

Decision Support System for Multistore Online Sales Based on Priority Rules and Data Mining

Alessandro Massaro*, Antonio Mustich, Angelo Galiano

Dyrecta Lab s.r.l., IT Research Laboratory, via Vescovo Simplicio, Italy

Received November 11, 2019; Revised December 4, 2019; Accepted December 5, 2019

Copyright©2020 by authors, all rights reserved. Authors agree that this article remains permanently open access under the terms of the Creative Commons Attribution License 4.0 International License

Abstract The work is focused on the design and implantation of an intelligent multi store E-commerce platform able to manage the orders and the warehouse stock by mean of priority association rules and data mining algorithms. The proposed Decision Support System (DSS) is structured into two main levels: the first one is related to priority rule definition due to the online product requests according with the availability check in different warehouses of stores, and the second one provides important information about sales prediction thus facilitating stock management and consecutively logistic. Specifically, the prototype platform is able to manage the warehouse products of different stores by means a simultaneous comparison of products available in the different stores linked to the platform, and by means of a scalable end-to-end tree boosting system XGBoost algorithm able to predict online sales. The paper has been developed within the framework of an industry project.

Keywords Multi E-Commerce, Warehouse Product Management, Data Mining

These factors can lead to definite rules based on the specific application context. Other studies in [2] have identified other factors such as business processes, logistics, and production data for the decision making process which can be implemented into an E-commerce system, by defining criteria and association rules [3]. However, it is essential for a strategic marketing to know all the information related to the complete online supply chain sale service [4]. The decision support model, applied to a multi-channel or multi-platform web service system, has been analyzed in [5]. Even if this model was applied to the manufacturing system, it suggested the design of multiple E-commerce platforms managed by a single Decision Support System (DSS). Data Mining (DM) algorithms are examples of tools that can be implemented to create a DSS system based on the analysis of different inputs [6], thus providing Knowledge Management (KM) and Knowledge Gain (KG). The DSS model is therefore based on digital knowledge, and works into the structuring, understanding and action phases typical of a DSS [7]. A strategic plan based on a distributed system [8], can follows different architectures and can be relevant to both online and offline models [9]-[10]. In [11]-[22] are reported other scientific details relevant to multichannel sales systems, suggesting the preliminary design of the research industry project. Knowledge Gain (KG) represents an important issue a bout Research and Development (R&D) research [23]. In this direction association rules [24] based on priority definition, and data mining tools [25], could provide an improvement of the KG, thus addressing the DSS to the Business intelligence and to the strategic marketing. Following the state of the art and the needs of the industry oriented on online strategic marketing, has been formulated the proposed paper. The paper is structured as follows:

- description of the main specifications of the industry project related to the development of the intelligent DSS multistore E-Commerce platform;

1. Introduction

Decision making process plays an in important rule in E-Commerce systems [1]. However, the same authors of [1] have identified eight main factors that can influence the implementation of an E-Commerce system such as:

- external influences;
- political conditions/situations;
- government initiatives;
- economic conditions;
- socio-cultural conditions;
- technological infrastructures;
- geographical conditions;
- knowledge.

- optimization design of the DSS managing priority rules (DSS level 1) and using graphical dashboards;
- implementation of sales prediction algorithms improving DSS (DSS level 2).

1.1. Main Specifications of the Industry Project

Starting to the DSS concept and to state of the art analysis, have been formulated the preliminary specifications of the project. Specifically, the preliminary project idea is summarized by the architecture shown in Fig. 1 where is sketched an online sales management system involving for the experimental phase nine physical stores (eight stores and one main warehouse). The system is structured as follows (preliminary specifications):

- at least nine prototype E-Commerce systems (one for each store; each store will have the possibility to display only the orders associated with the same store);
- at least nine prototype barcode reading/writing systems (each store, through barcode systems, loads and unloads products having at the same time an exact quantity of warehouse stocks; the registered operations provides important data for the DSS level 1);
- DSS (system capable of extracting useful information for decision-making processes in a short and versatile way; it will contain a database containing the stocks of each store, this update will be carried out periodically for example on a day of the week; upon the receipt of an order, the DSS will be called in order to decide which physical store is able to satisfy the requested order);
- one prototype module able to synchronize the database store systems and integrating marketplaces (such as eBay and Amazon);
- one centralized E-commerce prototype platform with a central MySQL database system;
- one prototype module system coupled to an Enterprise Resource Planning (ERP) management system (for example Danea software based on Magento framework or others) necessary to invoice the products sold, and to provide information about product availability;

- one prototype system to automate shipping processes with couriers (tracking system) and to reduce management time delay due to multiple orders (functionality related to at least nine prototype systems); by means of this system will be easier to enter into the database system the tracking numbers received by the courier for the orders shipped, and will be possible to automatically communicate the data relating to the waybills to the courier.

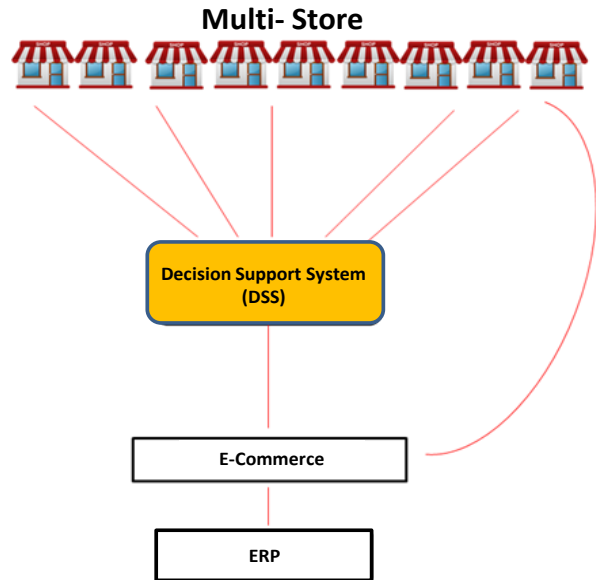


Figure 1. Conceptual scheme of the proposed "DSS MULTI-E-COMMERCE RULES MANAGEMENT" project

The proposed system is managed by a preliminary DSS unit having the logic illustrated by the flowchart of Fig. 2: basing on the policies, on adopted priorities, and on defined association rules, the orders received from the Marketplace and from the E-commerce platform will be managed by the flowchart of Fig. 2. In particular, the logic of the DSS level 1 is focused mainly on the availability check of the required products by analyzing first the availability of store nine and then the availability of the other stores. This logic will be successively improved as a priority rule by selecting the store having the maximum number of the required products (see section 2).

