Improvement of Implemented Infrastructure for Streaming Outlier Detection in Big Data with ELK Stack

Zirije Hasani[1(✉)] and Jakup Fondaj[2(✉)]

[1] Faculty of Computer Science, University of Prizren "Ukshin Hoti", Prizren, Kosovo
zirije.hasani@uni-prizren.com
[2] Faculty of Computer Science and Technologies, South East European University, Tetovo, Macedonia
j_fondaj@seeu.edu.mk

Abstract. Nowadays the usage of internet is constantly increasing the amount of data. As a result the need for analyzing this data has recently emerged as we need to face a new phenomena known as the Big Data. This research is focused in finding appropriate architecture for real-time big data analytics and its main task is to detect anomalies in this real-time data. There are some tools that are used and analyzed by us in order to find the best one, but in this paper we use Timeline and compare it with Fluentd which is the tool we used in previous research [12]. Here we are going to show the reasons why Timelion is better than Fluentd. Anomaly detection in real-time big data is a problem that faces many organizations and it is a challenge for researchers as well. Our research deals with developing infrastructure for monitoring e-dnevnik (education national system in Macedonia) application server and to detect errors in order to scale up the performance. In order to enable this infrastructure to detect anomalies in streaming data we implement different algorithms for anomaly detection in Timelion. Another important thing is to know how to visualize the results. In this paper, we show the visualization of an e-dnevnik log by using Logstash, Elasticsearch, Kibana, and also how Timelion helps us to identify anomalies in real time.

Keywords: Log data · Anomaly detection · World Wide Web
Real-time big data · Timelion · Visualization · Kibana · CSV log data
Fluentd · Logstash · Elasticsearch

1 Introduction

It is not an easy task to develop an infrastructure for real-time big data. There are not many studies in this area even less is the amount of research which deals with anomaly detection in real-time big data [1]. An outlier is a data point which is significantly different from other data. In data mining, an outlier is also referred to as 'anomalies, deviation or abnormalities' [1].

Streaming outlier detection scenario arises in the context of many applications such as sensor data, mechanical system diagnosis, medical data, network intrusion data,

Á. Rocha et al. (Eds.): WorldCIST'18 2018, AISC 746, pp. 869–877, 2018.
https://doi.org/10.1007/978-3-319-77712-2_82

newswire text posts or financial posts. In this research, for experimental work, we used network data that are data generated by monitoring the system. Some of this data are process monitoring, memory monitoring, system monitoring, input-output monitoring and monitoring of CPU. The idea is to implement an infrastructure which will be capable in detecting anomalies and at the same time enables to monitor the system in real time for this particular data.

The main components and their role of the infrastructure (Timelion, Logstash, Elasticsearch, and Kibana) are explained in this paper.

In order to define which infrastructure to implement, we have conducted a research for applications that are used for anomaly detection in real time big logs. The comparison of tools shows that Timelion [8, 9] is the best free application. This application enables us to add new functions in order to improve the process of anomaly detection.

In Sect. 2 we show related work about this topic. In Sect. 3, we make a comparison between the Fluentd tool (which we used in our previous work [12]) and Timelion, the new tool proposed in this work. In Sect. 4 we explain the proposed infrastructure, the components of this infrastructure and their role. In Sect. 5, we show how all these components are configured and also we have added filters to make pre-processing of data in order for the data to be loaded properly in Logstash. In Sect. 6, we show how we can visualize the e-dnevnik log data in real time by using the implemented architecture.

The main part of the paper shows the implementation of outlier detection algorithms in Timelion and the output that is given. Implementation of the Holt for anomaly detection and the result is visualized with Timelion/Kibana.

Finally, it is concluded that Timelion is the best tool to use for anomaly detection.

It has free access and we can implement different algorithms with an easy to use programming language.

2 Related Work

This paper is the result of four-year research in the field of Real-time Big Data analytics. During this time a number of papers were published as well [2–6, 10, 12].

There are many tools compared for anomaly detection in real-time big data. The idea was to find a tool which is open source where we can give our contribution. From previous and this research, we conclude that Timelion is the best open source tool for real-time anomaly detection in big data.

