Predictive Method for Service Composition in Heterogeneous Environments within Client Requirements

Saleh M. Altowaijri[1]
Department of Information Systems
Faculty of Computing and Information Technology
Northern Border University, Rafha 91911, Kingdom of Saudi Arabia

***Abstract*—Cloud computing is a new delivery model for Information Technology services. Many actors and parameters play an important role in provisioning of dynamically elastic and virtualized resources at the levels of infrastructures, platforms, and softwares. Now-a-days, many cloud services are competing and present often similar offers. From the customer side, it is not always easy to select a suitable service according to customer requirements and cloud services scoring. In a real-world scenario, this is more complicated since service scoring may change over time. Besides this, it interferes with many parameters such as hardware, network infrastructure, customer demand, etc. To tackle this issue, this research work presents a novel approach for the prediction of the future score of any service in order to satisfy user requirements when executing service composition in cloud environments. This approach deals with regression techniques in order to predict the expected future offer of service based on sampling service's history as well as user expectations.**

Keywords—Workflow; robust regression; prediction; cloud computing

I. Introduction

The technological evolution of software applications has gone through different architectural styles starting from trivial single-station programs reaching distributed applications augmented with more and more complex structures. These applications are developed in a component-based architecture; the latter represents independent software structures that communicate through interfaces within the same application or through components for distributed applications.

In this context, when companies are accessible over external networks and the components of one are developed and/or used in others, the adoption of distributed workflow of software components becomes crucial [1]. However, this kind of solution, even though using distributed workflows, relies on layers of proprietary infrastructure marked with a strong and sustainable link between offered functionalities and customers. It is noted that built applications based on this approach have become mostly obsolete.

The component-based architecture, in its earlier versions, was incompatible with the openness of the Internet infrastructure. This situation has become more complicated when combined with the globalization of the use of Internet. In fact, the distributed workflow should integrate more component-based applications, web services (seen as heterogeneous components) developed and published by different providers [2].

Now-a-days, web services are marked with a continuous growth in term of offers (provided services) and demands (requested services). Thus, the adoption of a reference architecture allowing the information exchange organization between service clients and service providers becomes a necessity. As result, it is essential to model any distributed workflow by an arrangement of executable offered services. In this case, the arrangement is represented by a four-step web process: development, publication, composition and execution [1]. In this specific context, the composition phase has a highly important role.

The field of services have still been the subject of many researches including, among others, the study and use of the service's concept [3], [4] (composition, orchestration, selection of services, semantic service composition) and service oriented architectures (SOA) (service oriented application design methods, distributed execution control mechanisms, quality of offered services and the security of services, etc. [5]). Particularly, the formulation of user needs is among the most controversial research topics [6], [7], [8]. In fact, a major importance has been given to users, service clients, who are struggling to formalize their increasingly complex needs in an unambiguous way. Faced with the complexity of needs, adopting a basic and composite formal representation will allow customers to capture their specification in a better and more accurate fashion. Thus, it is requested that service composition migrates to variable and dynamic process geared towards meeting client's goals/intentions. This is more accentuated in heterogeneous execution environments such as Cloud Computing [9].

Faced with the heterogeneity of environments, it becomes necessary to seek as much information as possible in order to succeed in modeling the future behavior of these environments. This will help in predicting client satisfaction and therefore selecting more eligible services. In this context, this paper proposes a predictive approach for selecting the adequate schedule of service composition based on customer requirements as well as providers offers that vary over time.

In this paper, the interest is to focus on satisfying customer intentions and demands over an SOA architecture and more particularly when it comes to composite services running on heterogeneous platforms, such as Cloud Computing environments. Therefore, the objective here attempts to answer the following questions: How to customize the selection of the adequate service composition in order to predict, anticipate and look ahead to any possible change in the environments' characteristics?

This paper is organized as follows. The next section presents the concept of service composition. Section 3 presents the regression-based models focusing mainly on the linear and ridge regressions. In Section 4, the approach to predict the suitable service composition over time matching customer requirements is presented. The obtained results are presented and discussed in Section 5. Finally, Section 6 presents the conclusions and future work of the paper.

II. About Service Composition Models

Several models are used to represent service compositions, such as state/transition diagrams [10], activity diagrams and Petri nets [11] or automata [12]. Most of the current works focus primarily on the expression of services through the features they offer. On the other hand, the expression of services in terms that are understandable by the end-user or customers, and even more by a manager, remains an unresolved problem. In this paper, it is believed that the existing formalisms based on activities, states and automata are not the most suitable to capture pertinent information to client. The problem is therefore to find the right matching between software services offered by providers and the requirement of customers.

