Personalized IaaS Services Selection Based on Multi-Criteria Decision Making Approach and Recommender Systems

Hamdi Gabsi*, Rim Drira†, Henda Hajjami Ben Ghezala‡
RIADI Laboratory. National School of Computer Sciences
University of Manouba,
la Manouba, Tunisia
*Email: *hamdi.gabsi@ensi-uma.tn, †rim.drira@ensi-uma.tn, ‡henda.benghezala@ensi-uma.tn*

***Abstract*—Cloud computing is becoming tremendously popular owning to its advantages such as elasticity, availability and on demand computing. Actually, the number of cloud providers and their offered services is rapidly growing, in particular for Infrastructures as a Service (IaaS). A huge number of IaaS providers and services is becoming available with different configuration options including pricing policy, storage capacity and computing performance. Therefore, IaaS provider selection and services configuration require a high level of expertise. For these reasons, we aim to assist beginner users in making educated decisions with regard to the technical needs of their application, their preferences and their previous experiences. To do so, we propose a hybrid approach merging both Multi-Criteria Decision Making Methods and Recommender Systems for IaaS provider selection and services configuration. Our solution is implemented in a framework called IaaS Selection Assistant(IaaSSelAss); its effectiveness is demonstrated through an evaluation simulation.**

Keywords*- *IaaS services selection; Recommender Systems; Multi-Criteria Decision Making;

I. Introduction

Cloud computing is a model for enabling ubiquitous, convenient and on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services). These resources can rapidly be provisioned and released with minimal management effort [1]. Particularly, Infrastructure as a Service (IaaS) offers highly scalable resources that can be adjusted on-demand. Due to the increasing number of IaaS providers and their heterogeneity, selecting the appropriate IaaS provider is a challenging task. In fact, each IaaS provider offers a wide range of resources and services which must be appropriately selected and correctly configured. This diversity leaves users in the agony of choice and lead to a steep documentation curve to compare IaaS providers and their services. Thus, it is crucial to assist cloud users during their selection process. In this context, several works such as [2]-[4] have shown an interest to address IaaS selection difficulties. However, these works focused mainly on assisting IaaS services selection based on technical application requirements and Quality of Services (QoS) (which we call application profile). Few studies have highlighted the importance of involving the user in the selection process by taking into account his preferences and his previous experiences (which we call user profile). Consequently, there is a need for a selection process centered on both user and application profiles.

In this paper, we propose a hybrid approach based on Recommender Systems (RS) and Multi-Criteria Decision Making Methods (MCDM). RS are programs which try to recommend suitable items (e.g., movies, music, books and products in general) to a given user by predicting his interest in items [5]-[7]. RS predict and provide relevant recommendations according to user's profile and based on rating given by other similar users profiles. Our solution detailed in this paper for assisting the choice of IaaS providers is based on applying recommendation techniques. Once the IaaS provider is chosen, the user needs to be assisted to handle the services selection and configuration. For us, the cloud services selection is a MCDM problem [3][8][9]. MCDM can be defined as a process for identifying items that match the goals and constraints of decision makers with a finite number of decision criteria and alternatives [9]. In our work, we consider IaaS Service selection as a MCDM problem since users have to make a decision to select a service amongst several candidates services with respect to different criteria. We study and choose the adequate MCDM technique to assist IaaS services selection. So, our approach aims to assist IaaS provider and services selection by involving the user in the selection process and by combining RS and MCDM techniques.

The contributions of this paper can be summarized as follows:

- Defining a classification for relevant criteria that should be used during the selection process. These criteria take into account both applications profile including functional and non functional requirements and user's profile including personal preferences, previous experiences and even lessons learned from experiences of other users.
- Presenting a new hybrid approach based on MCDM and RS techniques for IaaS provider and services selection.

- Implementing this approach in a framework which we term IaaSSelAss for IaaS providers and services selection.

The remainder of this paper is organized as follows: Section 2 summarizes existing IaaS service selection techniques; Section 3 discusses these techniques; Section 4 presents our contributions; Section 5 illustrates an evaluation simulation to showcase the working of our approach and finally Section 6 provides concluding remarks and outlines our ongoing works.

II. Related Work

Our work has taken shape in the context of a rich literature focused on simplifying the IaaS services selection with respect to application requirements. We present a classification of recent research approaches for IaaS services selection inspired from [8].

