Data intensive architectures

Project Report

Marco Baglio
x20199520@student.ncirl.ie

***Abstract*—This report proposes an analysis of the relationship between fire incidents and climate variables. The fire incidents in a town in the U.S. are joined with the values of temperature, wind speed and precipitation recorded in the same day by a set of climate stations. The two large datasets are joined by means of the Hadoop framework for parallel processing. The summary statistics on the joined datasets are computed by Hadoop and the visualization of results is generated separately. Finally, it is proposed the threshold values for a state of alert and suggestions for future works. The simple methodology proposed in this report with the opportune adjustments can be valuable for the monitoring of area at risks of fire or other incidents.**

I. Introduction

This report is written for the project assignment of Data Intensive architectures.
The objective of the analysis is the investigation of the influence of climate variables with occurrences of incidents.
Nowadays, the collection of data for all sort of applications is becoming more common and relevant information can be revealed by processing and analyzing these data. In a large scale, the collection of data from a large community offers the opportunity to investigate some key aspects of common phenomena such as incidents occurring in the community. By its own nature, the amount of data to treat is large and adequate programming architectures are necessary. When the key aspects are discovered, the decision makers can use them to limit the occurrences and mitigate the effects of those incidents.

The datasets involved in the analysis are:

- The incidents from fire department in the town of Cary, US. The dataset includes the following fields:
  - Alarmtime
  - Year
  - Incidentnum
  - Exp_no
  - Incidentcode
  - Incitypedesc
  - Indicentdesc
  - Majorcategory
  - Streetaddress
  - Mutl_aid
  - Station
  - Shif
  - Current_district
  - Current_fmz
  - Latitude
  - Longitude

The dataset is provided by U.S. Government's open data and contains 50377 records.

- The measurement of temperature, precipitation and wind from a set of climate stations:
  - Station
  - Name
  - Date
  - AWND, average wind speed in [MPH]
  - PRCP, precipitation in [inch]
  - TAVG, average temperature in [°F]
  - TMAX, maximum temperature in [°F]
  - TMIN, minimum temperature in [°F]

The dataset is provided by the National Centers for Environmental Information (NOAA) and contains 120212 records.
The two datasets are joined together with the scope of examine the climate values for each incident.

The research questions are as follows:

- What is the threshold value of Temperature to declare a state of alert?
- What is the temperature value that has the largest influence on the incident (maximum, minimum or average)?
- What is the influence of precipitation and wind on the occurrences of incidents?

II. Related Work

A method to collect and analyze data from wireless sensor is proposed in [1]. The sensors are installed in a forest and a parallel processing by means of Hadoop is adopted. In particular, visual cameras are placed on towers and take photos of the forest with a certain frequency. A motor is installed and allows the rotation of the camera 360°. Additionally, temperature and humidity sensors are installed at the location of the camera. The image data as well as temperature and humidity data are stored in an unstructured form. The MapReduce algorithms allows the reduction of the large datasets into simpler datasets. The machine learning tool Mahout is used to filter and classify the datasets and an image processing is used to monitor the entire forest. The procedure is the following:

- A set of nodes uniformly distribute over the area of the forest is created. All the nodes are synchronized to the same clock
- The nodes are clustered under a base station and the base stations to the control center
- When the values of humidity and temperature are under a threshold values, the frequency of measurement is 30 minutes. If the values are above the threshold the frequency is reduced.
- When a node locates the fire sends a danger alert to near nodes and starts the timer.

- All the nodes send the images to the base station that calculates the rate and direction of spread of fire by image processing.

The rate of spread of fire is estimated as

$$Rate\ of\ spread\ of\ fire = \frac{Distance\ between\ two\ nodes}{Time\ interval\ between\ reception\ and\ fire\ detection} \quad (1)$$

A framework of big data analytics for internet of small things is proposed in [2]. The framework is composed by the following layers:

- Small things layer: all the devices that measure temperature and humidity. This layer generates a stream of data that leads to a big volume of data to process
- Infrastructure layer: multiple gateways that receive data the underlying layer.
- Platform layer: the data are collected and pre-processed. The redundancy, the removal of noise and Min-Max normalization are some of the tasks undertaken by this layer.
- Application layer: the data are visualized by the user in form of graphs, plots, table, etc.