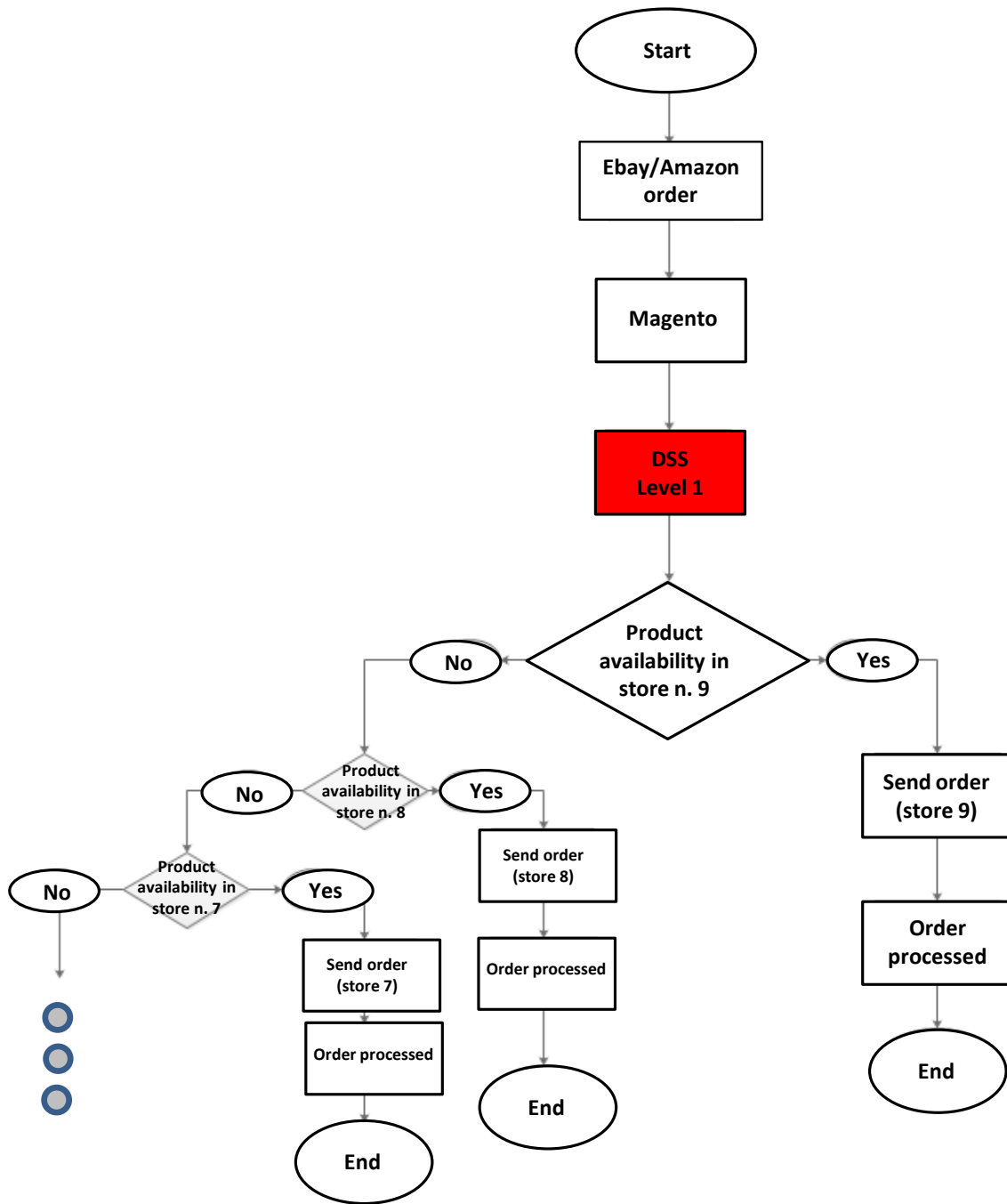


Figure 2. Preliminary DSS level 1 concerning priority rules

2. Platform Design and Methods

In this section is commented the design of the prototype platform by enhancing optimization aspects. In particular will be adopted the Unified Modeling Language (UML) in order to provide the sequence diagram of the whole platform and a flowchart describing the optimized DSS level 1.

2.1. DSS Level 1 Design and Optimization

The main functionality of the platform architecture of Fig. 1 is illustrated by the UML sequence diagram, and Fig. 3 indicate the following actors of the systems:

- Client (user requiring online a product);
- Marketplace/E-commerce system where products for sale are shown;
- Magento engine (open source framework of the multi-store E-Commerce platform);
- DSS (level 1) providing the product scheduling and the product reservation list by adopting priority rules;
- Store;
- Warehouse (location of product stock);
- DHL Server (server used for logistic and product tracking).

The basic functionality of the system is described by the following steps:

1. The user (client) close an order by navigating into the E-Commerce multi-store platform;
2. The E-Commerce provides information and the order type request to the Magento engine embedding the DSS level 1;
3. The DSS level 1 apply the priority flowchart by providing an automatic decision making involving different stores and warehouses;
4. A Server will track the product status (product, sent, product arrived, arrival predicted time, etc.).

