ICIC Express Letters
Part B: Applications
Volume **12**, Number **12**, December **2021**

 ISSN 2185-2766
pp. 1101–1109

AIR QUALITY CLASSIFICATION USING NAÏVE BAYES CLASSIFIER ON DISTRIBUTED RASPBERRY PI CLUSTER SYSTEM

Romi Fadillah Rahmat[1,*], Sarah Purnamawati[1]
Artambo Benjamin Pangaribuan[2], Reza Taqyuddin[1]
and Tifani Zata Lini[3]

[1]Department of Information Technology
Universitas Sumatera Utara
Jl. Alumni No. 3, Medan 20155, North Sumatra, Indonesia
{ sarah_purnamawati; rezataqyuddin }@usu.ac.id
*Corresponding author: romi.fadillah@usu.ac.id

[2]Department of Informatics
Universitas Pembangunan Nasional Veteran
DKI Jakarta 12450, Indonesia
artambo@upnvj.ac.id

[3]Department of Business Information Technology
University of Twente
Enschede 7522NB, The Netherlands
tifanizatalini@student.utwente.nl

Received March 2021; accepted June 2021

Abstract. *The air condition is influenced by the amount of pollution occurring in a specific area which consists of particulate matter, ozone, nitrogen oxides, and carbon dioxide. These pollutants are determined using a standard-categorization value abbreviated as AQI (Air Quality Index). The pollutant value varies in a specified time span, causing difficulty in classifying the air quality into certain categories of AQI. A distributed system, such as cluster machine, has a good performance to manage big data in distributed ways. The Raspberry Pi built as a cluster can increase the performance of processing and resources that are compulsory to manage varieties of big data. This study was completed by implementing the Naïve Bayes method on Raspberry Pi cluster server in which collected data using web scraping method will be distributed to each slave node cluster, and then the master node will send signal to process the data. Based on the results, it can be concluded that implementing Naïve Bayes on cluster server can build a model of collecting data and perform classification on distributed system with an accuracy of 98%.*
Keywords: Web scraping, Naïve Bayes, Raspberry Pi cluster, Air quality classification

1. **Introduction.** Air is an essential component for mankind in the process of respiration. Physical development in cities and industrials area is impacting air component. Changes on air component affect the air quality which results in air pollution. The decreasing air quality can damage the health of the surrounding community [1].

Air pollution is one of the major problems in large cities of developing countries. Align with the ascending of living standards; people nowadays tend to pay attention toward health and environment. Air-quality observation can be measured by the Air Pollutant Standard Index (PSI). There are five parameters of air pollution measured in the observation by PSI namely Carbon Monoxide (CO), Tropospheric Ozone (O_3), Particulates Matter (PM10 and PM25), Nitrogen Oxides (NO) and Sulfur Dioxide (SO_2).

Web scraping is the process of making a semi-structured document from the Internet, usually in the form of web pages in a markup language like HTML or XHTML, and

DOI: 10.24507/icicelb.12.12.1101

then analyze these documents to retrieve specific data from the page to be used for other purposes [2]. In order to retrieve information from large amount of raw sensor data from the Internet, web scraping would serve as the method to obtain these data (Extracting and Saving).

A distributed system is a collection of independent computers connected in a network to which the user is a coherent system. The main purpose of a distributed system is to be able to easily connect users with resources that are distributed through a network [3].

Intelligent algorithms in data mining or machine learning have been widely applied to analyzing the data of air pollution such as using Naïve Bayes, wavelet transform, cluster algorithm and SOM (Self-Organization Map) for analysis of air pollution in Taiwan [4]. In 2012, Gilbert et al. proposed a system with the incorporation of intelligent data mining and decision support for analyzing air quality data [5].

Naïve Bayes is one of the algorithms for classification using the Bayes' Theorem which assumes that any object to make predictions is not bound or free from each other. Briefly, Naïve Bayes classifier assumes that the presence of certain features in a class does not relate to other features [6].

Several studies on air quality have been conducted, such as a research on air quality prediction using machine learning approaches by Kaur et al. which aimed to investigate various big-data and machine learning based techniques for air quality forecasting [7].

Another research was conducted by Castelli et al. to employ a popular machine learning method, Support Vector Regression (SVR), to forecast pollutant and particulate levels and to predict the Air Quality Index (AQI). The result generated an accuracy of 94.1% on unseen validation data [8]. In this research, authors conducted a study in implementing the Naïve Bayes method on Raspberry Pi cluster server to build a model of collecting data and classification on distributed system in which the research data were acquired every hour to ensure the effectiveness of the result.