In this same context, it is noticed that the description of services is essentially captured by some characteristics (networks, hardware and software) generally related to the quality of services required by the client [1] and other scoring measurements. Quality of Service (QoS) defines the ability of services to communicate with external systems in a satisfactory manner.

Several works have focused exclusively on QoS at the network characteristics level [13], [14] while others have focused exclusively on the description of executable services [2, 1]. Therefore, it is important to consider the QoS associated with all the means involved to meet the expectations of the customer.

The objective is to assign the most appropriate service offer to the client. Moreover, ordinary composition models are service-oriented and neglect the role of the environment as well as the customer requirements. From a practical perspective, the service offers may vary over time. In order to ensure a better QoS in terms of satisfying users' intentions and requirements, it is desirable that the system should be able to predict this variation over time. "Predictions", also known as "estimations" or "forecasts", consist in proposing to score a variable in the future, according to a tolerable margin of error, and based on the previous experience of this same variable [15]. There are a few studies [13] focused on the modeling of the composition by integrating the prediction.

The main objective of this paper is to consider the notion of QoS estimation and thereby predict the degree of customer satisfaction based on sampling measurements of the previous experience. It is believed that, such problem has not been resolved in previous research work related to service composition and/or prediction techniques.

III. Regression-based Methods

In statistics, a classical problem is to process data modelled by a series of observations $x_1, x_2, \dots, x_n$ corresponding to the realization of some random variables $X_1, X_2, \dots, X_n$. And, it is interesting to find a theoretical distribution of the vector$(X_k)_{1\le k\le n}$ reflecting the properties of the observations $(x_k)_{1\le k\le n}$.

Concretely, the x_k values are well known. For example, the value of $x_k$may reflect the lifespan of a car engine numbered k. Knowing the law underlying an engine lifespan helps the manufacturer to improve the manufacture of engines. It is possible to estimate some indicators related to x_k, such as the average life of an engine which would be a typical estimation problem with confidence intervals. Then it is important to investigate the validity of the estimation in a context of a test problem. There are several methods for estimating data. In what follows, some of these methods are presented focusing particularly on the regression problems. The general purpose of the regression is to best explain a variable y (the response) as a function of other variables x (vector of explanatory variables, or regressors, or factors).

A. Simple Linear Regression

The data are presented by n pairs of observations; they are in the form of Table I.

If the estimation of the relationship between the two variables y and x_1 is linear, this implies that the variation of one variable is proportional to the other. By considering that one variable explains the other variable, their relation can be put in the form of a linear model: $y_i = \beta_0 + \beta_1 x_{1i} + u_i,\ 1 \le i \le n$ with u_i an error term, β_0 and β_1 some constants; in particular β_1 represents the increase of y by response to a unit increase of x_1.

Generally, β_0 and β_1 are estimated by the "least squares" method which consists in making minimum the sum of the squares of the error terms:

$$S(\beta_0, \beta_1) = \sum_{i=1}^{n} u_i^2 = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_{1i})^2$$

TABLE I. Simple Linear Regression : Presentation of the Data with 2 Variables

observation number	**y**	x_1
1	y_1	x_{11}
2	y_2	x_{12}
3	y_3	x_{13}
n	y_n	x_{1n}

TABLE II. MULTIPLE LINEAR REGRESSION : P-VARIABLE BASED DATA

observation number	y	x_1	x_2		x_p
1	y_1	x_{11}	x_{21}		x_{p1}
2	y_2	x_{12}	x_{22}		x_{p2}
3	y_3	x_{13}	x_{23}		x_{p3}
n	y_n	x_{1n}	x_{2n}		x_{pn}

B. Multiple Linear Regression

Multiple linear regression extends the previously single x_1 regressor by p regressors denoted $x_1, x_2, \dots, x_p$. The data are in the form of n sets of observations of the dependent variable y and p regressors; this set is shown schematically in Table II:

The regression between y and $x_1, x_2, \dots, x_p$ is written as follows:

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \dots + \beta_p x_{pi} + u_i, \qquad 1 \le i \le n$$

The constants $\beta_0, \beta_1, \dots, \beta_p$ are called the partial regression coefficients of the model. The least squares method is used to minimize the sum of squares of residues:

$$S(\beta_0, \beta_1, \dots, \beta_p) = \sum_{i=1}^{n} u_i^2$$

$$= \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_{1i} - \beta_2 x_{2i} - \dots - \beta_p x_{pi}\right)^2$$

C. Ridge Regression

In statistics, nonlinear regression is a form of regression in which the observation data is modeled by a nonlinear combination of model parameters and depends on one or more independent variables. The data are adjusted by a method of successive approximations. In what follows one important nonlinear regression method is presented: Ridge regression.