Analyzing big data was the main idea of our past research. We started by batch analytics of big data with Hadoop, but then we needed to analyze streaming data that came in real time and also visualize them. As a result of this need, an infrastructure is proposed [10] where preprocessing of Big data is done in real time by Logstash before they are saved in Elasticsearch. The visualization of the result is done by Kibana. By this proposed infrastructure we analyze the data from e-dnevnik[1] where the log files are SQL queries. The idea was to remove the unnecessary data, like comments, null values etc., so that the statistics taken from the data are more realistic. The above mentioned infrastructure did not have the option how to detect anomalies in the data. For this reason,

[1] http://ednevnik.edu.mk/.

we propose additional infrastructure where many of the components are removed from the first proposed infrastructure [10, 12] and here Timelion is added as a new component. Timelion has a large number of commands which allows us to manipulate with data. There is also the possibility to detect anomalies in real time big log data. In paper [12] we have proposed this infrastructure, but the Fluentd component that we proposed has been now removed from the infrastructure. The reason for this is given in the next section. There are no research published related to this topic.

3 Comparison Between Anomaly Detection Tools

In order to detect anomalies in real-time big data, we need to find appropriate tools for this purpose.

In our continuous work, we have analysed two main tools used for anomaly detection in real-time big data, Fluentd and Timelion. We have evaluated both tools and with Fluentd we have published the work in [12]. In this paper we are going to present another tool named Timelion and also make a comparison between these tools in order to conclude which one is better. There is some comparison [15] for big data tools but not for the tools which we proposed.

To compare these tools we have taken into consideration some characteristics such as whether they support streaming data, is it possible to implement our algorithms, are they easy to configure etc. In Table 1 below we have conducted a comparison of these tools based on different characteristics.

Table 1. Comparison between Timelion and Fluentd

Characteristics	Fluentd	Timelion
Streaming data	Yes	Yes
Implement our algorithms	No	Yes
Easy configuration	No	Yes
Visualization	No	Yes
Anomaly detection visualization	No	Yes
Open source	Yes	Yes
Log Management	Yes	Yes
User-friendly dashboard	No	Yes
The ability to scale to hundreds of servers	Yes	Yes
Full-text search	No	Yes
Powerful log analysis	Yes	Yes
The real-time reporting system	No	Yes
Create charts	No	Yes
Send reports and be alerted when something happens	No	Yes

To compare these tools, they are both added to our proposed infrastructure for anomaly detection in real-time big data and they are also tested on real-time data. From

the table above we can see that Timelion is better than Fluentd. It offers a possibility to do programming of our proposed methods for anomaly detection.

4 Infrastructure for Outlier Detection in Real Time Big Data

With the aim being to deal with anomaly detection in big log files in real time produced by e-dnevnik, we start with the solution proposed by Kiyoto Tamura [6]. Since Elasticsearch along with Timelion [9] has evolved during the past several years, we include in our architecture several new components and remove several other unnecessary ones. Our proposed architecture is shown in Fig. 1. This architectural design is based on several phases. The first phase is the input phase where the e-dnevnik data is collected and then put into Logstash. In Logstash we make an input, filtering and an output. After that, Logstash has a plugin which makes output to Elasticsearch, this is where the data is stored. The visualization is done by Kibana and anomaly detection is done with Timelion. In Timelion, the anomaly detection algorithm is implemented as well.

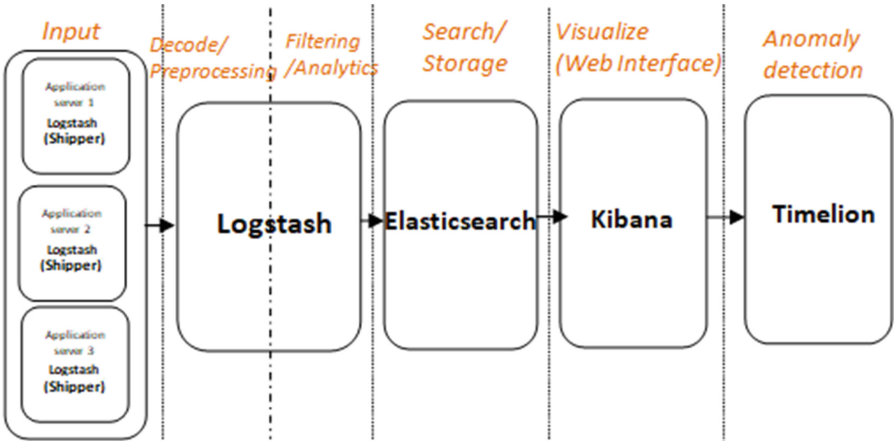

Fig. 1. Infrastructure for real-time anomaly detection in Big Data

The main component of the infrastructure is Logstash [4]. It is written in JRuby and runs in a Java Virtual Machine (JVM) [4]. It is easy to deploy as a single JAR file that can be started directly using a JAVA SE VM (no Apache Tomcat Containers are needed). Its architecture is simple compared to other similar software architectures since it consists of a three-phase pipeline (input, filter, and output) and it provides an easy way of extension of functionalities in each phase using plugins.