The framework is tested for the monitoring of humidity in a house. A single-node Hadoop is implemented. The humidity is monitored for the impact on the life of the house dwellers and to control the usage of electricity in relation to the humidity. The system analyzes the previous data of humidity and offers a prediction of the future. An interesting evaluation of the data Throughput versus the data size employed in the system reveals that the application of Hadoop is efficient when the data size are large enough (500 MB in this example).

The application of Hadoop big data architecture on remote sensing with satellites data is analyzed [3]. The sensors in satellites can provide level of CO, SO_2 and NO_x. A huge amount of data is produced that can be applied into the prediction of environmental issues. The framework proposed is very similar to the one presented in [2]:

- Data source layer
- Ingestion layer; the connection of all the data sources and pre-processing is performed on this layer
- Storage layer; the pre-processed data are to a Hadoop Distributed File System (HDFS)
- Processing and query layer; the batch processing is performed by MapReduce.
- Visualization layer; the results are plotted into charts, graphs and tables
- Monitoring layer; the performance of the whole framework are monitored on this layer.

The framework is applied to previous case studies (such as forest fire detection, air pollution monitoring, etc) in order to have a comparison. The results obtained are comparable with the difference that the system processed data from more satellites allowing a solution that is scalable and optimized.

A temperature monitoring system based on Hadoop and visible light communication is tested [4]. Six computers are used to build a Hadoop cluster to collect and process data from the sensors. A 10-storey school building with 20 rooms on each and five temperatures sensors in each room collecting data every minute. The test involved the verification of the system of transmission depending on the number of sensors. When the number of nodes is 200 the maximum error obtained is 3%. This system provides a long range monitoring of massive temperature data scalable according to the application required.

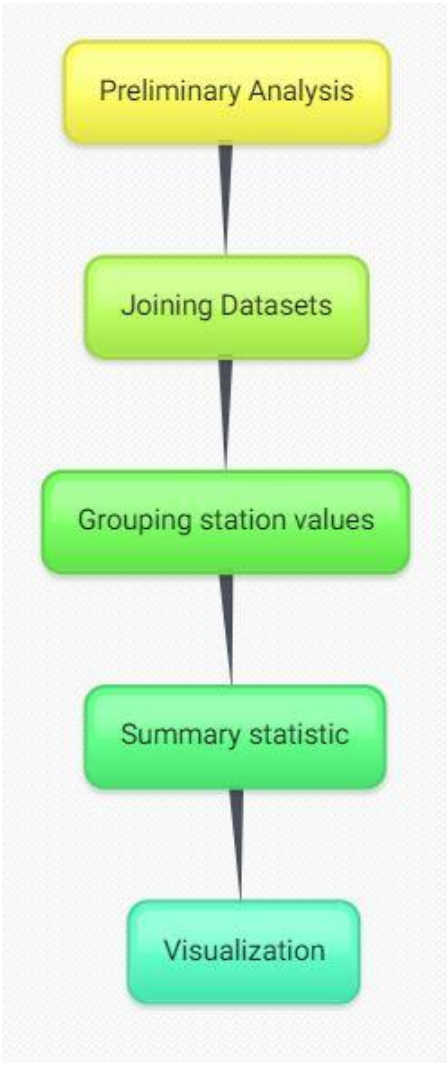

Figure 1 Overview of the workflow

III. METHODOLOGY

An overview of the workflow is represented in Figure 1. The coding language used is Python and the Hadoop framework is run on Debian Linux installed on a virtual machine. The scripts are based on the MrJob library.

The stages of the analysis are as follows:

1. Preliminary analysis
2. Joining datasets
3. Grouping station values
4. Summary statistic
5. Visualization

The next section will describe each part of the workflow.

A. Preliminary analysis

At this stage the datasets are inspected.

The dataset of climate station contains values for a range of time between the 01/01/2016 and 31/12/2019 with a daily frequency. The fields contain values of wind speed, precipitation, average temperature, maximum and minimum temperature. The values of 127 climate stations are concatenated vertically and sorted by date. Some stations do not provide values of one or more fields. Not all the range of dates are covered by each station. The imperial unit system is used. The original .csv file is separated by comma.

The dataset of fire incidents contains values for a range of time between the 01/01/2016 and 01/02/2021. A lot of fields are not required for our analysis so removed to reduce the amount of data transmitted in the network*: year, exp_no, incidentcode, incidentdesc, majorcategory, streetaddress, mutl_aid, station, shift, current_district, current_fmz, latitude, longitude* and *geopoint*. The *alarmtime* contains the date in a different format containing also hours, minutes and seconds. The original .csv file is separated by semi-colon, for this reason the separator for the other dataset is changed into semi-colon to have same separator.