Ridge Regression is a variant of Multiple Linear Regression that aims to overcome the obstacle of collinearity between explanatory variables. It achieves this result by renouncing the Least Squares method for estimating model parameters. In doing so, it introduces a bias on the parameter estimates. This slight disadvantage is covered by the reduction of parameter variance, and even by the reduction of their Mean Quadratic Error (MQE). This is an illustration of the fact that a biased but low variance estimator may be more efficient than an unbiased but high variance estimator. Ridge regression imposes a penalty on the size of the coefficients of β_i. Indeed, the β_i are taking large values inducing a large variance. By limiting the size of the coefficients, it is estimated to gain in terms of prediction error. For this purpose, the ridge regression minimizes the sum of the residual squares to which is added a term dependent on the norm of the coefficient vector:

$$\hat{\beta}^{ridge} = arg\ min_{\beta \in R^{p+1}} \{RSS(\beta) + \lambda\ ||\beta_i||_2^2\}, \quad RSS(\beta) = ||y - 1_n \beta_0 - X\beta_i||_2^2$$

Theorem: [15] suppose that X is centered and reduced. The $\hat{\beta}^{ridge}$ solution of the problem is given by:

$$\hat{\beta}^{ridge}{}_i = S_\lambda^{-1} X^T \tilde{y}, \qquad \hat{\beta}^{ridge}{}_0 = \underline{y} = \frac{1}{n} \sum_{i=1}^{n} y_i$$

where

$$S_\lambda = X^T X + \lambda I_P \ and\ \tilde{y} \triangleq y - \underline{y}\ 1_n$$

D. Gradient Boosting

Gradient boosting is a machine learning technique that consists in aggregating classifiers (called weak prediction models typically based on decision trees) developed sequentially on a learning sample whose weights of individuals are corrected progressively. Classifiers are weighted according to their performance.

IV. PREDICTIVE APPROACH FOR SERVICE COMPOSITION

The following functions/notations are proposed

- **c(b,p)** is the preferred/required value by the customer for the characteristic **p** satisfying a requirement **b**
- **m(b,p,i,t)** the measurement of the characteristic **p** satisfying a requirement **b** offered by the supplier **i** at the instant **t**

The main idea of our approach is summarized in Fig. 1.

As shown above, gathered information can be divided into two groups of data:

- Static data which are described by the customer requirements. these requirements are defined by c(b,p) and naturally maintain the same value for a long period of time.
- Dynamic data which corresponds to measurements given by m(b,p,i,t) at a certain sampling rate. if it considered that the current measurement of m is captured at t_0, then the previous measurements are considered at t_{0-1}, t_{0-2}, t_{0-3}, etc. while the future measurements are considered at t_{0+1}, t_{0+2}, t_{0+3}, etc.

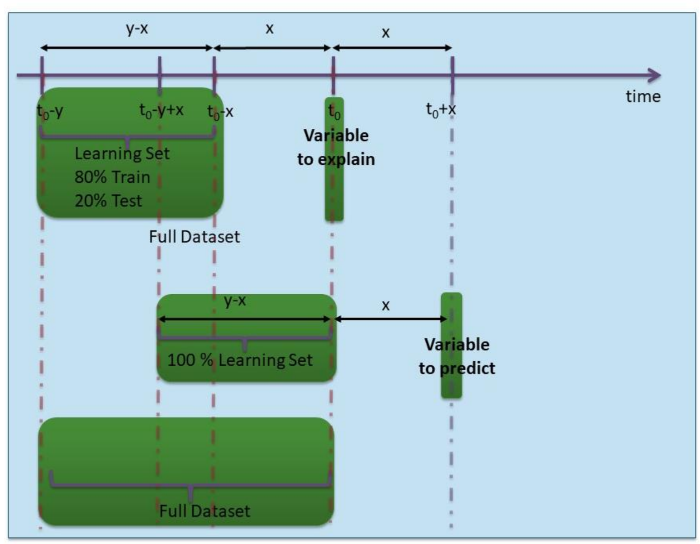

Fig. 1. Two Passes Regression Approach for Service Quality Prediction.