The DSS of the content management system (CSM) of the prototype platform (E-Commerce multi-store platform) of Fig. 2 has been optimized by the flowchart of Fig. 4 based on the main searching priority rule of the store characterized by the highest number of products not assigned (not required). The flowchart of Fig. 4 will create a list of the stores ordered by the new priority rules by updating dynamically the store position in the list (in the previous version of the DSS level 1 sequentially each store are checked thus requiring an highest computational cost). Furthermore, the optimized flow chart takes into account other logic conditions such as:

- order received by the platform;
- enough quantity (quantity threshold definition);
- store list empty;
- exclusion of store from the list;
- assignment of all products of an order to a single store (logistic optimization).

Orders are managed according with the following algorithm:

- 1) based on the priority assigned to the stores, it is automatically checked the store containing all the products assigned to the order being processed;
- 2) if there are one or more stores satisfying a specific request, it is chosen the one with the highest priority, otherwise it is chosen the store that contains the most products having the highest priority;
- 3) the order will be divided into sub-orders to be assigned to the relative store (each store will only see the sub-orders to manage).

The KG is improved mainly by the logic conditions of the prototype platform thus allowing order process automatism. For a complete discussion in Table 1 are listed all the advantages and the limits of the optimized DSS level 1.

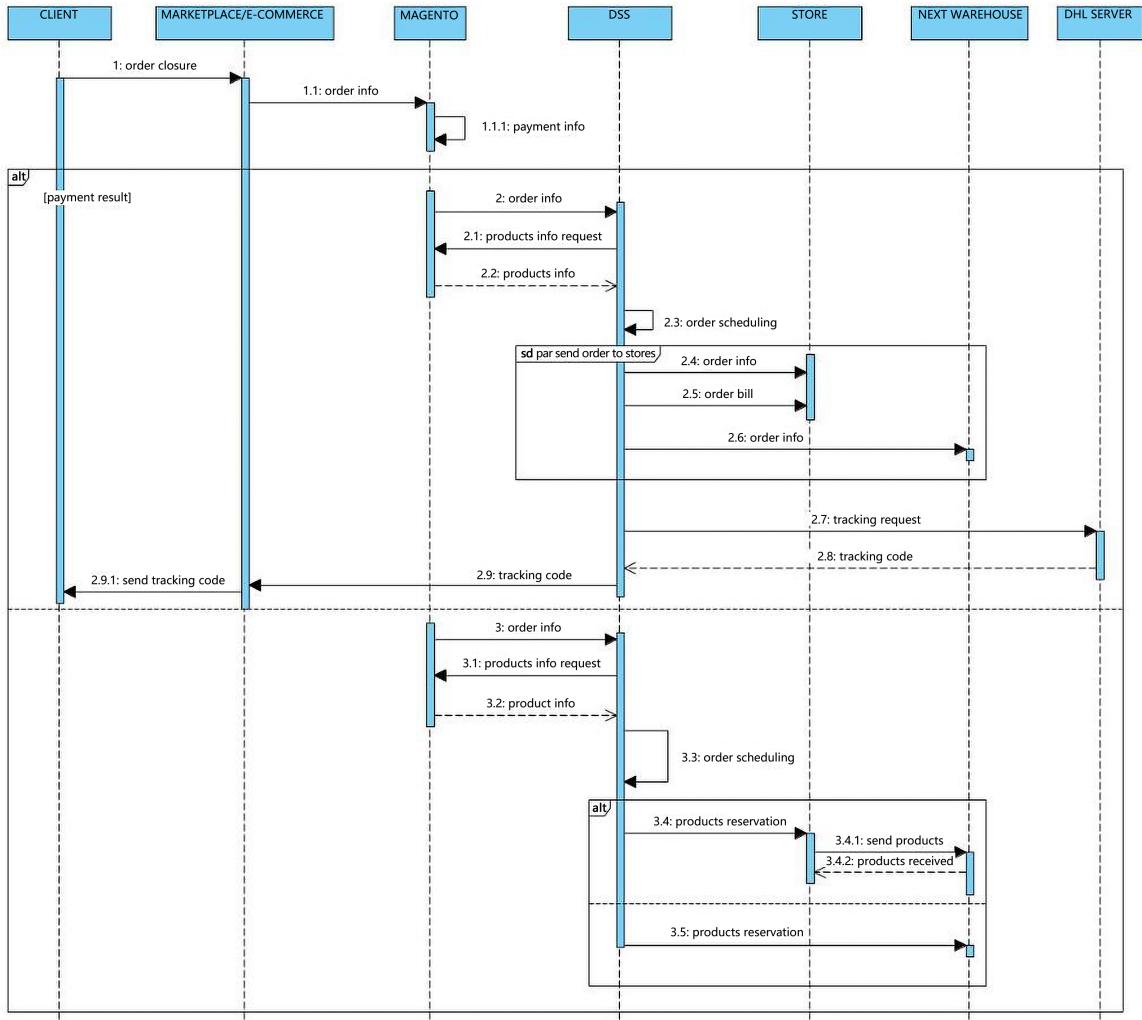


Figure 3. UML sequence diagram of the DSS level 1

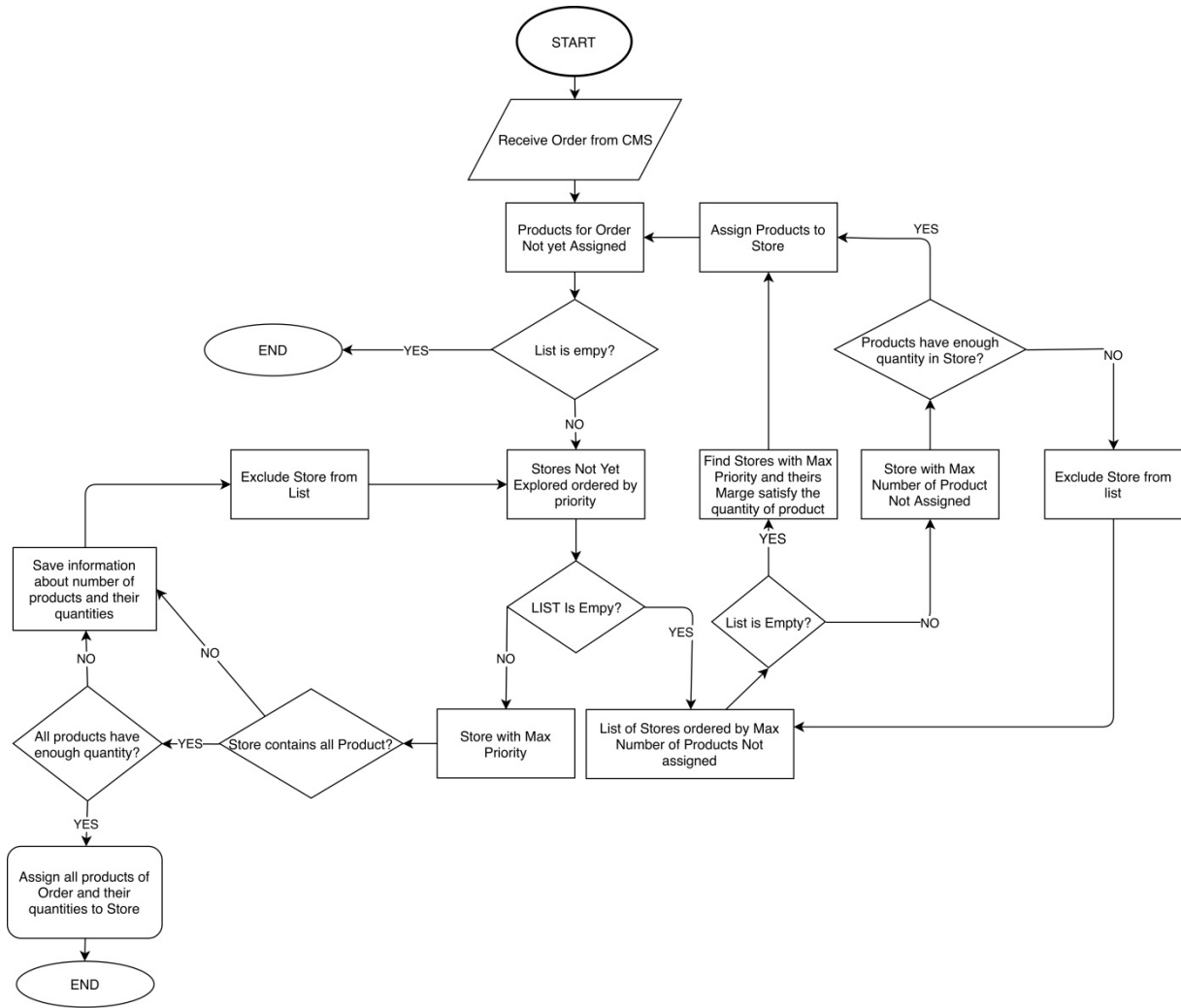


Figure 4. DSS level 1optimization.

Table 1. Advantages and limits of the DSS level 1

Advantages	Related Limit	Note
Full and sure availability of a specified product online required	Synchronization and updating of store database system	The DSS provides every times a store or a warehouse having the requested products. All the databases should be updated by cron jobs and data migration should be synchronized. A central database will collect all the information about product availability
Compatibility with common marketplace	Perfect linking of the multi-store platform with Ebay and Amazon portals	The linking with other common marketplace should be quick and sure. This require a development by Magento tool of a module guaranteeing the full reliability
Speed up of the ordering system	All ERP must be fully interconnected in order to allow efficiently the order automatism	The ERP of a single store must allow to the DSS to work correctly by guaranteeing simultaneously the correct functioning of the local and online system, respectively
Knowledge Gain about customer preferences (intelligent planning of warehouse stocks)	The online preferences could be change during the time. In order to manage correctly the product stock it is necessary also to predict future online sales. The DSS level 2 represents a solution for the sales prediction but must be trusted (a good training model should be created for a correct prediction)	The customer navigation and requests represent an important topic for strategic marketing