The paper is organized as follows. Section 2 reviews several previous researches related to air quality, web scraping and Naïve Bayes classifier as well as the proposed method to collect the air condition data using web scraping and classification using Naïve Bayes classifier, which consists of several steps. Result and discussion are presented in Section 3. We will discuss the result of the proposed method in this research. Section 4 describes the summary and suggestions for future research.

2. Materials and Methods.

2.1. **Related work.** There were numerous researchers who conducted studies on air quality monitoring and classification with various tools and techniques, which mainly aim to build a system which will enable to collect and monitor the air condition data and to classify it based on air quality index. Recently, it has also been implemented for the other purposes such as distributed weather and air station, air quality monitoring, and visualization of real time air data. In this section, we will discuss a few previous researches related to web scraping and Naïve Bayes classification approaches.

Air pollution is a condition that causes alteration in the composition of air than normal thus to endanger life and public health. According to the government decreet no 41, 1999, air pollution is a process of adding some substances, energy, and other components into the ambient air caused by human activities. The main source of pollution comes from transportation, with nearly 60% of pollutants produced consisting of carbon monoxide and about 15% is composed of hydrocarbons [8].

In urban and industrial areas pollutant parameters that need to be considered in connection with respiratory diseases are the SO_2 (Sulplhur Dioxide), CO (Carbon Monoxide), NO_2 (Nitrogen Dioxide) and Particulate Matters (PM10 & PM25) [9].

Air Quality Index (AQI) is a value used by government agencies to illustrate the public about air condition or how to predict the pollution that would happen. AQI calculation requires the average concentration of pollutants on a certain period which is obtained from the results of air monitoring [10]. The level of pollution by AQI is divided into six categories which can be seen in Table 1.

Table 1. AQI level

AQI value	**Health level**	**Color**
0-50	Good	Green
51-100	Moderate	Yellow
101-150	Unhealthy for sensitive groups	Orange
151-200	Unhealthy	Red
201-300	Very unhealthy	Purple
301-500	Hazardous	Maroon

The formula used to calculate the value of air quality index is as follows:

$$I = \frac{I_{high} - I_{low}}{C_{high} - C_{low}}(C - C_{low}) + I_{low} \tag{1}$$

Web scraping or web harvesting is a technique of computer program to perform the extraction of information from web pages. Web scraping application also called intelligent, automated, or autonomous agents only focuses on how to obtain the data through the retrieval and extraction of data structured with various data sizes [11].

A distributed system is a collection of independent computers connected in a network to which the user is a coherent system. The main purpose of a distributed system is to easily connect users with resources that are distributed through a network [3]. Based on research by Coulouris [12], a distributed system has 6 characteristics which are resources sharing, open sources, concurrency (a distributed system makes the operating system possible to execute some processes on each node), scability (a distributed system can be upgraded by adding new resources to fulfill the system requirement), fault tolerance (distributed system can be used as load balancing. If a node has trouble it will not affect the whole system), and transparency.

Naïve Bayes is one of the algorithms for classification that uses the Bayes' Theorem which assumes that any object to make predictions is not bound or free from each other. Briefly, Naïve Bayes classifier assumes that the presence of certain features in a class does not relate to other features. By implementing Bayes rules, formula can be described as follows.

$$P(Y = y_i|X = x_k) = \frac{P(X = x_k|Y = y_i)P(Y = y_i)}{\sum_j^k P(X = x_k|Y = y_j)P(Y = y_j)} \tag{2}$$

2.2. **General architecture.** The proposed method, to classify the air quality on distributed system Raspberry Pi cluster server in this study, consists of several steps, namely data acquisition using web scraping, distributing data in cluster machine and classifying the data. Figure 1 shows the general architecture of our proposed method.

2.3. **Data acquisition.** The data used in this study are obtained from aqicn.org website. It is an air quality monitoring website that provides description of the air quality by displaying the values of pollutant concentrations from several regions. In this phase, the data will be extracted to generate the values of pollutant parameters from the website such as Carbon Monoxide (CO), Ozone Surface (O_3), Particulate Matter 10 nm (PM10), Sulfur Dioxide (SO_2), Temperature, Dew Point, Precipitation, Humidity, and Wind Speed.

