A Multi-Agent Framework for a Hadoop Based Air Quality Decision Support System

Abdelaziz El Fazziki [1*], Abderrahmane Sadiq [1], Jamal Ouarzazi [2], Mohamed Sadgal [1]

[1] Computer Systems Engineering Laboratory, Cadi Ayyad University of Marrakech, Morocco
Abderrahmane.sadiq@edu.uca.ma, {elfazziki, sadgal}@uca.ma
[2] Laboratoire Physico-Chimie des Matériaux et Environnement (URAC 20), Cadi Ayyad University of Marrakech, Morocco
ouarzazi@uca.ma

Abstract. Tropospheric pollution is controlled by various factors such as the distribution of pollutant sources, the nature and amount of energy, as well as the land use and meteorological parameters. These factors must be taken into account in the management of the air quality. Thus, a development of an air quality decision support system able to manage these factors and to answer the questions of environmental managers in real-time is imperative. Such system requires an advanced modeling and information analyzing and processing techniques that should take into account some aspects, such as the integration of a large amount of data, the behavior of the system environment, the available data sources and the emerging paradigm related to the intelligent systems. To this end, we propose an approach based on the use of the agent technology and big data concept. For the air quality data collection and analysis, we use a Hadoop framework: HBase for data storage and a MapReduce based forecasting process; artificial neural network (ANN) based prediction and K-means as clustering algorithm. Finally, the approach is validated by a case study in which an air quality management support system for the Marrakech city is presented.

Keywords: Decision support systems, Agent technology, Air quality management, Hadoop MapReduce, Artificial neural network, Big Data.

1 Introduction

The continuous increases in productivity bring damage to the environment, due to the various factory emissions, vehicle exhausts and other pollution sources. In response to this concern, several studies on air quality management using forecasting and prediction based solutions have been done [1,2]. Therefore, this problem can be controlled by monitoring and alert forecasting, in the context of scientific researches which is the aim of the proposed system. Also, the solution to growing volumes of data that demand fast and effective retrieval of information is related to the integration of the principles of data mining over a distributed environment. The main objective of this paper is to suggest a solution applicable to large scale data and gives a great flexibility and speed to perform prediction and forecasting over a distributed framework. For this we propose a development approach, based on the use of multi-agent systems (MAS), a Chemistry-transport model (CTM) [1] and an ANN [3] over the Hadoop MapReduce framework [4,5]. The proposed approach will be validated with a case study in which an air quality management system will be presented. This system is used in order to perform predictions and forecasting for air quality and monitor the level of pollution, to ensure conformity with the local legislation and to evaluate control options.

The proposed approach is illustrated over a few sections starting with a brief literature review followed by the system overview in section 3. Section 4 is devoted to the development process presentation. The resulting multi-agent system is described Section 5. Section 6 and 7 are dedicated to the data modeling details and the MapReduce based data analysis process description. In section 8 a case study and the experimental results are presented, followed by a conclusion and perspectives in section 9.

2 Related Work

Many research projects have worked on air quality management and assessment systems such as [6] where the suggested system is designed to give a support in decision making, connected with air quality forecasting and managing models. Such system ensures an assessment of air pollution and allows predicting the air quality in diverse urban situations. Several studies have also investigated the knowledge discovery aspects of analyzing data collected from sensor networks. As example Li et al. [7], investigate a method of analyzing and monitoring data produced by different sensors distributed over Taiwan. The system allows investigating the use of a larger variety of data analysis components.

Other projects have also addressed the issue of air quality’s data integration; like Appetise project [8] that aims to produce a database containing pollution data combined with other related data such as weather records, and to develop tools for analyzing and visualizing this data. The TimeMap project [9] has also developed data analysis software that allows visualization of distributed spatiotemporal data sets, and interactive maps.

Information system enhancement using Hadoop as a data hub to optimize the decision making infrastructure is a new emerging strategy. Many research works have proposed a method to leverage the Hadoop framework by effectively integrating it to the existing data warehouse such as [10] that proposed a study on big data integration with data warehouse built using relational technology mainly for operational sources.

Concerning the use of MAS, we are based on works done by the authors in [11] which propose an agent-based decision support system development approach where the software agents use data mining methods for knowledge discovery, which will be used as a foundation for decision making and recommendation generation. This system provides all the necessary steps for a standard decision making procedure using intelligent agents [12].