The Input phase collects the logs and sends the collected events to the filter phase. Logs can generally arrive from various sources: Files, TCP/UDP files, Syslog, Microsoft Windows EventLogs, STDIN, Key-value stores and a variety of others. In our case log file includes Postgres SQL CSV log files and Key-value stores.

Logstash comprises a large collection of filters which enable us to extract structured data into variables, parse, modify and enrich the data before they are pushed to the Elasticsearch.

The other main component of this infrastructure is Timelion. In Timelion we may describe queries, make a different transformation of data, implement statistical methods as well as visualize the data in order to learn from them [9].

Elasticsearch enables efficient indexing and storing of the event logs enabling a full-text search on them. It is an open-source distributed search engine library built on top of Apache Lucene [9]. Elasticsearch [6] allows us to implement store, index and search functionality, as such helps us in easier and more efficient computation of various data analytics. ElasticSearch is a NoSQL data store where data are stored as documents. Although it is mainly used by Java applications, the important thing is that applications need not be written in Java in order to work with ElasticSearch, since it can send and receive data over HTTP in JSON to index, search, and manage our Elasticsearch cluster.

The last component is Kibana [6]. This is an HTML/JS frontend web interface to Elasticsearch for viewing the log data. The beauty of Kibana is that we can easily search the data with different queries, produce charts, histograms, and other visual products.

5 Visualizing the CSV Log Data with Kibana

The infrastructure presented in this paper can be used for anomaly detection in real time Big Data. After configuring Logstash, the data is able to be visualized in Kibana. With Elasticsearch and Kibana we have the possibility to visualize the log data [11] from our e-dnevnik application servers. In the next picture, we show how the result is visualized by Kibana where we have the possibility to draw our own charts, histograms etc. Kibana offers many possibilities for data analysis and visualization. Some of their main functions are shown in Figs. 2, 3 and 4.