There were several steps taken in building the web scraping API built consisting of several steps. First was to check the input URL. This step was to verify that URL given

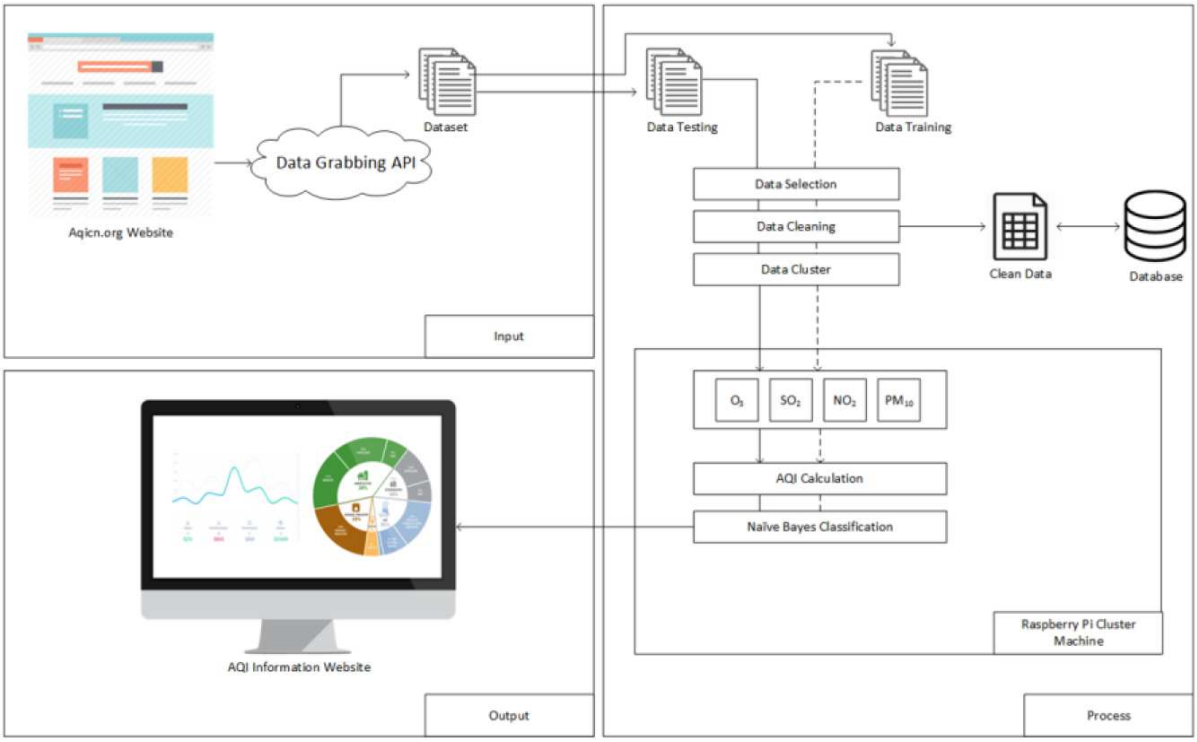


FIGURE 1. General architecture

was a valid URL. The given URL must be originated from aqicn.org. Once the URL checked, the next step would be obtaining the html semantic structure from the URL by rendering it into the html code. Then, the result of DOM object of the generated structure would be parsed into string. The process continued to extracting the wrapper class element from the code structure. The wrapper class was aqiwgt-table-aqiinfo. The class obtained the div elements as shown in Figure 2.

Afterwards, the DOM element inside wrapper class would be parsed to get the specific value from an element. The element that contained the data is marked with class: ***PM25***, ***PM10***, $\boldsymbol{O_3}$, $\boldsymbol{NO_2}$, $\boldsymbol{SO_2}$, ***CO***, $\boldsymbol{t}$, $\boldsymbol{d}$, $\boldsymbol{p}$, $\boldsymbol{h}$ and $\boldsymbol{w}$. It is shown in Figure 3. Each field has 3 values, namely cur, min and max.

The last step would be checking if the data to be restored were worth saving or not the duplication of the previous one. If the data were worth saving, then the query, to save it into the database, will be executed.

2.4. **Data modification.** The acquired data from previous phase will be modified. Modifying the data was achieved by the following steps:

Step 1: Select the required data. This step would select the data from the table by getting the current value of AQI for each pollutant. Data to be used were cur_PM25, cur_PM10, cur_CO, cur_NO and cur_O3 and status.