A. MCDM-based approaches for cloud service selection

Over the years, MCDM has emerged as an important research area having immense practical significance in numerous scientific and engineering problems. This technique can be defined as a process for identifying items that best fit the goals and constraints of decision makers with a finite number of decision criteria and alternatives [8]. The most popular MCDM methods used for cloud service selection are the analytic hierarchy process/analytic network process (AHP/ANP) [10], Multi-Attribute Utility Theory (MAUT) [11], and Simple Additive Weighting (SAW) [8].

Several research studies used MCDM based approaches for cloud service selection. We focused on Zia et al. [9] who propose a methodology for multi-criteria cloud service selection based on cost and performance criteria. The authors present this selection problem in a generalized and abstract mathematical form. Table I illustrates the mathematical form. The service selection process is fundamentally a comparison between the vector service descriptor D against all rows of the decision matrix followed by the selection of the services whose description vector best matches with the user's requirement vector.

B. Recommender systems

RS can be defined as programs which attempt to recommend suitable items to particular users by predicting a user's interest in items based on related information about the items, the users and the interactions between them [5]. Generally, RS use data mining techniques to generate meaningful suggestions taking into account user's preferences. Many different approaches using RS have been developed to deal with the problem of cloud services selection.

TABLE I. PROBLEM FORMALIZATION [9]

Mathematical form	Description
Services set	S_1, S_2, ..., S_n A set of services contains all the service offerings from which the user (decision maker) will select the suitable service with regard to his requirements. a service is to be selected by the user (decision maker).
Performance criteria set	C_1, C_2,..., C_n A set of values where C_i represents a criterion that may be a useful parameter for service selection.
Performance measurement functions set	To each criteria C_i there corresponds a unique function f_i which when applied to a particular service, returns a value p_i that is an assessment of its performance on a predefined scale.
Service descriptor (vector)	A row vector D_i that describes a service S_i, where each element d_j of D_i represents the performance or assessment of service S_i under criteria C_j. Performance criteria must be normalized to eliminate computational problems resulting from dissimilarity in measurement units. The normalization procedure is used to obtain dimensionless units that are comparable.
Decision matrix	The service descriptor vectors D_i can be combined to form the decision matrix where each value is the evaluation of the service s_i against the criteria c_j.
User requirement criteria vector	A vector R where each value r_i is the user's minimal requirement against a criteria c_j. These values must be normalized as the vector service descriptor.
User priority weights vector	A vector W where each value w_i is the weight assigned by a user to criteria. c_i

Zhang et al. [6] have offered a cloud recommender system for selecting IaaS services. Based on user's technical requirements, the system recommends suitable cloud services. The matching between technical requirements and cloud services features is based on a cloud ontology. The proposed system uses a visual programming language (widgets) to enable cloud service selection.

Zain et al. [7] propose an unsupervised machine learning technique in order to select cloud services. In fact, the authors classify cloud services into different clusters based on their QoS. The main focus of this study is to offer users the option of choosing a cloud service without engaging into any financial contact. Table II summarizes the most used approaches by identifying the approach's input, the approach's output and the application areas.

The research studies cited previously did not fail to take into consideration the application's functional requirements. Despite the importance of these requirements, we consider

 ISBN: 978-1-61208-651-4

TABLE II. SELECTION APPROACHES

Domain	Method	Input	Output	Application	Literature
Multi-criteria decision-making (MCDM)	SAW	Subjective assessment of relative importance of criteria.	Evaluation value of alternatives.	Applied when requiring low decision accuracy.	[8][9][12]
Multi-criteria optimization	Matrix factorization	Different types of data of interest to users and represented by matrix .	QoS estimation and a set of recommended services.	Applied to a problem that involves different types of data and has missing entries.	[13][14]
Logic based matching approach	First-order logic	Service description and user requirements.	Matched services	Applied to filter out unmatched services to reduce computation complexity.	[8][15]
Recommender System	Collaborative filtering	User's profile	Recommended items	Applied to find personalized recommendations according to user's profile.	[4][6][16]

that this is not enough and users should be more involved in the selection process and hence, their preferences and previous experiences should be taken into account.