3 An Overview of the System

In this system we propose the use of a chemistry-transport model [1] for the air quality estimation and modeling. For the data analysis, we use an online analytical processing (OLAP) tool. We also proposed the use of an ANN based predictions algorithm [3]. Concerning the data gathering, cleaning and integration, we use a Hadoop and MapReduce based process to make needed algorithms applicable to large scale data and give a great flexibility and speed to execute a process over the distributed framework. The aimed system is composed of the following set of components (see Figure 1):

- Monitoring data integration component (MDIC)
- External data integration component (EDIC)
- CTM and Prediction component (CTMPC)
- OLAP component (OLAPC)
- User-interface component (UIC)

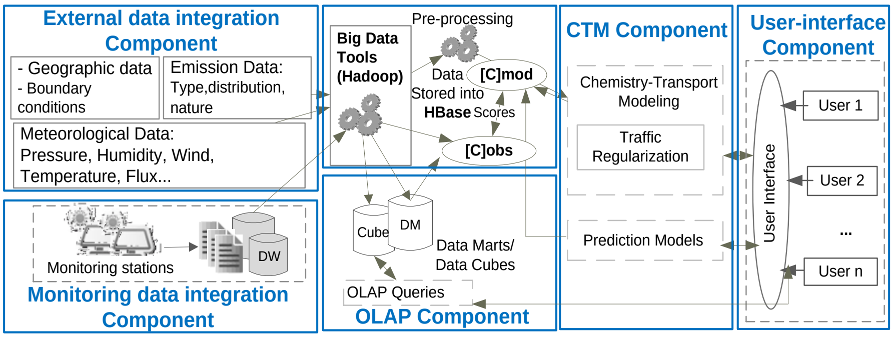


Fig. 1. Components of the air quality management system

4 The System Development

In order to provide an adequate solution in terms of robustness and agility, we use a multi-agent framework to represent the decision support system components. The objective is to propose an architecture that consists of a set of autonomous agents able to set their own goals and actions and interact and collaborate with each other through a communication protocol [13]. The proposed system components are structured into agents. These agents should be identified during the development process.

4.1 The Development Process

In the development process, we are based on the MDA [14] Paradigm and Prometheus methodology [15] which has been developed to support the complete software development lifecycle from problem description to implementation. It offers an environment for analyzing, designing, and developing heterogeneous multi-agent systems.

This methodology consists of three phases: System Specification, architectural design and the detailed design.

The MDA based development process is an iterative process, allowing incremental development and provides the rollback possibility to previous phase [14]. It consists of describing the system as different models expressed with various levels of abstraction. In our case the first level of abstraction is the Computation Independent Model (CIM) which corresponds to the analysis and goal capturing stages. The second level is the Platform Independent Model (PIM) in which we define the agent models based on Prometheus concepts. The third level is the Platform Specific Model (PSM). In our case this level is dedicated to the JACK agent files generation. The last development stage is the automatic Java code generation from PSM. Figure 2, shows the proposed development approach stages.

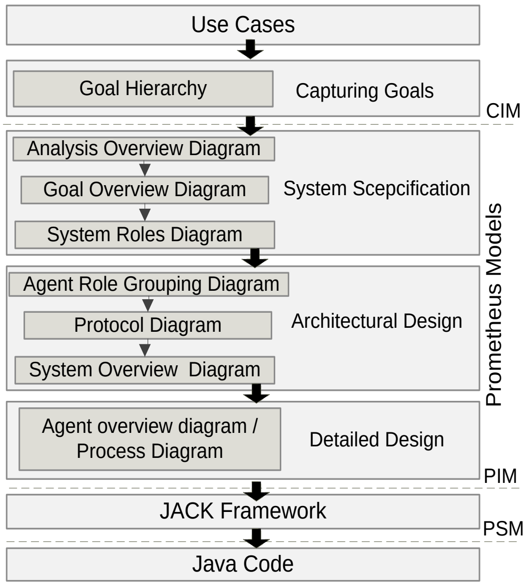


Fig. 2. The modeling methodology phases

4.2 Agents Modeling

In the following sub-sections we describe the different modeling stages in which we use the Prometheus Development Tool (PDT) [15] to apply the proposed development approach and generate the different needed diagrams and models.

The Domain Analysis. The first modeling step is the global domain analysis and the identification of the different use cases and actors. These elements are then used in the goals capturing stage, which consists in defining the user and system general goals (external and internal goals). Figure 3, shows the resulting goal hierarchy diagram.