The first script (0Test.py) aims to investigate the functionality of Hadoop and the correctness of the coding. The functionality of Hadoop is visualized in Figure 2:

1. The data are split into splits or blocks that match the number of map workers
2. Fork the program to masters and worker. The workers run the map and reduce functions. The master assign the task to the idle workers.
3. The map workers read the block of data assigned. Each worker can handle more than one block. The map function requires an input in form of (key,value) pair and yields an intermediate (key, value) pair:

$$Block: (key_1, value_1) \xrightarrow{Map\ Function} (key_2, value_2) \quad (2)$$

   In this case, the chosen key is the date. The values are two fields randomly selected for testing purposes. The date format at the moment is different between the two datasets.
4. The values obtained are sorted by key. The intermediate keys are grouped together and passed to only one Reduce function:

$$(key_2, value_2) \xrightarrow{Reduce\ Function} (key_3, value_3) \quad (3)$$

   Typically the $value_3$ is the final output and the final key_3 is omitted because not necessary. In this case, the reducer simply return the (key,value) pair obtained from the mapper.

B. Joining Datasets

This stage regards the join of the two datasets. An inner join is performed. The key is date in the format DD/MM/YYYY. The key is already found for the fire incidents. The key is in the field *alarmtime* for the climate stations in the format (YYYY-MM-DD…) therefore the date is converted into the chosen format. The mapper recognizes the input file by the number of fields and assign a string into the value in order to allow the reducer to recognize the origin of the data. The joined dataset contains the following fields:

- Station name
- Average wind speed
- Precipitation
- Average temperature
- Maximum temperature
- Minimum temperature
- Incident Type
- Incident ID

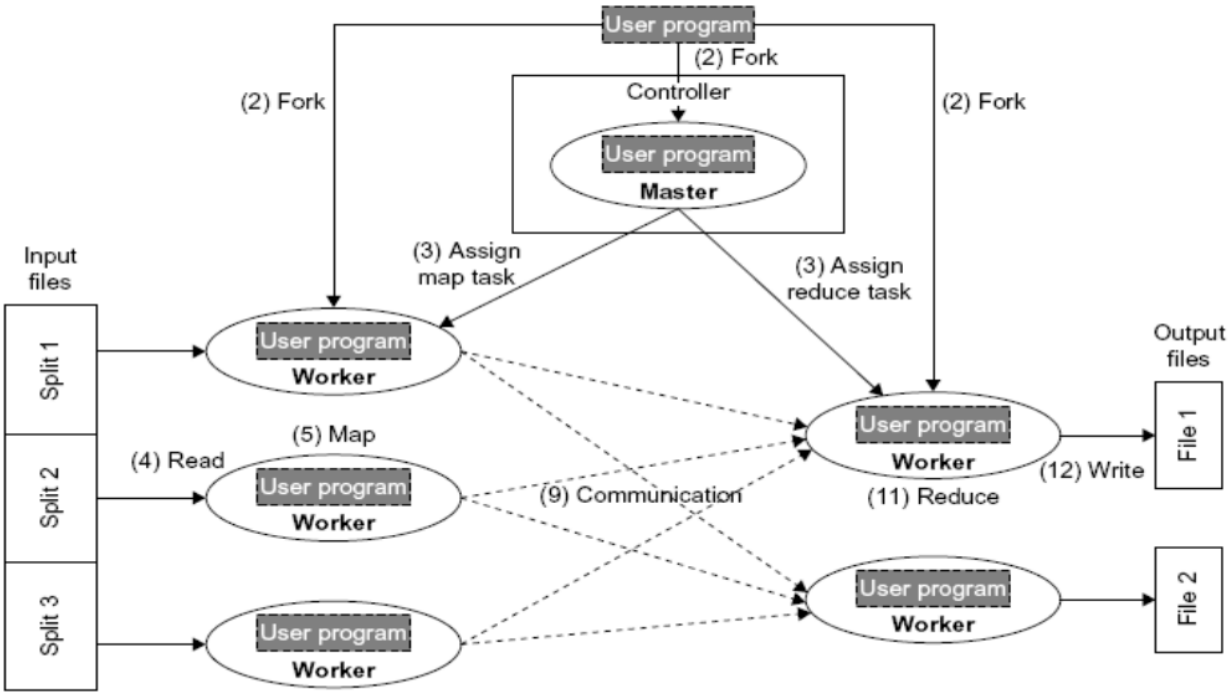

Figure 2 Framework of MapReduce implementation (Courtesy of Yahoo! Pig Tutorial [54])