Step 2: Convert the AQI values. The AQI needs to be converted to its original pollutant concentration. It can be done by applying reverse formula from Formula (1). For example, we convert the PM25 AQI values 105 to its concentration:

$$cc = \left(\frac{I - I_{low}}{I_{high} - I_{low}} \times (C_{high} - C_{low})\right) + C_{low} \tag{3}$$

$$cc = \left(\frac{4}{49} \times (19.9)\right) + 35.5 \tag{4}$$

From this formula, we can calculate PM25 concentration from its 105 AQI which is 37.12.

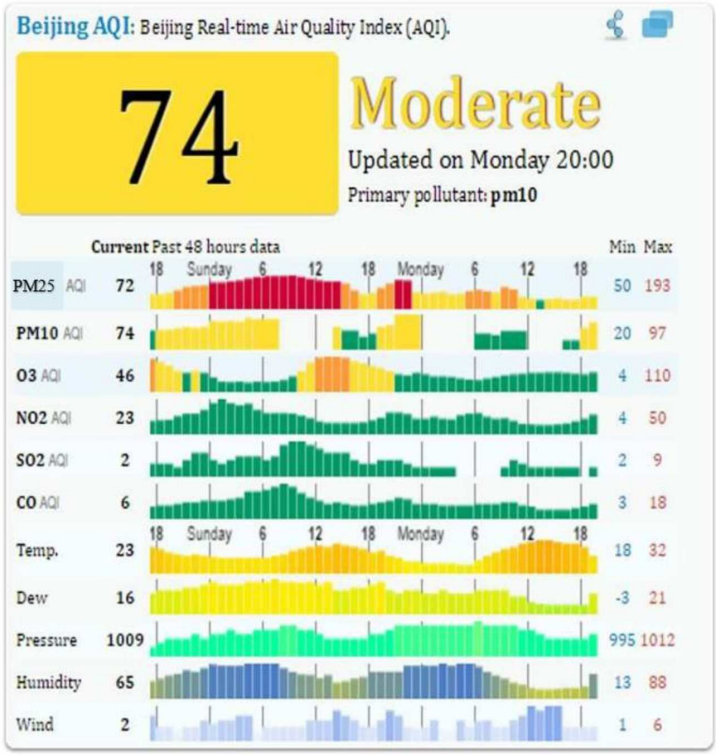


FIGURE 2. Special wrapper class

FIGURE 3. Selected field contains cur, min, and max of PM25

Step 3: Clean the data from anomaly or invalid value. The cleaning process is to ensure consistency of the data. For example, if the data has no value or NULL, we can clean by removing data or changing the value to 0.

Step 4: This step is to normalize the data. Multiple distinctive data value compounds its calculation or even its classification. The data can be normalized using some data transformation techniques such as min-max normalization or Z-score.

2.5. **Data distribution.** After the data modification phase was accomplished, the next phase would be to distribute the data to each connected cluster node. In this study, the cluster server was built using 5 Raspberry Pi 2. Each node had been installed Raspbian Jessie operating system and connected in a network. Each node must have installed nfs-kernel-server, rpcbind portmapper, passwordless ssh, mpich2, c++ and python modules, python 2.7.12 and python 3.5.2.

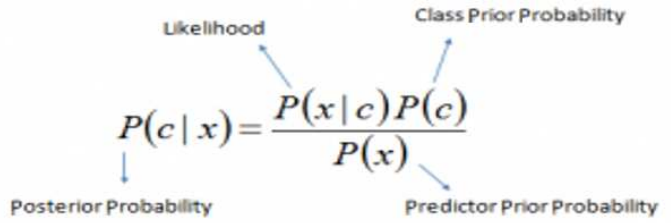
2.6. **Data classification.** This phase was the final phase of this system. The classification was done by implementing Naïve Bayes classifier method. Figure 4 shows the PM25 parameter has significant value than another parameter.

