlow data in real environment new opportunities were provided for the ISMS. NFAI module's adoption and development of new malware behaviour datasets will further improve the performance of ISMS.

By introduction of the new NFAI module into ISMS:

1) The overall rate of false positives dropped.
2) Malware identification rate increased.
3) Malware unknown to security vendors was discovered. The discovered malware are mostly members of different botnets.

The results of our work show that different classifiers can be adapted to identify patterns of malicious activity in NetFlow data. In addition, we would suggest using a dynamic feature set selection as stated in [13] and use at least two different classifiers to detect malicious activity in NetFlow data: neural networks and decision trees classifier. By adapting ensemble learning methods more specialized threat detection modules can be developed and integrated into the platform for increasing its accuracy of malware detection. The level of the ISMS accuracy was raised by requesting feedback from users whose devices identified as infected. The results showed that students are very prone to giving feedback.

Further we are planning to introduce different automated actions, based on the identified risk level. Low risk alerts could be only informative, for example, if user has unwanted software installed (e.g., click gatherers, redirectors), while high risk alerts could be acted upon immediately. The platform discussed in this paper can be adapted by any organisation facing similar challenges.

Since the adoption of the NFAI module into ISMS platform, in 3 months' time we determined 6 unknown malware infections and unwanted software installations, which would remain undetected without the NFAI module. We will continue to further expand the ISMS platform by adding different modules, based on current threat level in IS security landscape.

REFERENCES

[1] S. Adler, "IOTW: As The SolarWinds Hack Investigation Continues, New Insights Reveal A New Suspect," CyebrSecurityHub, 15 01 2021. [Online]. Available: https://www.cshub.com/attacks/articles/iotw-as-the-solarwinds-hack-investigation-continues-new-insights-reveal-a-new-suspect. [Accessed 13 05 2021].

[2] L. Tung, "Microsoft: These Exchange Server zero-day flaws are being used by hackers, so update now," ZDNet, 03 03 2021. [Online]. Available: https://www.zdnet.com/article/update-immediately-microsoft-rushes-out-patches-for-exchange-server-zero-day-attacks/. [Accessed 13 05 2021].

[3] L. Morgan, "IOTW: DC Police Department Hit with Ransomware; Hackers "Quit"," CyberSecurityHub, 07 05 2021. [Online]. Available: https://www.cshub.com/attacks/articles/iotw-dc-police-department-hit-with-ransomware-hackers-quit. [Accessed 13 05 2021].

[4] P. Muncaster, "Microsoft Patches Four Zero-Day Exchange Server Bugs," Infosecurity Magazine, 03 03 2021. [Online]. Available: https://www.infosecurity-magazine.com/news/microsoft-patch-four-zeroday/. [Accessed 13 05 2021].

[5] B. F. a. G. Sands, "Former SolarWinds CEO blames intern for 'solarwinds123' password leak," CNN, 21 02 2021. [Online]. Available: https://edition.cnn.com/2021/02/26/politics/solarwinds123-password-intern/index.html. [Accessed 13 05 2021].

[6] "Internet security threat report vol24," Symantec, 2019. [Online]. Available: https://docs.broadcom.com/doc/istr-24-2019-en. [Accessed 13 05 2021].

[7] T. Foltýn, "GitHub knocked briefly offline by biggest DDoS attack ever," Eset, 2018. [Online]. Available: https://www.welivesecurity.com/2018/03/02/github-knocked-briefly-offline-biggest-ddos-attack/. [Accessed 13 05 2021].

[8] Accenture, "Accenture Cyber Threatscape Report 2020," 24 11 2020. [Online]. Available: https://www.accenture.com/us-en/insights/security/cyber-threatscape-report.

[9] H. M. Farooq and N. M. Otaibi, "Optimal Machine Learning Algorithms for Cyber Threat Detection," 2018.

[10] A. M. Z. G. B. O. Ganame K., *Network Behavioral Analysis for Zero-Day Malware Detection – A Case Study.,* Vancouver, BC, Canada: Springer, Cham., 2017.

[11] T. Jirsik, M. Cermak, D. Tovarnak and P. Celeda, "Toward Stream-Based IP Flow Analysis," *IEEE Communications Magazine,* vol. 55, no. 7, pp. 70-76, 2017.

[12] S. H. Mousavi, M. Khansari and R. Rahmani, *A fully scalable big data framework for Botnet detection based on network traffic analysis,* 2020.

[13] F. Pacheco, E. Exposito, M. Gineste, C. Baudoin and J. Aguilar, "Towards the Deployment of Machine Learning Solutions in Network Traffic Classification: A Systematic Survey," 2019.

[14] A. H. Lashkari, G. D. Gil, J. E. Keenan, K. F. Mbah and A. A. Ghorbani, *A Survey Leading to a New Evaluation Framework for Network-Based Botnet Detection,* New York, NY, USA: Association for Computing Machinery, 2017.

[15] Suricata, "Suricata," 28 01 2020. [Online]. Available: https://suricata-ids.org/. [Accessed 22 01 2020].

[16] EU, "General Data protection regulation," 27 04 2016. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1532348683434&uri=CELEX:02016R0679-20160504. [Accessed 24 01 2021].

[17] "Apache Spark," 24 01 2021. [Online]. Available: http://spark.apache.org/downloads.html.

[18] A. Kafka, "Apache Kafka," Apache Kafka, 24 02 2021. [Online]. Available: https://kafka.apache.org/. [Accessed 10 02 2021].

[19] V. a. K. J. Minkevics, "Methods, Models and Techniques to Improve Information System's Security in Large Organizations," in *n Proceedings of the 22nd International Conference on Enterprise Information Systems - Volume 1: ICEIS*, 2020.

[20] "fprobe," SourceForge, 2016. [Online]. Available: https://sourceforge.net/p/fprobe/wiki/Home/. [Accessed 13 05 2021].

[21] "nfdump," GitHub, 2021. [Online]. Available: https://github.com/phaag/nfdump. [Accessed 13 05 2021].

[22] M. Sheng, Machine Learning for Computer and Cyber Security Principles, Algorithms, and Practices, New York: CRC Press Taylor & Francis Group, 2019.

[23] K. P. Shung, "Accuracy, Precision, Recall or F1?," 15 05 2015. [Online]. Available: https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9. [Accessed 25 03 2021].