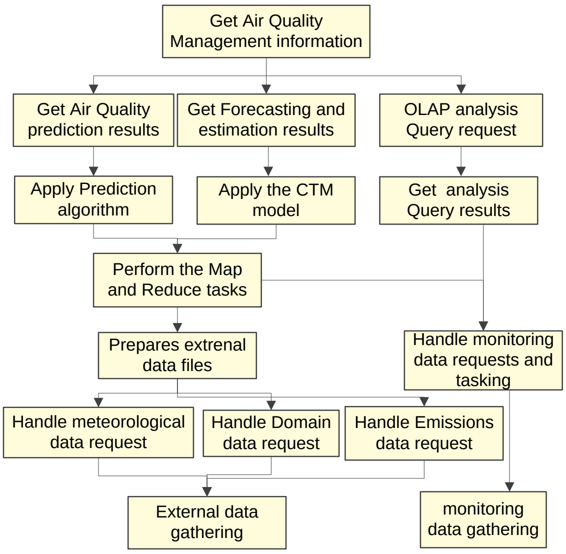

Get Air Quality Management information
Get Air Quality prediction results
Get Forecasting and estimation results
OLAP analysis Query request
Apply Prediction algorithm
Apply the CTM model
Get analysis Query results
Perform the Map and Reduce tasks
Prepares extrenal data files
Handle monitoring data requests and tasking
Handle meteorological data request
Handle Domain data request
Handle Emissions data request
External data gathering
monitoring data gathering

Fig. 3. Goal Hierarchy Diagram

After the goals capturing, we can establish the PIM using the PDT and based on the Prometheus models.

The System Specification. In this stage we use the Prometheus analysis overview diagram. This diagram is designed to show the interactions between the system and the environment. At this abstract level we have to identify the actors, scenarios, percepts and actions through two steps: The actors and the scenarios identification and then the actions and percepts between the actors and the system definition. Figure 4, illustrates a part of the system analysis diagram. Each specified scenario in this diagram must then be associated with a goal using a scenario diagram which represents the scenario aims. The analysis stage is followed by the goal overview diagram. In this diagram, from each high-level goal several sub goals can be defined.

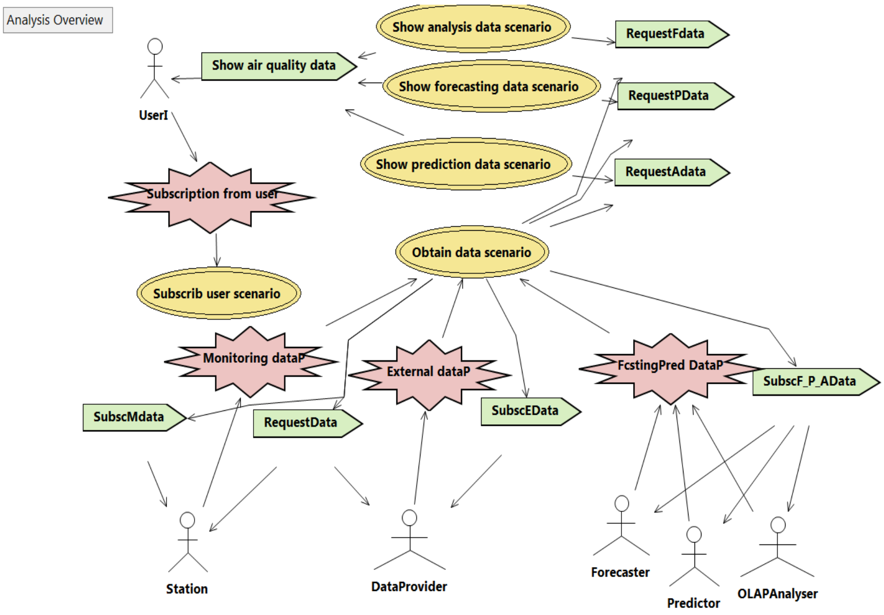


Fig. 4. The system analysis diagram

The Architectural Design. The next step is to transform the structured goals into roles which are the building blocks used to define agent's classes. After roles are created, tasks are associated with each role and gives details about how the goal is accomplished. The agent classes are then identified from the component role. Furthermore, a Prometheus social diagram can be used to represent each agent, the beliefs they have about the environment, the set of goals and sub-goals, and the different plans to achieve them [15].

The Detailed Design. Once we have established the complete roles diagram, we can use the Architectural Design phase in order to group roles into agents using the Agent Role Grouping diagram and introduce and develop agent interactions using the protocol diagram and the system overview diagram.