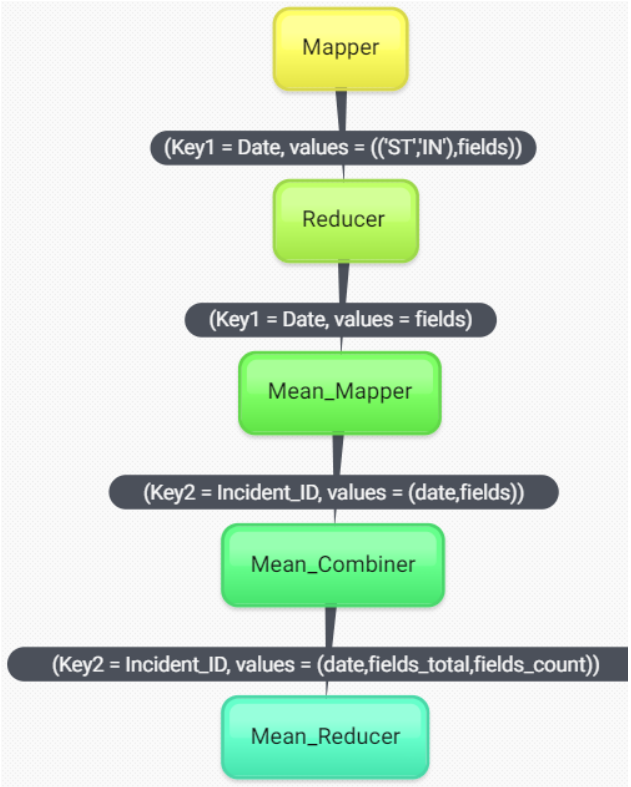

Figure 3 Workflow for joining and grouping the datasets.

The final output contains 127 for each incident corresponding to values of each climate station. The scope of the next step is to average the values of all the stations into one value.

C. *Grouping station values*

The workflow of this stage is plotted in Figure 3. At this stage, the script requires the introduction of steps because more than the Map – Reduce functions is required. The first two steps are identical to the previous stage with the only difference that the name of the climate station is removed because irrelevant when the all the stations are averaged. The third stage shows a new map function:

$$(Date, fields) \xrightarrow{Mean\_Mapper} (incident\_ID, (date, fields)) \quad (4)$$

The Mean_mapper function takes the (key,value) pair from the reducer and provides a new key, the Incident_ID. The values now contains the date and the fields from the reducer. The values (key,values) pair are now processed by the combiner function. This function is defined as the "Mini-reducer" processing the output data from the mapper before passing to the reducer. It decreases the amount of data that the reducer has to process reducing the network congestion. In this case, the combiner calculates the sum and the count of the fields to be averaged. Each variable has its own count because not all the fields are always present.
Finally, the reducer completes the calculation of the sum and count for each field. The average is calculated and the output is saved as file.

D. *Summary stastics*

The output values are now processed to obtain the summary statistics for each fields. Minimum, maximum and a set of quantiles values is calculated. Two separate scripts are used:

1. The mapper function reads the file, converts the grade Fahrenheit to Celsius, emits the incident_ID as key and the fields to be summarized. The reducer calculate the summary static per incident_ID.
2. The mapper function reads the file, converts the grade Fahrenheit to Celsius, emits a singular key and the fields to be summarized. The reducer calculates the summary static for all incidents.

The summary statistics are saved into file and processed and visualized in the next step.

E. *Visualization*

The summary statistics are visualized on Jupyter Notebook on Python with the library plotly. The next section will examine the plots and values obtained.