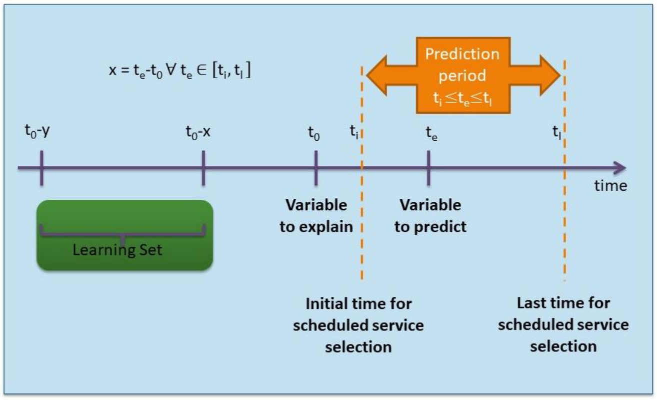

Fig. 2. Adaptation of the Prediction Model to a Scheduled Service Selection.

To seek accurate prediction results, t_0 is estimated by considering for example dynamic data in the instants t_{0-10}, t_{0-9}, t_{0-8}, t_{0-7}, t_{0-6}, t_{0-5}, t_{0-4} in the learning phase. This phase will ensure that the model will learn based on last measurements of the service offer and considering the customer expectations. In the prediction phase, the model will be feed by some shifted measurements; say for example, the instants t_{0-6}, t_{0-5}, t_{0-4}, t_{0-3}, t_{0-2}, t_{0-1}, t_0 to predict t_{0+3}. This concept is summarized in Fig. 2. It is noticed that the size of the window is decided by the expert depending on the available data and their variance in time.

The prediction adopting this approach is estimated to be more accurate and meet with client requirement. This is especially true if an efficient regression method is used to predict the future variable. The next section presents the results of the implementation of this approach.

V. Results and Interpretation

Data is considered as presented in Table X. Two phases are distinguished. The learning phase deals with c, t_{0-10}, t_{0-09}, t_{0-08}, t_{0-07}, t_{0-06}, t_{0-05}, t_{0-04} variables, in order to predict t_0. Data is split into 80% for learning and the remaining 20% for testing the accuracy of the results.

In the prediction phase, the model is fitted by c, t_{0-06}, t_{0-05}, t_{0-04}, t_{0-03}, t_{0-02}, t_{0-01} and t_0 in order to predict t_{0+3} as illustrated in Table III.

Our approach is implemented using python 3.7.3 and the development environment Spyder 3.3.3. The library sklearn is mainly used for implementing the regressors models. In what follows the curves with square lines of the training variables to explain t_0 are illustrated in Fig. 3, 4, 5 and 6.

Mainly, three different regression methods are applied: Gradient boost, Ridge and Linear.

The following indicators to measure the accuracy of the predicted results are used.

- Mean Absolute Error (MAE)

$$MAE = \frac{1}{n}\sum |y - \hat{y}|$$

- Mean Absolute Percentage Error (MAPE)

$$MAPE = \frac{100\%}{n}\sum \left|\frac{y - \hat{y}}{y}\right|$$

- Mean Percentage Error (MPE)

$$MPE = \frac{100\%}{n}\sum \left(\frac{y - \hat{y}}{y}\right)$$

- Mean Squared Error (MSE)

$$\mathrm{MSE} = \frac{1}{n}\sum (y - \hat{y})^2$$

- Root Mean Squared Error (RMSE)

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum (y - \hat{y})^2}$$

TABLE III. Prediction Variables and Phases

		Comment	
Variables		**Training phase**	**Prediction phase**
c(b,p)	c	regressor variables	regressor variable
m(b,p,i,t_{0-10})	t_{0-10}		
m(b,p,i,t_{0-09})	t_{0-09}		
m(b,p,i,t_{0-08})	t_{0-08}		
m(b,p,i,t_{0-07})	t_{0-07}		
m(b,p,i,t_{0-06})	t_{0-06}		regressor variables
m(b,p,i,t_{0-05})	t_{0-05}		
m(b,p,i,t_{0-04})	t_{0-04}		
m(b,p,i,t_{0-03})	t_{0-03}		
m(b,p,i,t_{0-02})	t_{0-02}		
m(b,p,i,t_{0-01})	t_{0-01}		
m(b,p,i,t_0)	t_0	predicted variable	
m(b,p,i,t_3)	t_{0+3}		predicted variable