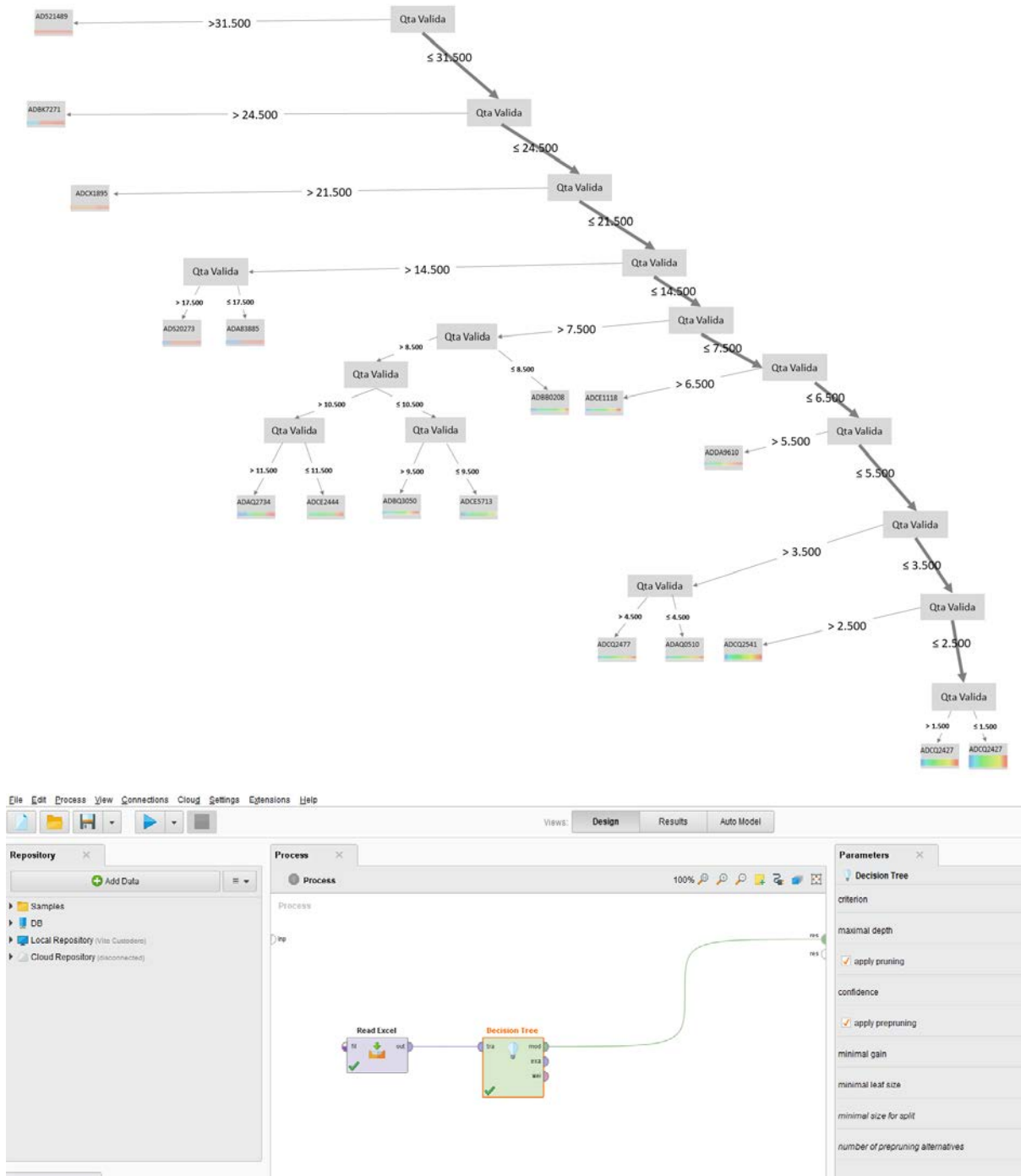


Figure 5. DSS level 1. Above: Decision Tree dashboard representing graphically all product availability. Below Rapid Miner workflow implementing decision Tree algorithm

A graphical dashboard indicating all products availability has been performed by Decision Tree algorithm by using Rapid Miner [25] Decision Tree model (see an example illustrated in Fig. 5). The Decision Tree graphical dashboards, together with the order automatism of the DSS level 1 will allow to control efficiently the warehouse stocks and to plan efficiently the warehouse supply.

2.2. DSS Level 2

As mentioned in the Table 1, the DSS level 2 represents a solution to furthermore improve the whole DSS system. In fact the online sales prediction facilitate the intelligent stock planning by guaranteeing every time the product availability. The prediction evaluation has been performed in the case of study by XGBoost, a scalable machine

learning system for tree boosting [26]-[27]. The implementation of this algorithm is possible by adopting an open source package. The XGBoost DSS level 2 algorithm is implemented by adopting the model of Fig. 6 where the input data are mainly:

- the order processed by the DSS level 1;
- the products ordered during the adoption of the multi-store platform;
- the product category (product classification);
- the store where orders have been executed.

In particular the experimental dataset is composed of the following characteristics:

- quantity of the product sold;
- shop to which the order has been assigned;
- division of the product sold;
- categories to which the product sold belongs;
- color of the product;
- product size.

By considering these attributes will be possible to predict also the product categories.

For the implementation of the DSS level 2 integrating

XGBoost has been adopted the IDE PyCharm, Python 3.6 and Pandas libraries

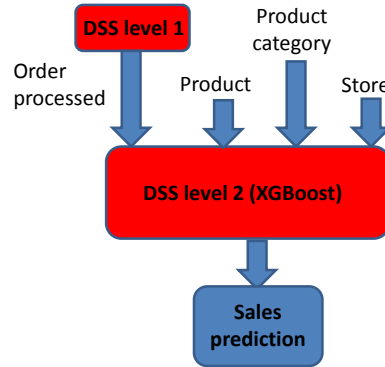


Figure 6. DSS level 2.

In Fig. 7 is illustrated the Graphical User Interface (GUI) of the imported experimental dataset. In Fig. 8 is illustrated a part of the experimental dataset extracted from the central database system having 25209 records.

For a complete discussion, in Appendix are reported different scripts about the XGBoost implementation.