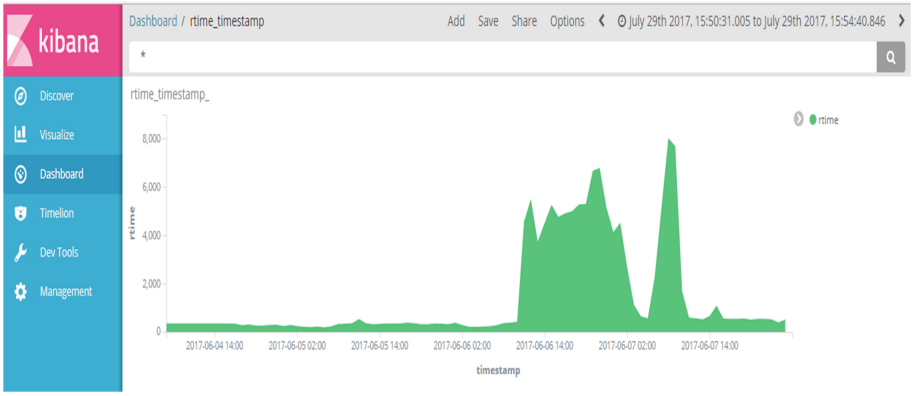

Fig. 2. Visualization of csv e-dnevnik data in real time by Kibana

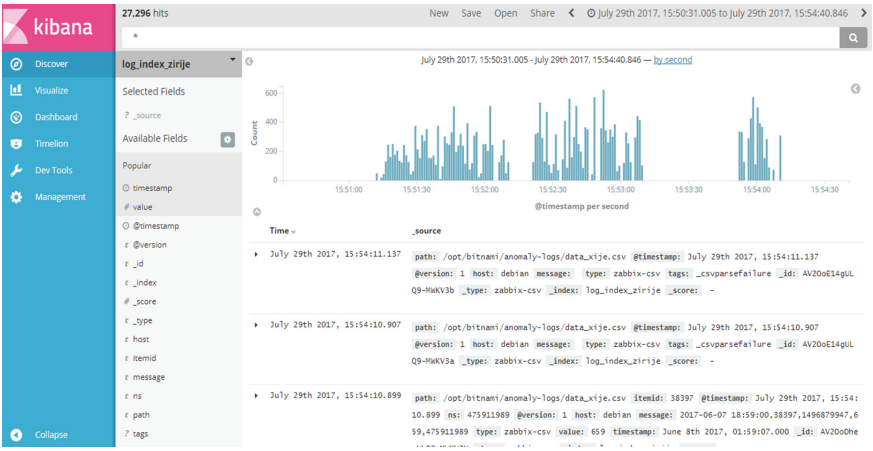

Fig. 3. CSV messages in Kibana

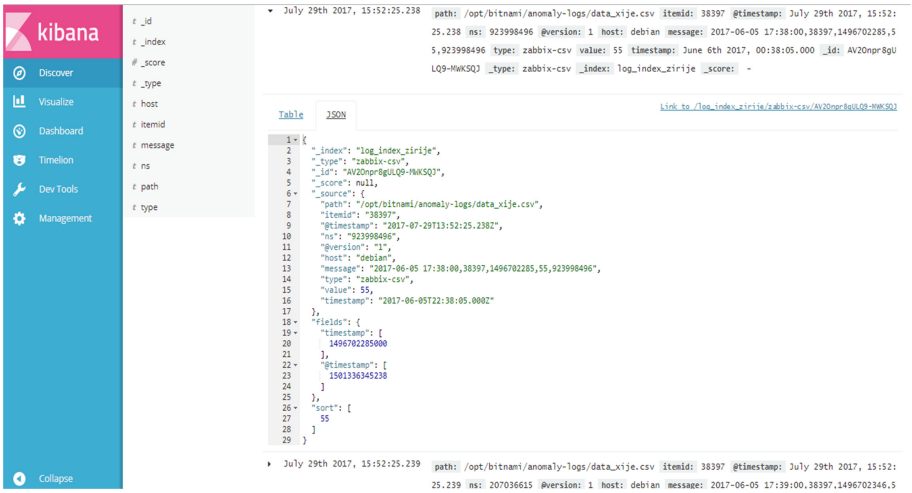

Fig. 4. e-dnevnik csv data shown in JSON format in Kibana

The result is shown in a period of time of three days. We can see here the different time periods for the number of requests in the chart. We can also see the content of the log file for every event that happens in our application server. The interesting thing about Kibana is that the data can be read and exported in JSON format as shown in Fig. 4 below.

Kibana produces different attributes for the data and we can filter the results based on the attributes we prefer. In Fig. 3 we have shown the attributes that are produced for CSV data.

6 Anomaly Detection with Timelion

As it is defined above, this research work is related to the processes of detecting anomalies in big data that are generated in real time. Anomaly in our case in considered if for example the number of request is increased in period of time from 19:00 pm to 06:00 am and also if the request are in not working days as Saturday and Sunday. The Infrastructure as input receives the CSV log data from application servers of e-dnevnik.

As part of the infrastructure, we have added Timelion, which enables us to program different algorithms which are analysed [6] for anomaly detection in real-time big data. The following picture shows the results from implementation of Triple Exponential Smoothing, also known as the Holt-Winters method [13] (Fig. 5).

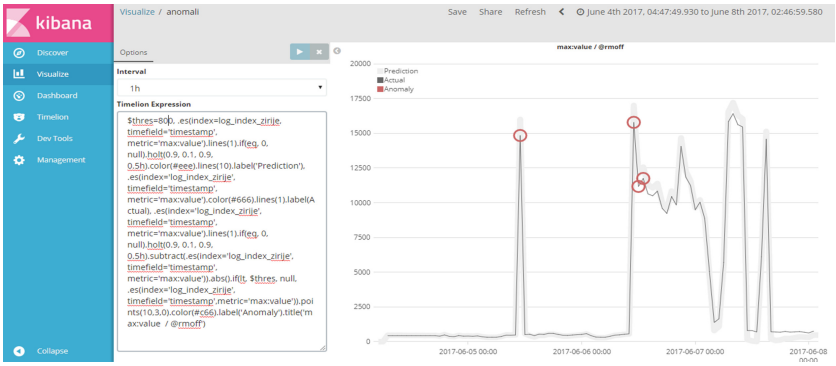

Fig. 5. Anomaly detection in Timelion

From the results we can see the anomalies which are pointed with red circles. These anomalies were previously known by researchers and here we can prove that the algorithm works well. This visualization of anomaly detection cannot be done in Fluentd because they don't offer that function. Based on the figure we get we may go further with analysis to find the reason why the anomalies were happening in that time period. The number of anomalies depends on the threshold we define, in our case the smaller the threshold the larger number of anomalies appears.