4.3 Code Generation

The Prometheus Development tool is extended with the ability to generate skeleton code in the JACK agent-oriented programming language [16] using a PDT code generator extension which maintains also synchronization between the generated code and the design when either of them changes.

5 The Resulting Multi-agent System Structuring

According to the modeling process we can assign each generated agent to the suitable component. Table 1, Shows the different agents assigned to system components and Figure 5, illustrates an overview of the different agents and their interactions.

Table 1. The system agents

Components	Agents
Monitoring data integration component	Station agents community
External data integration component	Data provider agent
CTM and Prediction component	Model agent
OLAP component (OLAPC)	OLAP Agent
User-interface component	UI Agent

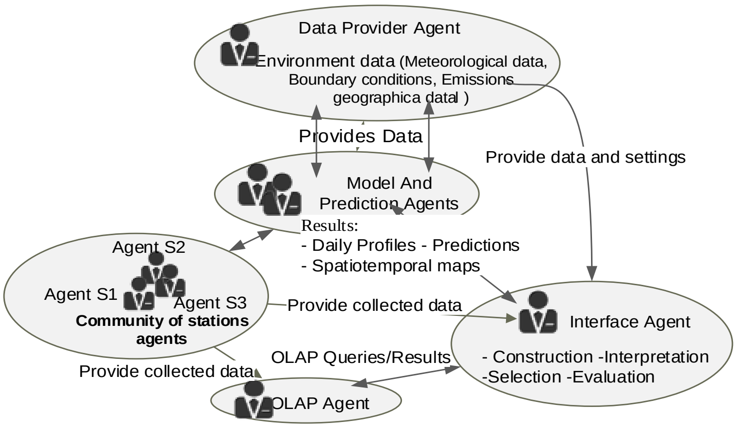

Data Provider Agent
Environment data (Meteorological data, Boundary conditions, Emissions geographica datal)
Provides Data
Provide data and settings
Model And Prediction Agents
Results:
- Daily Profiles - Predictions
- Spatiotemporal maps
Agent S2
Agent S1
Agent S3
Community of stations agents
Provide collected data
Interface Agent
- Construction -Interpretation
-Selection -Evaluation
OLAP Queries/Results
Provide collected data
OLAP Agent

Fig. 5. General architecture of the agent's framework

5.1 Station Agents

These agents represent all monitoring stations distributed in the study area and provide the required functionality during the data extraction, transformation and loading process. The Station agents are used to retrieve data from internal data sources (e.g. Relational databases, and XML/Text files provided by stations) (See Figure 6). A station agent is also responsible for data validation, accuracy, the type conversion, etc. Collaboration between station's agents will allow a better understanding of the spatiotemporal evolution of surface air quality.

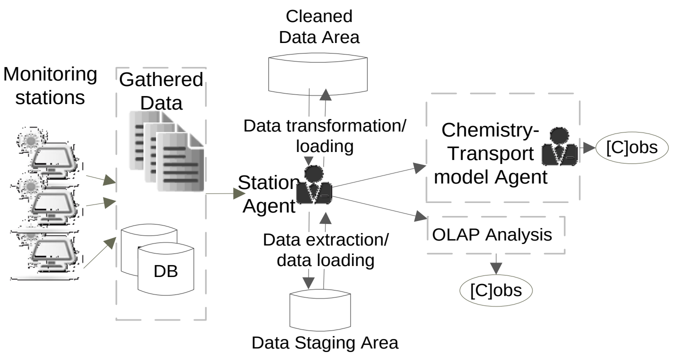


Fig. 6. The station agents functioning

5.2 Data Provider Agent

Responsible for the integration of data gathered from external sources (e.g. MOZART2, LMDZ, WRF, EMEP) [1] and prepares all input data required for the good functioning of the model agent. It sets up a register of emissions for the region in order to make regional modeling and prepare the needed tropospheric emissions data, domain data and meteorological parameters. Figure 7, shows this agent interaction with the other system agents.