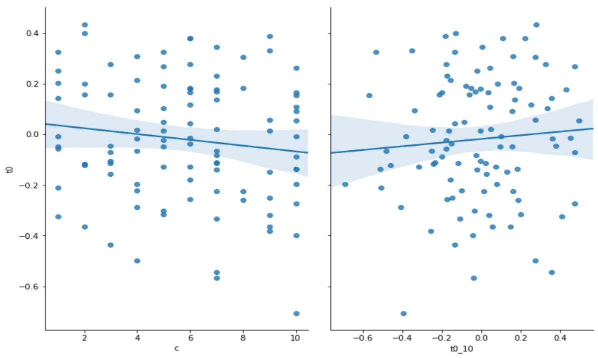

Fig. 3. t_0 Explained by c and t_{0_10}.

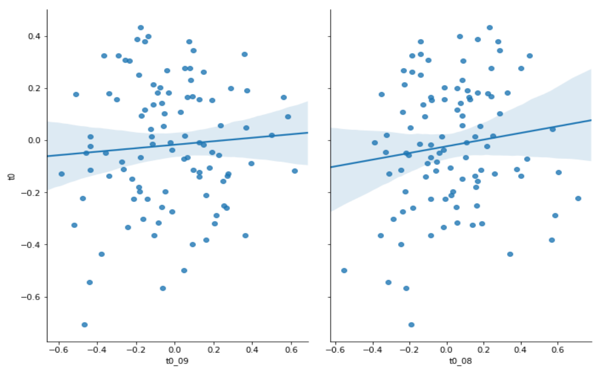

Fig. 4. t_0 Explained by t_{0_09} and t_{0_08}.

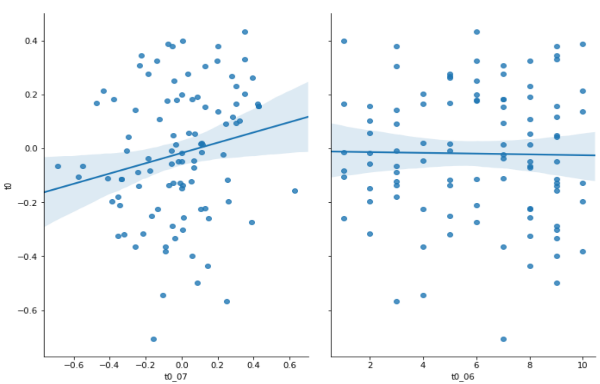

Fig. 5. t_0 Explained by t_{0_07} and t_{0_06}.

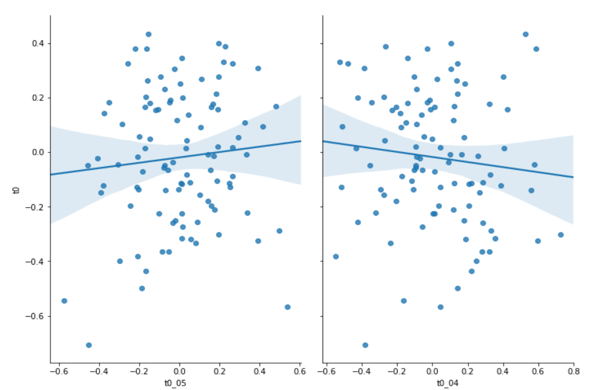

Fig. 6. t_0 Explained by t_{0_05} and t_{0_04}.

A. Linear Regression

In what follows, Table IV summarizes the related error indicators.

TABLE IV. GRADIENT BOOSTING REGRESSOR PERFORMANCE

MAE	0.2523649668
MAPE	2.7418724410
MPE	-0.0067128932
MSE	0.0937286533
RMSE	0.3061513568

In what follows the curves are illustrated in Fig. 7, 8, 9 and 10 to explain t_{0+3} with the different data variables.