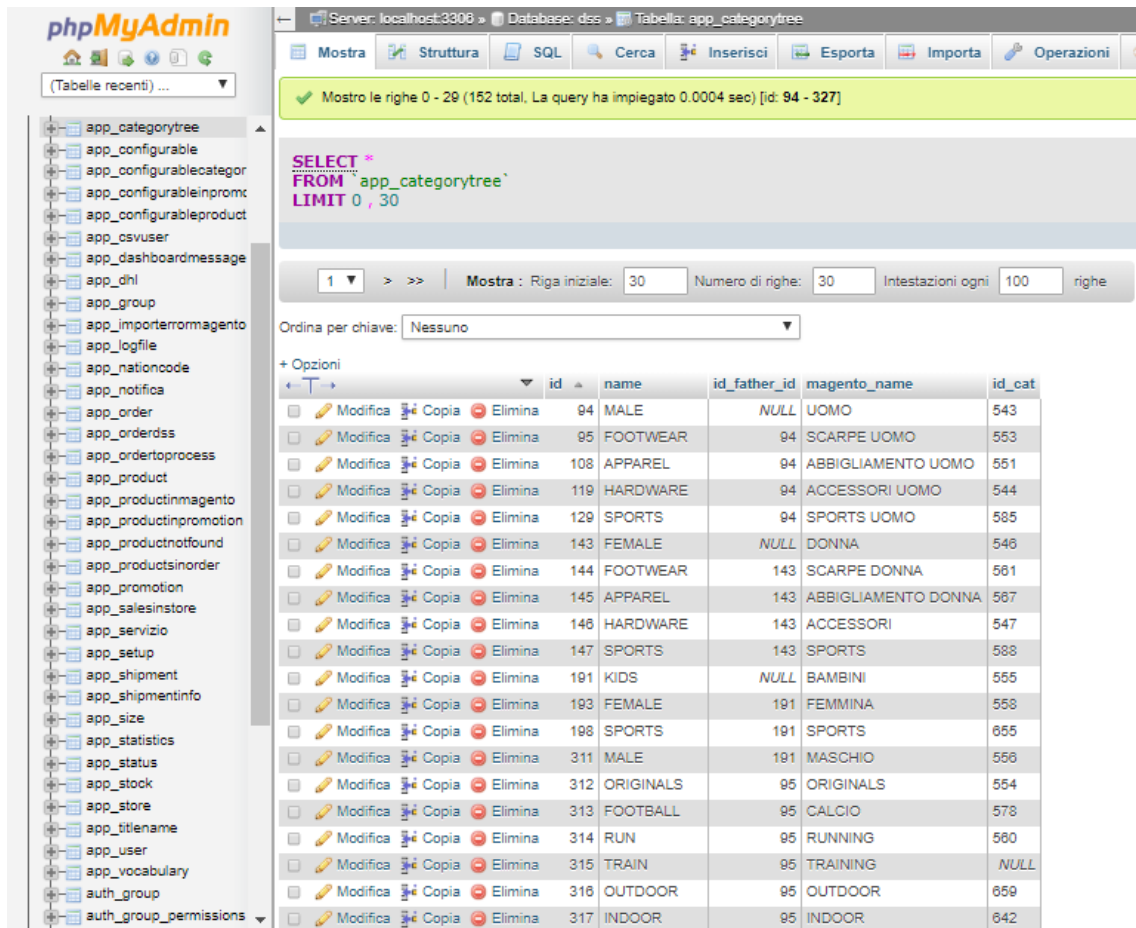


Figure 7. phpMyAdmin interface of the experimental dataset

```
In [14]: df_final
```

	qty	created_at	name	color	size	age	division	categoria	store
0	1.0	2018-12-11	T-SHIRT (SHORT SLEEVE)	BLACK	40	ADULT	APPAREL	MAGLIE DONNA	Centro Comm.le Barblu
1	1.0	2018-12-11	T-SHIRT (SHORT SLEEVE)	BLACK	40	ADULT	APPAREL	SHORTS	Centro Comm.le Barblu
2	1.0	2018-12-11	T-SHIRT (SHORT SLEEVE)	BLACK	40	ADULT	APPAREL	DONNA	Centro Comm.le Barblu
3	1.0	2018-12-11	T-SHIRT (SHORT SLEEVE)	BLACK	40	ADULT	APPAREL	ABBIGLIAMENTO DONNA	Centro Comm.le Barblu
4	1.0	2018-12-11	BALL (LAMINATED)	WHITE/FOOBLU/BRCYAN/C	5	ADULT	HARDWARE	UOMO	Centro Comm.le Barblu
5	1.0	2018-12-11	BALL (LAMINATED)	WHITE/FOOBLU/BRCYAN/C	5	ADULT	HARDWARE	ACCESSORI UOMO	Centro Comm.le Barblu
6	1.0	2018-12-11	BALL (LAMINATED)	WHITE/FOOBLU/BRCYAN/C	5	ADULT	HARDWARE	PALLONI	Centro Comm.le Barblu

Figure 8. Experimental dataset extracted from the central database system.

A root-mean-square error (RMSE) [28] of 0.36 (using a partitioning of 60 % as training dataset and 40 % of testing dataset) proves the good accuracy of the model. Below are indicated the scripts adopted for the RMSE evaluation:

```
rmse = numpy.sqrt(mean_squared_error(y_test, preds))
print("RMSE: %f" % (rmse))
```

The RMSE, measures the difference between values predicted by the model and the observed values. This parameter is then defined as the average difference between two time series $x_{1,t}$ (predicted values) and $x_{2,t}$ (observed values) considered in a period T:

$$RMSE = \sqrt{\frac{\sum_{t=1}^T (x_{1,t} - x_{2,t})^2}{T}} \quad (1)$$

Fig. 9 is illustrated the RMSE percentage trend versus and the dimension of the training dataset: a stable percentage around 0.36 and 0.4 is observed for a dataset training partitioning between 60% and 80 %, thus confirming that the training model is well balanced and well structured (no missing values are checked in the database table and corrected data are inserted as table records).