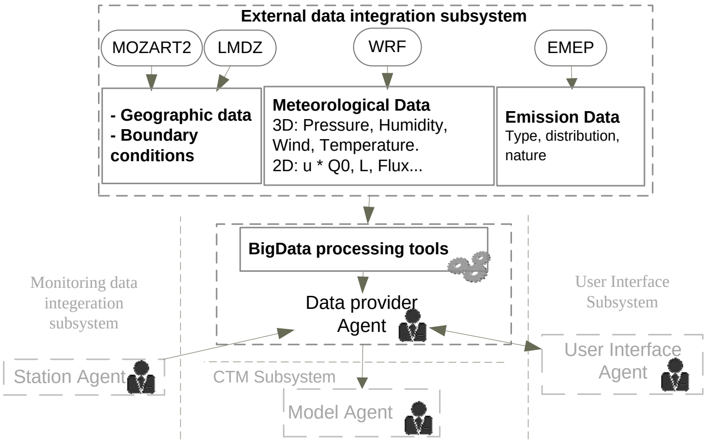


Fig. 7. Data provider agent in interaction with other components

5.3 Model Agent

A Model agent performs the deterministic modeling. It has in charge the functionality corresponding to a chemistry-transport model, which brings together a set of equations representing the transport and chemistry of gaseous species, allowing the quantification of the evolution of a set of pollutants according to time on different domains, taking into account all parameters (e.g. meteorological, boundary conditions, emissions, etc.). This agent uses the resources provided by the Hadoop MapReduce framework [20] in order to explicitly calculate and provide an average concentration over a surface represented by the meshes of his grid. The following figure illustrates this agent functioning.

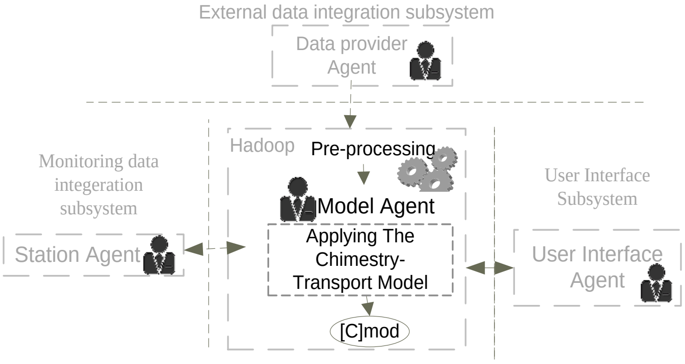


Fig. 8. The model agent functioning and interactions

5.4 Prediction Agent

A Prediction agent is responsible for the generation of the air quality's long-term prediction using an ANN, which is capable of modeling highly nonlinear relationships while taking into account the data distribution factors. The strong capability of ANN in predicting fuzzy data and the efficiency of this approach in modeling dynamic systems has promoted their implementation in this work to predict air quality based on gathered data (see Figure 9). Given the big amount of data, training time for ANN is very large. To address this, we use a MapReduce based on an ANN training process since the MapReduce programming model has the ability to rapidly process large quantity of data in parallel. Also, In order to reduce the time cost of data loading, we store the large scale training data-sets in Hadoop HBase [20], and concurrently load one of them into the memory of computing nodes across the cluster when needed [17].

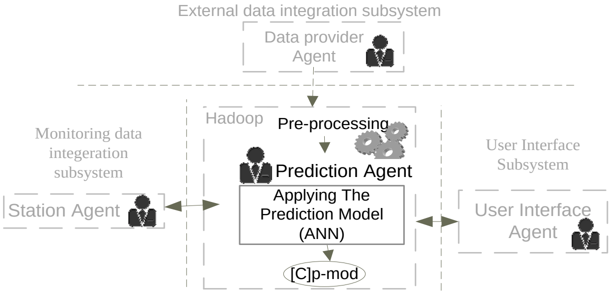


Fig. 9. The prediction agent functioning

5.5 OLAP Agent

The purpose of the OLAP agent is to convert the amount of monitoring data into valuable information by applying quick and effective analysis and create various views and representations of this data. It provides all the basic functionality of an OLAP system and also the missing intelligence in traditional OLAP systems. The aim is performing OLAP analyses on behalf of an agent or a user and reporting its result back to the requesting entity and all other entities that should be informed [18,19].

5.6 User-interface Agent

The user-interface agent enhances the ability of the system user to use and entirely benefit from the DSS. It is responsible for all communications between the air quality management center and the other agents in order to transmit raw data of air pollutant concentrations measured by each station, data gathered by the data provider agent as well as the forecasting and prediction results.

6 Data Modeling

In this work all data are extracted and stored into a Hadoop HBase. HBase is a database with high reliability, high performance, column storage, scalable characteristics based on the Hadoop distributed file system (HDFS). Its goal is the hosting of very large tables with billions of rows and millions of columns atop clusters of commodity hardware [4]. An HBase table is organized as key-value and each table contains a series of row records. Through the HBase feature of column-oriented store and versioning, the time-series data sets are built based on the primary key Row-key and timestamp. The following Figure illustrates a part of the conceptual model concerning the pollutant data and Table 2, shows an HBase table example.