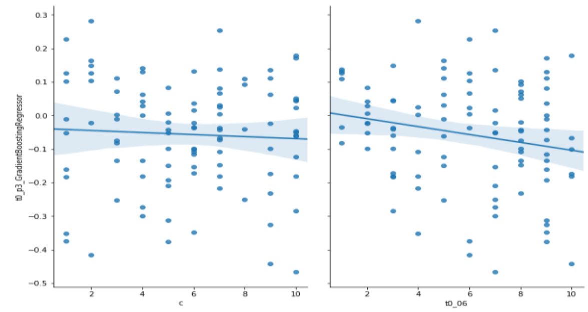

Fig. 7. t_{0+3} (Predicted by Gradient Boost Regressor) Explained by c and t_{0_06}.

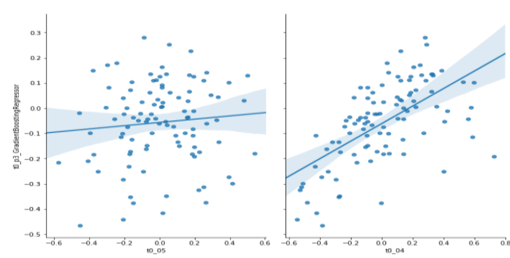

Fig. 8. t_{0+3} (Predicted by Gradient Boost Regressor) Explained by t_{0_05} and t_{0_04}.

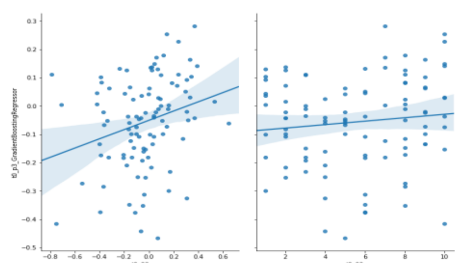

Fig. 9. t_{0+3} (Predicted by Gradient Boost Regressor) Explained by t_{0_03} and t_{0_02}.

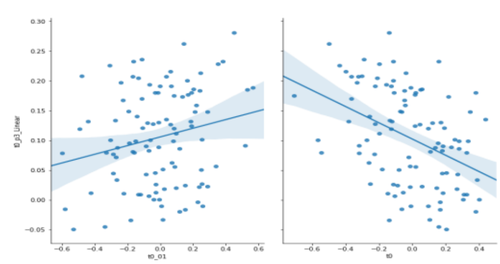

Fig. 10. t_{0+3} (Predicted by Gradient Boost Regressor) Explained by t_{0_01} and $t_{0.}$

B. *Ridge Regression*

The related error indicators are summarized in Table V.

TABLE V. RIDGE REGRESSION PERFORMANCE MEASUREMENTS

MAE	0.2178260161
MAPE	1.1485854694
MPE	0.5720847218
MSE	0.0713895360
RMSE	0.2671882034

In what follows, the curves with square lines of the training variables are illustrated in Fig. 11, 12, 13 and 14 to explain t_{0+3} with the different data variables.

C. *Linear Regression*

In Table VI, the related error indicators are summarized.

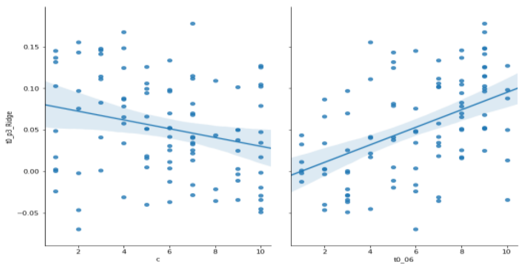

Fig. 11. t_{0+3} (Predicted by Ridge Regression) Explained by c and t_{0_06}.

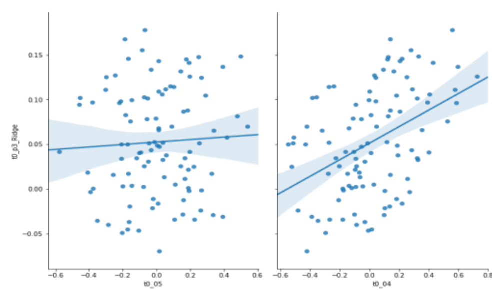

Fig. 12. t_{0+3} (Predicted by Ridge Regression) Explained by t_{0_05} and t_{0_04}.

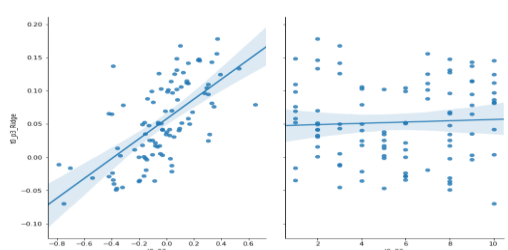

Fig. 13. t_{0+3} (Predicted By Ridge Regression) Explained by t_{0_03} and t_{0_02}.