IV. RESULTS

In this section, the results obtained are discussed in detail. The final output contains several types of incident. The distribution of incident types over the total is provided in Table I. The total number of incident types is 151. Half of the incidents falls under the category of *EMS call* and *Medical assistance*. The other categories that are less common but relevant on the distribution of incidents are *Dispatched & cancelled en route* and *smoke detector activation.*

Incident Type	**Count [%]**
EMS call	35.2
Medical assistance	15.8
Dispatched & cancelled en route	5.9
Smoke detector activation	4.2
Alarm system activation	3.2
Smoke detector activation due to malfunction	2.9
Motor vehicle accident with injuries	2.9
MVA/No Patient Care provided by Cary FD personnel	2.0
Alarm system sounded due to malfunction	1.9
No Incident found on arrival at dispatch address	1.7
Motor Vehicle Accident with no injuries	1.6
…	…
Building fire - Low Risk	0.003
Aircraft standby	0.003
Munitions or bomb explosion (no fire)	0.003
Watercraft rescue	0.003
Overpressure rupture of steam boiler	0.003
Biological hazard investigation	0.003

Table I Count of incident types as percentage of the total number of incidents

Additionally, the dataset contains other categories less frequent that expand largely the total number of incident types (e.g. *Biological hazard investigation* or *Building fire-low risk)*. The original dataset should contain all incidents recorded in the fire department; however, some incident types are not strictly "fire incidents".

The *EMS call*, *Medical assistance*, *Dispatched & cancelled en route* and *smoke detector activation* categories are individually analyzed. The cumulative distribution of the average temperature (Figure 6) shows similar values for all the incident types. The range of temperatures is between 0 and 30 degrees Celsius. The 75 percent of incidents occurs for values of temperature larger than 10 degrees. Similarly, all the incident types have a similar cumulative distribution of the wind speed (Figure 6). The range of wind speed is between 0 and 25 miles per hour. The 75 percent of incidents occurs for values of wind speed above 7.5 miles per hour. The cumulative distribution of the precipitation (Figure 6) shows similar values for all the incident types. The range of precipitation is between 0 and 40 inches. The 75 percent of incidents occur for precipitation larger than 17 inches.

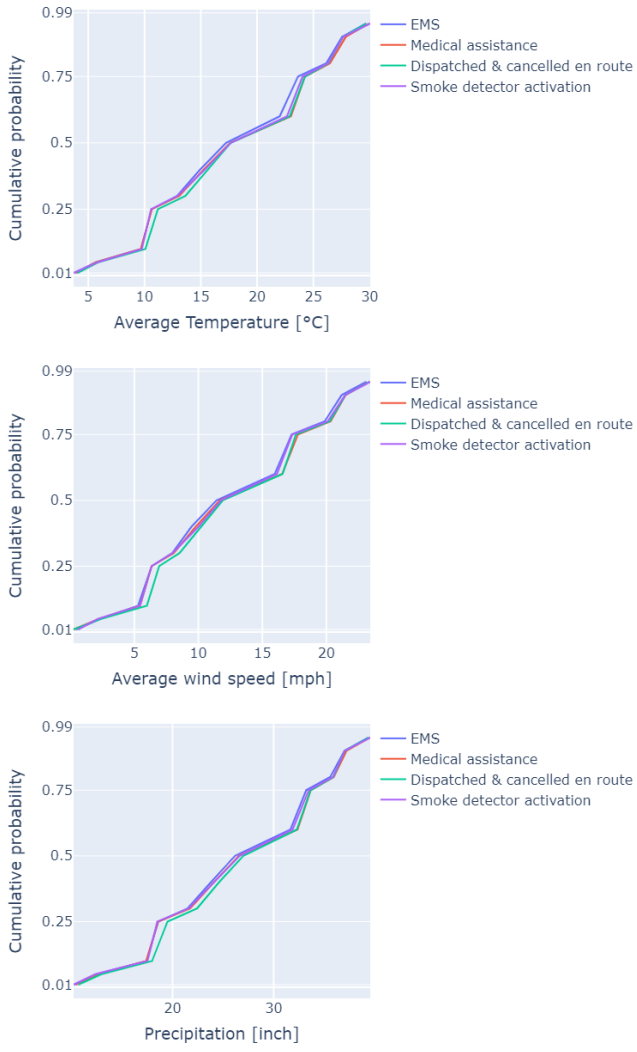

Figure 6 Cumulative probability of average temperature, average wind speed and precipitation. The most frequent incident categories are shown for comparison

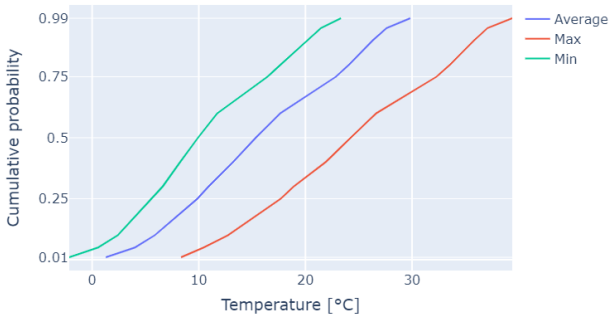

Figure 4 Cumulative probability of average, maximum and minimum temperature for all the incidents.