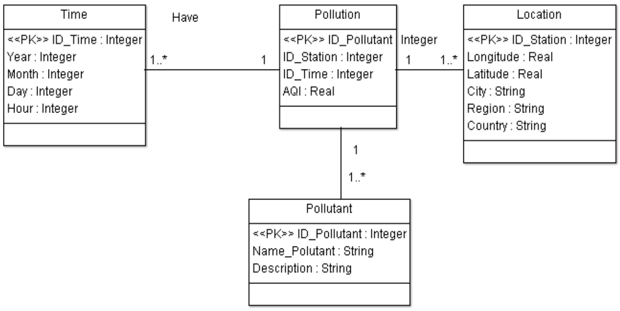


Fig. 10. The air quality management's data warehouse logical star schema

Table 2. The Ozone (O3) table in HBase

RowKey	Column family			Timestamp
	value	type	unit	
31254100302	80.91	O_3	mg/m^3	1237054252

7 Data Analysis

The chemistry-transport estimation process uses a multi-phase MapReduce process to get emissions of various time resolutions [20]. The data are loaded from the monitoring stations, meteorological, geographical and emission databases. First, we perform a data cleaning process using a single MapReduce phase. In the second Map stage (see Figure 10), we use the cleaned data set files to calculate the Atmospheric Pollution Index (API) by applying the Murena [21] method for each pollutant and applying the K-means clustering algorithm [22] for the data analysis. Simultaneously, we use meteorological, geographical, emission and boundary conditions data to generate a spatiotemporal distribution of the pollutants [23]. In the reduce phase, we store intermediate results into the output database.

In the third Map phase (see Figure 11) the forecasting model (CTM) is applied in order to calculate the emission estimation in a given period. The intermediate results use the pair of geographic zone identifier and timestamp as the key map and the amount of emission as the value. In the Reduce phase, the amounts of emissions that have the same key are accumulated together [20].

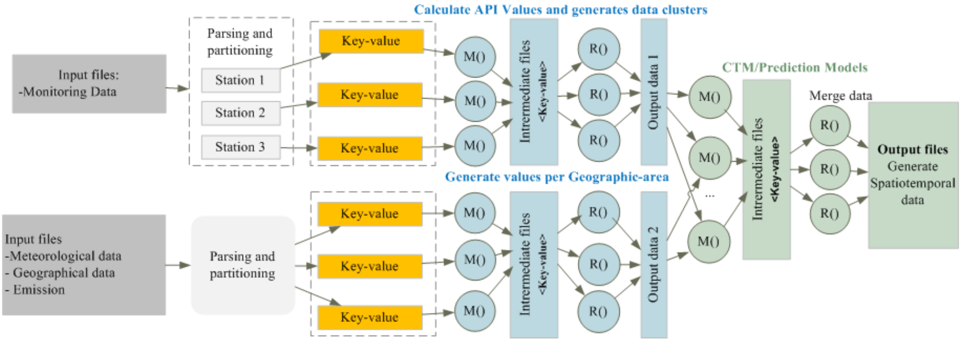


Fig. 11. The second and third Map and Reduce steps

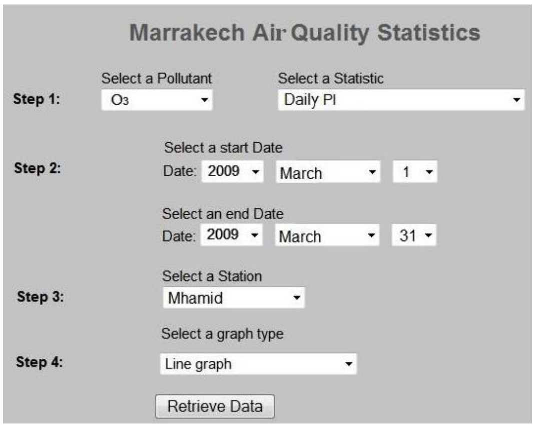
8 Case Study

8.1 Description

In this case we are interested in Marrakech-City, which is not an industrial city, but it suffers from the effects of pollutants produced by vehicle exhaust systems. This study is based on three stations that provide information and measures of the air pollutants concentration (Jamaa EL Fna station, Mhamid station and Daoudiat station). The study focused on the following pollutants: Sulfur Dioxide (SO2), Nitrogen dioxides (NO2), Carbon Monoxide (CO), Particulate Matter, and Ozone (O3).