no	cur_cc_pm25	cur_cc_pr	cur_cc	cur_cc_	cur_cc_c	cur_t	cur_	cur_p
1	228	125	15	55	14	3	-9	1020
2	246	129	17	57	12	2	-8	1020
3	252	129	18	67	6	0	-6	1020
4	272	129	19	64	5	1	-7	1019
5	270	115	29	66	4	-1	-8	1019
6	263	120	37	71	5	-2	-8	1019
7	278	134	38	71	5	-3	-7	1018
8	274	134	38	71	5	-2	-7	1017
9	289	138	49	71	6	-3	-7	1017
10	281	138	49	71	6	-4	-7	1017
11	269	126	34	56	4	-5	-7	1017
12	274	126	34	56	4	-4	-7	1017

13	248	123	42	63	5	3	-5	1018
14	256	135	43	75	5	7	-6	1018
15	244	135	43	45	5	8	-7	1018
16	240	135	43	45	5	14	-6	1017
17	218	135	12	56	25	16	-7	1016
18	184	135	12	56	25	17	-8	1016
19	161	135	16	38	25	20	-8	1014
20	174	135	8	33	35	19	-8	1014
21	180	135	8	27	37	19	-8	1013
22	172	75	6	27	36	13	-6	1014
23	178	90	8	58	17	13	-6	1014
24	188	101	10	66	9	12	-5	1014
25	208	95	12	54	13	11	-5	1014

FIGURE 4. Selected 24 hours of AQI value of PM25

Based on Equation (5), then it was compulsory to convert the data to its frequency table and the probability as shown in Figure 4.



$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \cdots \times P(x_n|c) \times P(c) \tag{5}$$

From Table 2, it can be seen that there were top two classes which have higher value than other classes. These were obtained using Bayesian formula (2) where each class probability was calculated. The class with the highest probability was most likely to be predicted as classification result.

TABLE 2. Frequency and the probability table

Level	Frequency	Probability	Probability ratio
Good	0	0/24	0
Moderate	0	0/24	0
Unhealthy for sensitive groups	0	0/24	0
Unhealthy	7	7/24	0.29
Very unhealthy	17	17/24	0.70
Hazardous	0	0/24	0
Total data	**24**	–	–

Then, the likehood probability for 4 new objects could be calculated. With ratio 1 : 3, a compilation of objects is one object with probability unhealthy and three objects with probability very unhealthy.

3. **Results and Discussion.** In this section, the result of two main phases, data grabbing and classification, will be discussed and analyzed separately.

3.1. **Data grabbing.** The proposed method for data grabbing is using web scraping method. The script was written using php code. In order to execute the script, manually access where the script was stored, in this case, on http://localhost/airquality/grab/grab.php.

In Windows operating system, scheduling this task can be done using windows task scheduler. We set the scheduler so it will execute for every 30 minutes in each hour. Windows server is used as a backup system if Raspberry Pi server failed.

In Raspberry Pi server, access to the script can be done by calling command curl. In Linux, scheduling task can be set using cronjob. This system has been set to execute the script in the minute of 59 every hour. Figure 5 shows the air quality has AQI value 38 and status good before executing curl command. After executing curl command, the data will eventually change if the server has already updated the data as shown in Figure 6.

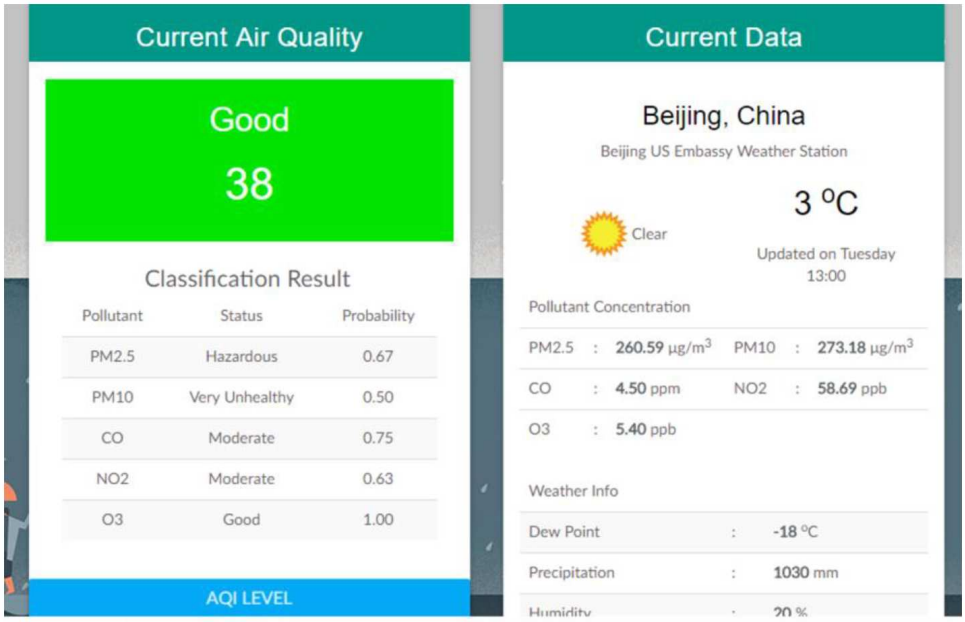


FIGURE 5. Before data grabbing (1.54 pm)