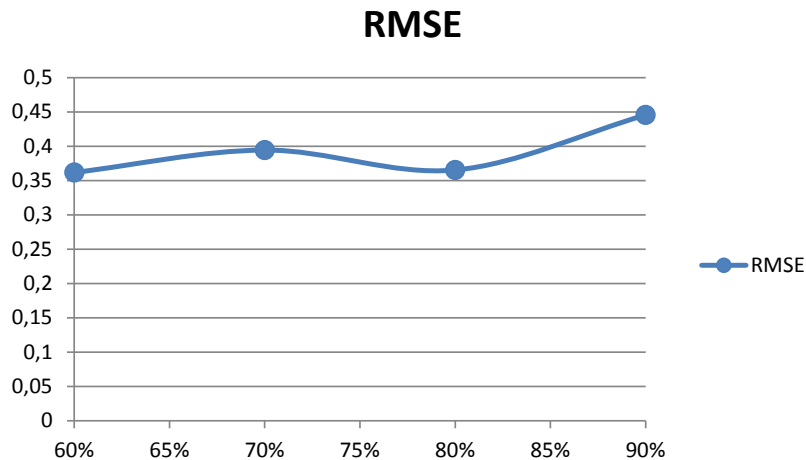


Figure 9. Experimental RMSE percentage versus training dataset partitioning

2.3. Appendix

Below is reported the script about the XGBoost model initialization.

```

import xgboost
import numpy
import pandas
import os
import sys

""" APERTURA FILE DUMP DATABASE """
relative_path = os.path.dirname(sys.argv[0])
import_orders = os.path.join(relative_path, 'app_order.csv')
import_store = os.path.join(relative_path, 'app_store.csv')
import_ordersdss = os.path.join(relative_path, 'app_orderdss.csv')
import_products = os.path.join(relative_path, 'app_product.csv')
import_products_in_order = os.path.join(relative_path, 'app_productsinorder.csv')
import_categories = os.path.join(relative_path, 'app_categorytree.csv')
import_categories_product = os.path.join(relative_path, 'app_categoryassignment.csv')

```

The .csv file are imported by adopting the following Pandas function (file reading and data frame creation):

```

""" LETTURA FILE CSV """
orders = pandas.read_csv(import_orders)
stores = pandas.read_csv(import_store)
order_dss = pandas.read_csv(import_ordersdss)
products = pandas.read_csv(import_products)
products_in_order = pandas.read_csv(import_products_in_order)
category_pattern = pandas.read_csv(import_categories)
category_assignment = pandas.read_csv(import_categories_product)

""" CREAZIONE DATA FRAMES """
orders_df = pandas.DataFrame(orders, columns=['id', 'created_at'])
stores_df = pandas.DataFrame(stores, columns=['id', 'nome'])
order_dss_df = pandas.DataFrame(order_dss, columns=['id', 'qty', 'product_id', 'store_id', 'created_at'])
products_df = pandas.DataFrame(products, columns=['id', 'name', 'color', 'size', 'age', 'division'])
products_in_order_df = pandas.DataFrame(products_in_order, columns=['id', 'quantity', 'product_id'])
categories_df = pandas.DataFrame(category_pattern, columns=['id', 'magento_name'])
categories_assignment_df = pandas.DataFrame(category_assignment, columns=['category_id', 'product_id'])

```

The final dataset has been created by the following Python script (final data frame creation):

```

""" CREAZIONE DATA FRAME FINALE """
categories_df = pandas.merge(categories_df, categories_assignment_df, left_on='id', right_on='category_id')
df_final = pandas.merge(order_dss_df, products_in_order_df, left_on='product_id', right_on='id')
del df_final['product_id_x']
del df_final['id_y']
del df_final['quantity']
df_final.columns = ['id_order_dss', 'qty', 'store', 'created_at', 'product_id']
df_final = pandas.merge(df_final, products_df, left_on='product_id', right_on='id')
del df_final['id']
df_final = pandas.merge(df_final, categories_df, left_on='product_id', right_on='product_id')
del df_final['id_order_dss']
del df_final['product_id']
del df_final['category_id']
del df_final['id']
df_final = pandas.merge(df_final, stores_df, left_on='store', right_on='id')
del df_final['store']
del df_final['id']
df_final.rename(columns={'nome': 'store', 'magento_name': 'categoria'}, inplace=True)

```

3. Conclusions

The proposed work discusses the design and the implementation of a multi store E-Commerce platform based on the DSS working at different levels. A first DSS level (DSS level 1) is able to automate online orders by analyzing the product stock availability of different stores and warehouses connected to the prototype platform. The second level (DSS level 2) allows to optimize the stock planning of each store by means of the online sales prediction by reducing the risk of product non available. In this way the DSS will guarantee every time the product availability. The DSS level 2 is developed by means of the XGBoost algorithm, and represents an element to be considered for the warehouse stock management in medium/long period thus improving business intelligence. A good DSS model accuracy is achieved by using real experimental data of an industry research project. The combination of logic conditions, association rules and data mining algorithms such as XGBoost and Decision Tree will gain the knowledge according to R&D Research guidelines [23]. The application of different methodologies for a specific case of study proves how it is possible to formulate a DSS supporting industry processes associated with warehouse management, stocks optimization and advanced business intelligence. The paper discusses the scientific methodology to improve a DSS based on a multistore management system where priority rules can be formulated by means of an automatic check of available product and by sales prediction. The proposed approach analyzes all the step to implement and apply decision makers for online sales by means of a digital platform managing different store data, association rules based on priorities, and by data mining algorithms a multilevel DSS.

Acknowledgements

The work has been developed within the framework of the project titled: Sistema di Supporto Decisionale (DSS) per la Vendita Online Multistore basato su Regole di Priorità: “DSS Multi-E-Commerce rules management” (Decision Support System for Multistore Online Sales based on Priority Rules “DSS Multi-E-Commerce rules management”). We are very grateful to the following Dyrecta Lab researchers: B. Boussahel, V. Custodero, M. Legrottaglie, S. F. Massari, G. Meuli, L. Pellicani, N. Savino, M. Solazzo, F. Tarulli, E. Valenzano, and V. Vitti.

REFERENCES

- [1] K. A. P. Narayanasamy, M. Velmurugan. Application of decision support system in E-commerce, Communications

of the IBIMA, Vol. 5, No. 1, pp 156-169, 2008.