8.2 Application

Atmospheric Pollution Index Generation. The system generates the analysis queries based on selected options from an easy to use user interface to get the resulting API. Figure 12 shows the user interface to select the Ozone API during March 2009 in the Mhamid station. The result is shown in Figure 13.



Marrakech Air Quality Statistics

Step 1: Select a Pollutant: O3 — Select a Statistic: Daily PI

Step 2: Select a start Date — Date: 2009 March 1

Select an end Date — Date: 2009 March 31

Step 3: Select a Station: Mhamid

Step 4: Select a graph type: Line graph

Fig. 12. Selected options to display the Ozone API

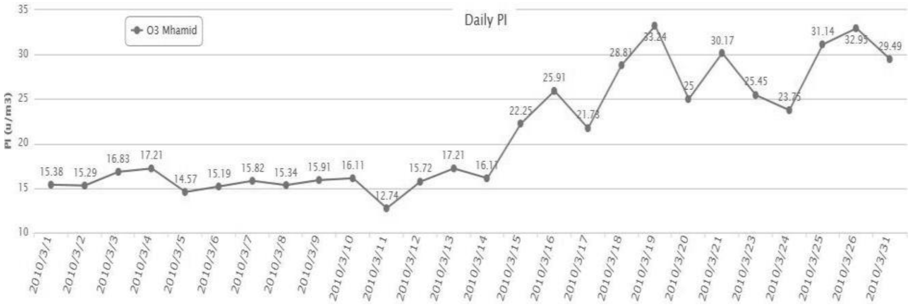


Fig. 13. The Ozone pollution index in Mhamid station results

Air Quality Prediction. We use a three-layer perceptron ANN model and data concerning the study area described above to predict pollutants level. We used six neurons in the input layer including temperature, solar radiation, the NO_2 concentration, CO concentration and the wind velocity. The number of hidden layers and values of neurons in each hidden layer are the parameters to choose in the model construction. Therefore, one or two hidden layers and different value of neurons were chosen to optimize the ANN performance. The last layer is the output, which consists of the target of the prediction model. Here, O_3 was used as the output variable and a hyperbolic tangent sigmoid function was used as the transfer function. A year data set was divided into two parts: 80% used for training the networks and the remaining 20%

employed in testing the networks. The mean square error was chosen as the statistical criteria for measuring the network performance.

The following graph shows the performance of the network above. It represents a comparison between the observed and predicted Ozone concentration based on the mean square error.

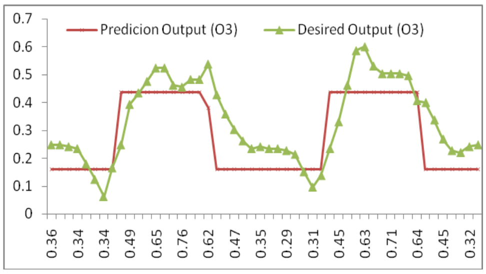


Fig. 14. Model performance of the chosen network

9 Conclusion and Discussion

The main contribution of this work is the definition of a development process based on big data and intelligent systems concepts. We have, through this paper presented the implementation of an air quality management system over a distributed data gathered from different monitoring stations and other external databases and managed by using Hadoop to ensure a fast data loading, fast queries processing and an efficient storage. The Hadoop highly efficient fault tolerant nature, flexibility, extensibility, efficient load balancing and the platform-independent are also useful features for development of any distributed process. We also have adopted an MDA based approach and automatic rule transformations, in order to obtain an adaptive system. The MDA principle is based on reusable model transformations to define specific platform models. Thus, we used an agent oriented MDA approach based on a set of models that are constantly evolving, reflecting current needs and which are associated to a set of agents.

The case study addresses the generation of pollutants API and performing predictions using ANN. Our experimental results show that the algorithms deployed in the large-scale data processing system is feasible and efficient. During the experiment, we found that the data block size impacts the performance significantly. For big number of small data blocks, the processing jobs increases the number of collaboration during the Map and Reduce operation and decrease the performance, since Hadoop has the advantage on handling large size of files.

The perspectives of this work are the integration of multi-criteria decision support tools for decision-making and the use of generated data to address the traffic regulation issue.