3.2. **Data classification.** The proposed method for classification is Naïve Bayes. Before classifying the data, it is essential to split the data to each node. Data from database will be converted into json data file. Json is structured into data for each pollutant, calculation of probability in each class and 24 hours data summary. Naïve Bayes is later executed as background process which will update the json file and append its classification result. The data from json then will be parsed so it can be displayed in user interface. The parsed json can be seen in Figure 7.

Data from previous hours were stored in json file to ease monitoring. Previous hours AQI data were displayed in web as graph for user convenience in using the information as shown in Figure 8.

Based on the conducted test, the accuracy of this research is at 98% where 5764 data rows were process, and the data grabbing was executed for 240 days and 16 hours.

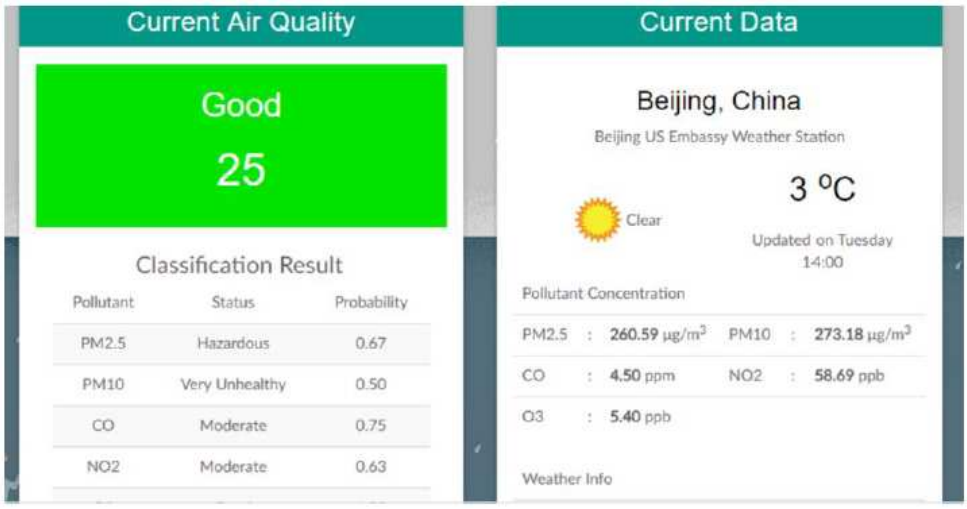


FIGURE 6. After data grabbing

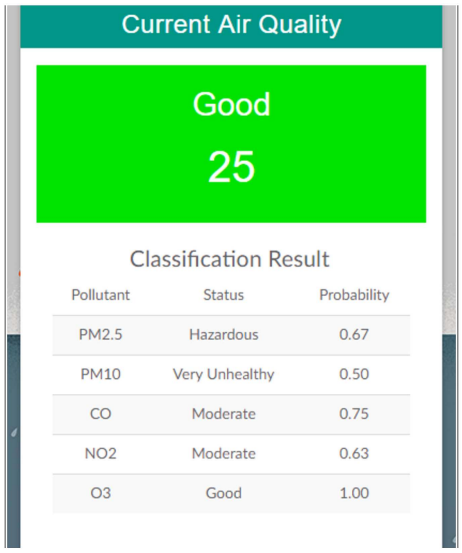


FIGURE 7. Classification result

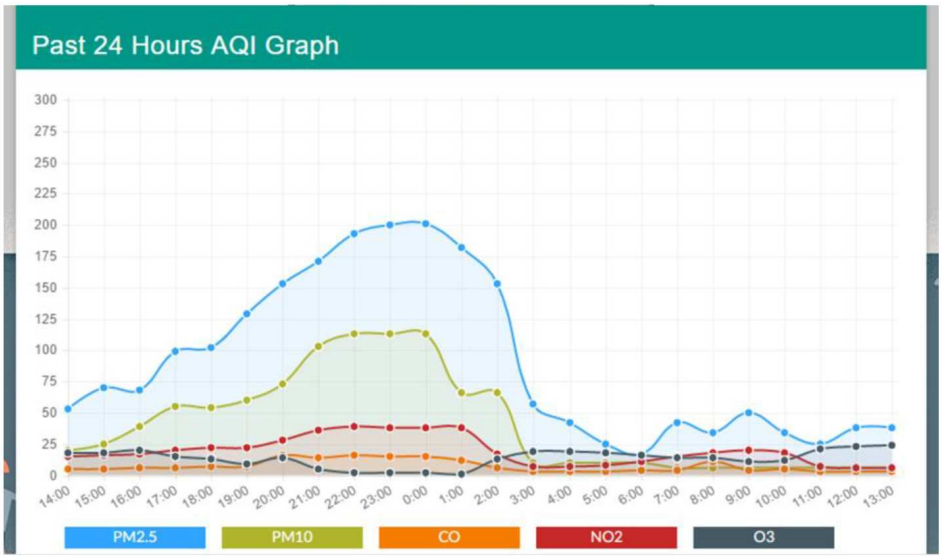


FIGURE 8. Graph on AQI level for past 24 hours

4. **Conclusion.** Based on all testing results, there were several conclusions obtained in this research. First, by implementing the web scraping method, it is possible to build a platform to gather and monitor air quality data. By implementing the Naïve Bayes method, the data can be classified to specific class in daily basis with an accuracy of 98%. By implementing both methods in a cluster machine, resource required to support this system can be pushed to minimum. It is highly suggested to add the wireless-sensors network as reliable data source for future work. In addition, having the data checked periodically while monitoring is essential in avoiding any data loss. Adding some scheduling method would enable the data execution concurrently.

REFERENCES