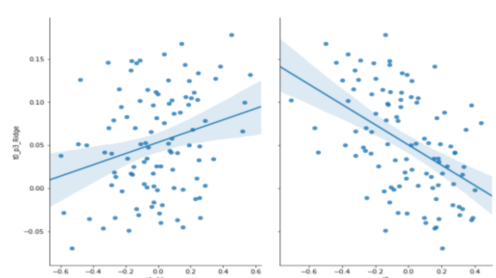

Fig. 14. t_{0+3} (Predicted by Ridge Regression) Explained by t_{0_01} and t_0.

TABLE VI. LINEAR REGRESSION PERFORMANCE MEASUREMENTS

MAE	0.2149517136
MAPE	1.1767757992
MPE	0.5045521139
MSE	0.0698653593
RMSE	0.2643205617

The following figures illustrate the curves with square lines of the training variables in Fig. 15, 16, 17 and 18 to explain t_{0+3} with the different data variables.

Table VII illustrates a resume of the error estimator for the three applied regression methods.

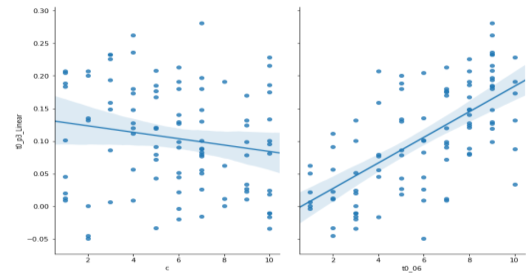

Fig. 15. t_{0+3} (Predicted by Linear Regression) Explained by c and t_{0_07}.

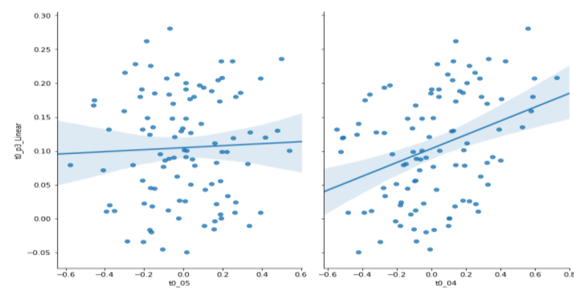

Fig. 16. t_{0+3} (Predicted by Linear Regression) Explained by t_{0_05} and t_{0_04}.

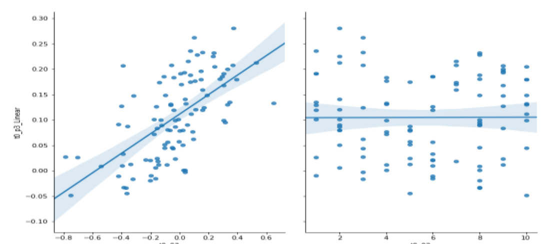

Fig. 17. t_{0+3} (Predicted by Linear Regression) Explained by t_{0_03} and t_{0_02}.

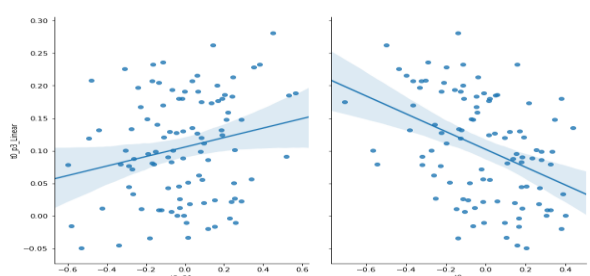

Fig. 18. t_{0+3} (Predicted by Linear Regression) Explained by t_{0_01} and t_0.

TABLE VII. PERFORMANCE MEASUREMENTS

	Gradient Boost	Ridge	Linear
MAE	0.2524	0.2178	0.2150
MAPE	2.7419	1.1486	1.1768
MPE	-0.0067	0.5721	0.5046
MSE	0.0937	0.0714	0.0699
RMSE	0.3062	0.2672	0.2643

In this particular context, the linear and ridge regression presented similar results. Mainly, the ridge regression present better results regarding noisy input data. Therefore, it is believed that ridge regression present in better results when some part of the data is missed or noisy.