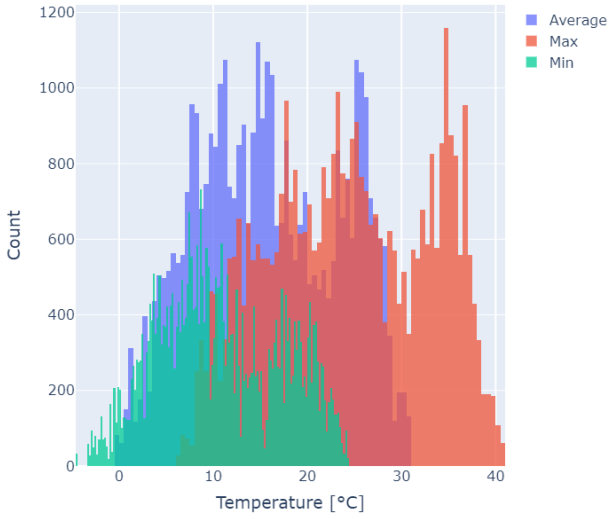

Figure 5 Distribution of average, maximum and minimum temperature for all the incidents

Finally, the analysis is carried out on the totality of incidents. The cumulative distribution of the average, maximum and minimum temperature (Figure 4) shows a trend almost coincident. The 75 percent of incidents occurs for average temperature above 10 degrees, maximum temperature above 18 degrees and minimum temperature above 6 degrees.
The distribution of temperatures shows that there are two modes where the incidents are most frequent (Figure 5), the correspondent values are:

- Average temperature: 13 degrees and 25 degrees
- Maximum temperature: 25 degrees and 35 degrees
- Minimum temperature: 9 degrees and 20 degrees

Overall, the distributions have a very similar shape but they are offset by almost a fixed value.

V. Conclusions and Future Work

This paper presented an analysis of fire incident in a town in the U.S. depending on the values of climate stations. The Hadoop framework for parallel processing was adopted due to the large size of the datasets.
The threshold values to increase the level alertness are identified at average temperature of 10 degrees, minimum temperature of 6 degrees and maximum temperature 18 degrees.

Furthermore, the analysis revealed that maximum, minimum and average temperatures have the same influence on the occurrences of incidents.
Precipitation and average wind speed appears to be slightly influent for the incidents occurrence. The threshold values can be considered 7 mph for the wind and 19 inch for the precipitation.

The vast distribution of incident types is a limitation of the analysis, because a part of the incidents might not be related to climate values. The values from the climate station can be related to the location of the incident increasing the overall accuracy in future works. The combined cumulative distribution of the climate values can be object of further analysis in order to consider, for example, combination of wind speed and temperature. The humidity and pollution should be influent on some categories of incidents, therefore future analysis should consider them.

The paper proposed a methodology that can be applied into more specific applications, where the incidents are more strictly related to the climate values. For example, the same analysis can be conducted on monitoring forests accompanying the methodologies proposed in the literature review.

VI. BIBLIOGRAPHY

[1] Rajasekaran, T., Sruthi, J., Revathi, S., & Raveena, N. (2015). Forest fire prediction and alert system using big data technology. In Proceedings of the International Conference on Information Engineering, Management and Security, ICIEMS (pp. 23-26).

[2] Gohar, M., Ahmed, S. H., Khan, M., Guizani, N., Ahmed, A., & Rahman, A. U. (2018). A big data analytics architecture for the internet of small things. IEEE Communications Magazine, 56(2), 128-133.

[3] Semlali, B. E. B., El Amrani, C., & Ortiz, G. (2020). Hadoop paradigm for satellite environmental big data processing. International Journal of Agricultural and Environmental Information Systems (IJAEIS), 11(1), 23-47.

[4] Zhou, T., Lee, X., & Chen, L. (2018). Temperature monitoring system based on hadoop and VLC. Procedia computer science, 131, 1346-1354.