- [2] P. Suchánek, R. Šperka, R. Dolák, M. Miškus. Intelligence decision support systems in E-commerce, Efficient Decision Support Systems - Practice and Challenges in Multidisciplinary Domains, Prof. Chiang Jao (Ed.) 2011, ISBN: 978-953-307-441-2, InTech.
- [3] R. Šperka, K. Slaninová. The usability of agent-based simulation in decision support system of E-Commerce architecture. International Journal Information Engineering and Electronic Business, Vol. 1, pp 10-17, 2012.
- [4] P. R. Nair, V. Raju. Overview of information technology tools for supply chain management. Communications, Computer Society of India (CSI), Vol. 33, No. 9, pp 20–27, 2009.
- [5] I. Bessedik, N. Taghezout. A multi-agent framework for a web-based decision support system applied to manufacturing system, Proceedings of the 2nd Conférence Internationale sur l'Informatique et ses Applications (CIIA'09), Saida, Algeria, May 3-4, 2009.
- [6] S. Srinivasan, J. Singh, V. Kumar. Multi-agent based decision support system using data mining and case based reasoning, International Journal of Computer Science Issues, Vol. 8, No. 2, pp 340-349, 2011.
- [7] E. De Kock. Knowledge-based decision support systems, Ch.2, University of Pretoria etd., 2003.
- [8] S. D. Pinson, J. A. Loica, P. Moraitis. A distributed decision support system for strategic planning, Decision Support System, Vol. 20, No. 1, pp 35-51, 1997.
- [9] T. Kollmann, A. Kuckertz, I. Kayser. Cannibalization or synergy? Consumers' channel selection in online-offline multichannel systems, Journal of Retailing and Consumer Services, Vol. 19, No. 2, pp 186–194, 2012.
- [10] K. Xie, J. Xiao, Y. Wu, Q. Hu. Ecommerce: channel or strategy? insights from a comparative case study, Proceeding of Thirty Fifth International Conference on Information Systems, Auckland, 2014.
- [11] C. Lazaris, A. Vrechopoulos. From multi-channel to “omnichannel” retailing: review of the literature and calls for research, Proceeding of 2nd International Conference on Contemporary Marketing Issues, (ICCM). 2014.
- [12] R. Yan. Managing channel coordination in a multi-channel manufacturer-retailer supply chain, Industrial Marketing Management, Vol. 40, No. 4, pp 636-642, 2011.
- [13] P. C. Verhoef, P. K. Kannan, J. J. Inman. From multi-channel retailing to omni-channel retailing: introduction to the special issue on multi-channel retailing. Journal of Retailing, Vol. 91, No. 2, pp 174-181, 2015.
- [14] E. Breugelmans, K. Campo, H. Huiying, An empirical analysis of the impact of cross-channel promotions in multi-channel grocery retailing. 2014. Online Available: <https://core.ac.uk/download/pdf/34614316.pdf>
- [15] M. Van Birgelen, A. De Jong, K. De Ruyter. Multi-channel service retailing: The effects of channel performance satisfaction on behavioral intentions, Journal of Retailing, Vol. 82, No. 4, pp 367-377, 2006.
- [16] H. Schramm-Klein. Multi-Channel-Retailing,

Springer-Verlag, 2003.

- [17] P. I. Jeffers, B. R. Nault. Why competition from a multi-channel e-tailer does not always benefit consumers. *Decision Sciences*, Vol. 42, No. 1, pp 69-91, 2011.
- [18] V. Badrinarayanan, E. P. Becerra, C.-H. Kim, S. Madhavaram. Transference and congruence effects on purchase intentions in online stores of multi-channel retailers: initial evidence from the US and South Korea, *Journal of the Academy of Marketing Science*, Vol. 40, No. 4, pp 539-557, 2012.
- [19] G. Wagner, H. Schramm-Klein, S. Steinmann. Effects of cross-channel synergies and complementarity in a multichannel e-commerce system—an investigation of the interrelation of e-commerce, m-commerce and IETV-commerce. *The International Review of Retail, Distribution and Consumer Research*, Vol. 23, No. 5, pp 571-581, 2013.
- [20] The Economist: special report artificial intelligence. Online available from: <https://shop.economist.com/products/special-report-on-artificial-intelligence>
- [21] P. Duchessi, R. O'Keefe, D. O'Leary. A research perspective: artificial intelligence, management and organizations, *Intelligent Systems in Accounting, Finance and Management* Vol. 2, pp 151-159, 1993.
- [22] G. F. Luger. *Artificial intelligence- structures and strategies for complex problem solving,* Pearson, 6th Edition, 2009.
- [23] Frascati Manual. The measurement of scientific, technological and innovation activities guidelines for collecting and reporting data on research and experimental development. OECD (2015), ISBN 978-926423901-2, 2015.
- [24] A. Massaro, P. Lisco, A. Lombardi, A. Galiano and N. Savino. A case study of research improvements in a service industry upgrading the knowledge base of the information system and the process management: data flow automation, association rules and data mining, *International Journal of Artificial Intelligence and Applications (IJAIA)*, Vol. 10, No. 1, pp 25-46, 2019.
- [25] A. Massaro, V. Vitti, A. Galiano, and A. Morelli. Business intelligence improved by data mining algorithms and big Data systems: an overview of different tools applied in industrial research, *Computer Science and Information Technology*, Vol. 7, No.1, pp 1-21, 2019.
- [26] T. Chen, C. Guestrin. XGBoost: a scalable Tree Boosting System, *Proceeding of KDD '16*, August 13-17, 2016, San Francisco, CA, USA.
- [27] R. Bekkerman, M. Bilenko, and J. Langford. *Scaling Up Machine Learning: Parallel and Distributed Approaches*. Cambridge University Press, New York, NY, USA, 2011.
- [28] M. V. Shcherbakov, A. Brebels, N. L. Shcherbakova, A. P. Tyukov, T. A. Janovsky, V. A. Kamaev. A survey of forecast error measures, *World Applied Sciences Journal*, Vol. 24, 2013, pp 171-176.