References

1. Menut, L., Bessagnet, B., et al.: CHIMERE 2013: a model for regional atmospheric composition modeling. Journal of Geoscientific Model Development. 6, 981-1028 (2013)
2. Ivan, T.C., Žarko, M.C., Vlastimir, D.N., Predrag, M.Ž., Mladen A.T.: Air quality estimation by computational intelligence methodologies. THERMAL SCIENCE. 16, S493-S504 (2012)
3. Russo, A., Raischel, F., Lind, P.G.: Air quality prediction using optimal neural networks with stochastic variables. Atmospheric Environment. 79, 822-830 (2013)
4. Apache Hadoop Documentation, http://hadoop.apache.org.
5. Lammel, R.: Google's MapReduce Programming Model – Revisited. Science of Computer Programming. 70, 1-30 (2008)
6. Fedra, K.: AirWare: an urban and industrial air quality assessment and management information system. SATURN-EURASAP, Urban Air Quality Management Systems, Munich, pp. 73-91 (2002)
7. Li, S.T., Chou, S.W., Pan, S.J.: Multi-resolution spatiotemporal data mining for the study of air pollutant regionalization. In: 33rd Hawaiian International Conference on System Sciences, Island of Maui, Hawaii (2000)
8. Matejicek, L.: Spatial modelling of air pollution in urban areas with GIS: a case study on integrated database development. Advances in geosciences. 4, 63-68 (2005)
9. Johnson, I., Wilson, A.: The TimeMap project: developing time-based GIS display for cultural data. Journal of GIS in Archaeology, 1, 124-135 (2003)
10. Das, K., Mohapatro, A.: A Study on Big Data Integration with Data Warehouse. International Journal of Computer Trends and Technology. 9 (4), 188-192 (2014)
11. Sokolova, M.V., Fernandez-Caballero A.: Modeling and implementing an agent-based environmental health impact decision support system. International Journal of Expert Systems with Applications, 36 (2), 2603-2614 (2009)
12. Gloria, P.W., Jain, L.: Recent Advances in Intelligent Decision Technologies. Lecture Notes in Computer Science. 4692, 567-571 (2007)
13. Lavbic, D., Rupnik, R.: Multi-Agent System for Decision Support in Enterprises. Journal of Information and Organizational Sciences. 33(2à), 269-284 (2009)
14. Mellor, S., Scott, K., Uhl, A., Weise, D.: MDA distilled: principles of Model-Driven Architecture. Addison-Wesley (2004)
15. Padgham, L., Thangarajah, J., Winikoff, M.: AUML protocols and code generation in the Prometheus design tool. In: 6th international joint conference on Autonomous agents and multiagent systems, pp. 1374-1375. Honolulu, Hawaii, (2007) [doi>10.1145/1329125.1329451]
16. Busetta, P., Ronnquist, R., Hodgson, A., Lucas A.: JACK Intelligent Agents - Components for Intelligent Agents in Java. Technical report, Agent Oriented Software Pty. Ltd, Melbourne, Australia (1998)
17. Rong, G., Furao, S., Yihua, H.: A Parallel Computing Platform for Training Large Scale Neural Networks. In: IEEE International Conference on Big Data, pp. 376-384. IEEE, Silicon Valley, CA (2013)
18. Muhammad, S.: Development and implementation of air quality data mart for Ontario, canada: A case study of air quality in Ontario using OLAP tool. Master Thesis (2010)
19. Foster, D., McGregor, C., El-Masri, S.: A Survey of Agent-Based Intelligent Decision Support Systems to Support Clinical Management and Research. In: Proceedings of the Workshop on Multi-Agent Systems for Medicine, Computational Biology and Bioinfor-

matics in association with the 4th International Joint conference on Autonomous Agents and Multi-Agent Systems, pp. 16-34. Utrecht, Netherlands (2005)

20. Junyan, Z., Junkui, Z., Siqi, J., Qi, L., Yue, Z.: A MapReduce Framework for On-road Mobile Fossil Fuel Combustion CO2 Emission Estimation. In: International Conference on Geoinformatics, pp. 1-4. Shanghai (2011)
21. Murena, F.: Measuring Air Quality over Large Urban Areas: Development and Application of an Air Pollution Index at the Urban Area of Naples. Atmospheric Environment. 38, pp. 6195-6202 (2004)
22. Zhao, W., Ma, H., He, Q.: Parallel K-Means Clustering Based on MapReduce. In: Proc. 1st International Conference on Cloud Computing, pp. 674-679. Beijing, China (2009)
23. Wei, F., Sheng, V.S., XueZhi, W., Wubin P.: Meteorological Data Analysis Using MapReduce. The Scientific World Journal. 2014 (2014)