[1] P. Santi, *Air Quality Analysis of Global Atmosphere Watch (GAW) Station in Kototabang Hill, Agam Regency, West Sumatra*, Bachelor Thesis, Universitas Gajah Mada, Yogyakarta, 2012.

[2] M. Turland, *php|architect's Guide to Web Scraping with PHP*, PHP|Architect Press, United States, 2010.

[3] A. Tanenbaum, *Distributed Systems Principles and Paradigms*, Pearson Education, Singapore, 2003.

[4] S.-T. Li and L.-Y. Shue, Data mining to aid policy making in air pollution management, *Expert Systems with Application*, vol.27, no.3, pp.331-340, 2004.

[5] K. Gilbert, M. Sanchez-Marre and B. Sevilla, Tools for environmental data mining and intelligent decision support, *International Congress on Environmental Modelling and Software*, Canada, 2012.

[6] M. Koduvely, *Learning Bayessian Model with R*, Packt Publisher, United Kingdom, 2015.

[7] G. Kaur, J. Gao, S. Chiao, S. Lu and G. Xie, Air quality prediction: Big data and machine learning approaches, *International Journal of Environmental Science and Development*, vol.9, no.1, pp.8-16, 2017.

[8] M. Castelli, F. M. Clemente, A. Popovič, S. Silva and L. Vanneschi, A machine learning approach to predict air quality in California, *Complexity*, Hindawi, DOI: 10.1155/2020/8049504, 2020.

[9] S. Fardiaz, *Polutions of Water and Air*, Kanisius Publisher, Yogyakarta, 2003.

[10] G. A. C. R. Holzworth, *Air Pollution*, 3rd Edition, 1976.

[11] Z. Liao, Y. Peng and Z. Y. Liang, A web based visual analytics system for air quality monitoring data, *The 22nd International Conference on Geoinformatics*, pp.1-6, 2012.

[12] G. Coulouris, J. Dollimore, T. Kindberg and G. Blair, *Distributed Systems: Concepts and Design*, 5th Edition, Pearson, 2011.

[13] A. Josi and L. A. Abdillah, Web scraping technique on scientific article search engines, *arXiv Preprint*, arXiv:1410.5777, 2014.

[14] R. F. Rahmat, M. F. Syahputra and M. S. Lydia, Real time monitoring system for water pollution in Lake Toba, *International Conference on Informatics and Computing (ICIC)*, pp.383-388, 2016.