VI. CONCLUSION AND PERSPECTIVES

This paper focused on the composition of remote services required by a customer. One fundamental problem is how to select in each intermediate step the most adequate service according to the client requirements expressed in a numeric form. Besides, service offers are measured periodically in a form of global score since service behavior may change over time. However, the future score of a given service cannot be available before its actual measurement. This future score will help the client to select the most convenient service. Thus, a regressive predictive approach is proposed to predict the future score of a service based on the past measurements and the client experience. This approach differentiates between the data variables of the learning phase and the data variables of the prediction phase. A perspective of this work is to focus on the size of the widow related to learning and predicting phases in accordance with data variance over time. This will aim to decide about the most adequate size of data columns in learning phase to seek more accurate prediction results. Moreover, this will be enhanced by a real case-study result.

ACKNOWLEDGMENT

The author gratefully acknowledges the approval and the support of this research from the Deanship of Scientific Research study by the grant no CIT-2018-3-9-F-7953, Northern Border University, Arar 91431, KSA.

REFERENCES

[1] Gil, Y., Gonzalez-Calero, P. A., Kim, J., Moody, J., & Ratnakar, V. (2011). A semantic framework for automatic generation of computational workflows using distributed data and component catalogues. Journal of Experimental & Theoretical Artificial Intelligence, 23(4), 389-467.

[2] A. Alkamari. Composition de services web par appariement de signatures. Technical report, MONTRÉAL,, 2008.

[3] Kurniawan, K., Ekaputra, F. J., & Aryan, P. R. (2018, October). Semantic Service Description and Compositions: A Systematic Literature Review. In 2018 2nd International Conference on Informatics and Computational Sciences (ICICoS) (pp. 1-6). IEEE.

[4] Rao, J., & Su, X. (2004, July). A survey of automated web service composition methods. In International Workshop on Semantic Web Services and Web Process Composition (pp. 43-54). Springer, Berlin, Heidelberg.

[5] Chung, F. I., & Lo, C. L. (2018). Service-Oriented Architecture Application in Long-Term Care Institution: A Case Study on an Information System Project Based on the Whole Person Concept in Taiwan. International Journal of Computing Sciences Research, 1(3), 17-37.

[6] Amina Bourouis, Kais Klai, Nejib Ben Hadj-Alouane and Yamen El Touati “On the Verification of Opacity in Web Services and their Composition”. IEEE Transactions on Services Computing, vol.PP, no.99, pp.1-1 doi: 10.1109/TSC.2016.2605090. http://ieeexplore.ieee.org/document/7558131/

[7] Mahmud, R., Kotagiri, R., & Buyya, R. (2018). Fog computing: A taxonomy, survey and future directions. In Internet of everything (pp. 103-130). Springer, Singapore.

[8] Messaoud, W. B., Ghedira, K., & Halima, Y. B. (2016). Towards behavioral web service discovery approach: State of the art. Procedia Computer Science, 96, 1049-1058.

[9] Ben Halima, Y., Jamoussi, Y., Ben Ghezala, H., & Tata, S. (2012, June). Prediction of cloud environment characteristics to satisfy user requirements in service compositions. In Proceedings of the 2012 IEEE 21st International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises (pp. 84-91). IEEE Computer Society.

[10] M. Mecella B. Pernici. Building flexible and cooperative applications based on eservices. Technical report, roma, italy, 2002.

[11] Ngu A. H. Benatallah B. M. Dumas, Q Z. Sheng. Declarative composition and peer topeer provisionning of dynamic web services. page 297 308, San Jose, California USA,, 2002. 18th international Conference on Data Engineering (ICDE 2002), IEEE Computer Society.

[12] M. Lenzerini M. Mecella Berardi D. D. Calvanese, G. De Giacomo. Automatic composition of e-eservices that export their behavior. Trento, Italy 2003. Proceedings of the 1st International Conference on Service Oriented Computing (ICSOC’03),.

[13] Azween B Abdullah Mohamad Izuddin B Nordin and Mahamat Issa Hassan. Goal based cloud broker for medical informatics application : A proposed goal based request and selection strategy. Department of Computer and Information Sciences Universiti Teknologi PETRONAS Bandar Seri Iskandar, 31750 Tronoh, Perak, Malaysia, 2011 International Conference on Telecommunication Technology and Applications Proc .of CSIT vol.5 (2011) (2011) IACSIT Press, Singapore, 2011

[14] Niraj Tolia Gunho Lee. Topology aware resource allocation for data intensive workloads. ACM 978.1.4503.0195, 2010.

[15] Bernard Delyon. Régression. Technical report, 2019.5