7 Conclusion

From this research work, we can conclude that as a result of our previous paper published on [12] as well as the work done here, we were able to do a comparison between these two proposed tools, Timelion and Fluend. At the same time we are able to define which one is more appropriate for our needs. The testing is done in real time data that come from e-dnevnik application. From all the comparison that is made in this paper it is clear that Timelion is better than Fluentd.

The other conclusion from this work is that we add a tool (Timelion) in our architecture. This tool enables us to implement our proposed algorithms [6] and also visualize the result. This was not previously possible with Fluentd.

Visualization of anomaly detection in real-time big data is a very good thing because the anomaly can be understood even by a person who knows little about this area of study. Anomaly detection is not an easy task and it is even harder when we have to deal with real-time data.

In the future work, we plan to extend the usage of functions that Timelion offers. In the next phase of our research, we are going to implement other algorithms for anomaly detection in real-time big data. For example moving median, Atlas algorithm, etc. The aim is to find the appropriate algorithm which will work in real time environment and decrease the number of TN anomalies.

References

1. Aggarwal, C.C.: Outlier Analysis. Springer Science+Business Media, New York (2013)
2. Hasani, Z., Kon-Popovska, M., Velinov, G.: Survey of technologies for real-time big data streams analytic. In: 11th International Conference on Informatics and Information Technologies, Bitola, Macedonia, 11–13 April 2014
3. Hasani, Z., Kon-Popovska, M., Velinov, G.: Lambda architecture for real-time big data analytic. In: ICT Innovations 2014 Web Proceedings (2014). ISSN 1857-7288
4. Hasani, Z.: Performance comparison throws running job in Hadoop by defining the number of maps and reduces. In: 12th International Conference on Informatics and Information Technologies 2015, Bitola, Macedonia, 24–26 April 2015
5. Hasani, Z.: Virtuoso, system for saving semantic data. In: 12th International Conference on Informatics and Information Technologies 2015, Bitola, Macedonia, 24–26 April 2015
6. Hasani, Z.: Robust anomaly detection algorithms for real-time big data: comparison of algorithms. In: 6th Mediterranean Conference on Embedded Computing (MECO). IEEE (2017)
7. Bitnami. https://docs.bitnami.com/virtual-machine/apps/elk/. Accessed 02 July 2017
8. Kibana Timelion - Anomaly Detection, 18 January 2017. https://rmoff.net/2017/01/18/kibana-timelion-anomaly-detection/. Accessed 28 July 2017
9. Timelion. https://www.elastic.co/guide/en/kibana/current/timelion.html. Accessed 28 July 2017
10. Hasani, Z., Jakimovski, B., Kon-Popovska, M., Velinov, G.: Real-time analytics of SQL queries based on log analytic. In: ICT Innovations 2015 Web Proceedings (2015). http://proceedings.ictinnovations.org/attachment/conference/12/ict-innovations-2015-web-proceedings.pdf. ISSN 1857–7288
11. Tamura, K.: Elasticsearch, Fluentd, and Kibana: Open Source Log Search and Visualization. https://www.digitalocean.com/community/tutorials/elasticsearch-fluentd-and-kibana-open-source-log-search-and-visualization. Accessed 7 Jan 2016
12. Hasani, Z.: Implementation of infrastructure for streaming outlier detection in big data. In: Rocha, Á., Correia, A., Adeli, H., Reis, L., Costanzo, S. (eds.) Recent Advances in Information Systems and Technologies, WorldCIST 2017. Advances in Intelligent Systems and Computing, vol. 570. Springer, Cham (2017)

13. Kibana Timelion - Anomaly Detection, 18 January 2017. https://rmoff.net/2017/01/18/kibana-timelion-anomaly-detection/. Accessed 05 July 2017
14. Timelion. https://www.elastic.co/guide/en/kibana/current/timelion.html. Accessed 12 July 2017
15. Comparison between Fluentd and Logstash. https://logz.io/blog/fluentd-logstash/. Accessed 20 Sept 2017