To the best of our knowledge, no specific research study has taken into account both the user's profile and the application's requirements. Consequently, there is a need to a structured selection process where clearly both selection criteria are defined and used.

III. HYBRID APPROACH FOR IAAS SERVICES SELECTION BASED ON RS & MCDM

We propose a hybrid approach to assist users in selecting IaaS providers and services based on RS and MCDM. In this section, we start by detailing our selection criteria. Then, we detail our approach.

A. Selection Criteria

In order to recommend the appropriate IaaS services, it is important to specify precise selection criteria. Our purpose is to personalize the selection process according to the user's profile and respond to his application requirements. To this end, we classify our selection criteria into three categories. The first category is the application's profile which includes technical requirements. The second category is the user's profile which represents user's personal preferences and previous experiences. The third category is the previous experiences of other users with their ratings. Figure 1 illustrates our proposed selection criteria. As shown in Figure 1, the selection criteria is classified as the following:

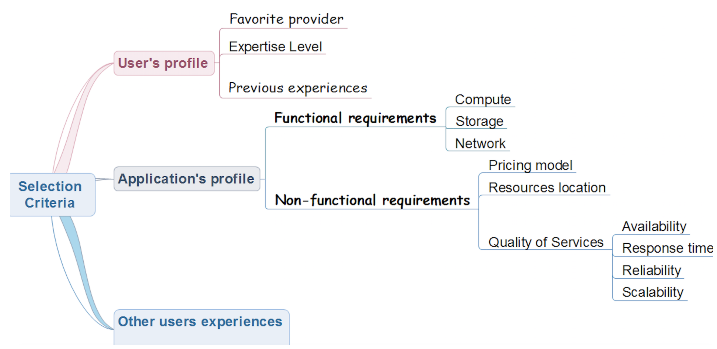

Figure 1. Selection Criteria

- **User's profile:** it includes user's favorite providers, expertise level in cloud and previous experiences. A favorite provider can be chosen based on previous successful experiences using this provider. We take this choice into consideration while identifying the appropriate cloud provider meeting user's requirements. In our case, the user can specify one or multiple favorite providers. The user's expertise level can be: beginner, intermediate or expert. The weight of a user's previous experience in our knowledge base increase with his level of expertise and experience in order to enhance our recommendations relevance A previous experience contains the selected IaaS provider, the deployed application profile and a rating out of 5 presenting a feedback and an evaluation of this experience. We suppose that evaluating ratings are trustworthy and objective.
- **Application's profile:** the application's profile defines the functional and non-functional application requirements.
  Functional requirements are classified into three categories [17].
  - Storage: represents storage needs in terms of memory space.
  - Network: represents connection needs and network usage.
  - Compute: gathers calculation needs and the virtual machine's capacity.

  Non-functional requirements include pricing models, the quality of services (QoS) and the resources location.
  - The pricing model: depends on the user's estimated budget. The pricing model can be on demand, reserved or bidding and can be evaluated per hour or per month.
  - QoS: we focus on the response time, the availability and the reliability. The availability is the time ratio when the service is functional to the total time it is required or expected to function in. The

 ISBN: 978-1-61208-651-4

reliability is represented by the percentage of how long the service can perform its agreed function without interruption.
  - Resources location: The user can precise his nearest resources location because it is important to take into account the proximity when selecting the cloud infrastructure services. According to [15], during the interaction between the users and servers, there is a strong inverse correlation between network distance and bandwidth. Thus, factoring the proximity into the selection of IaaS services can significantly reduce the client's response time and increase the network bandwidth.
- **Previous users experiences:** The more the knowledge base of our recommender system is rich, the more recommendations will be relevant. Therefore, previous users experiences which include deployed application profile, selected IaaS provider and the evaluating rating will improve the accuracy of our recommendations.

B. RS and MCDM based selection approach

The selection of IaaS provider and services configuration is a complex issue. To tackle this issue, we propose a two steps selection process. The first step focuses on selecting the IaaS provider based on the collaborative filtering which is a RS approach. The purpose of this step is to reduce the number of inappropriate IaaS provider which may not interest the user. The second step concerns the configuration of services within the selected provider from the first step. It's based on SAW which is a MCDM method. Our proposed approach shows how MCDM techniques and RS are complementary in order to involve both technical and personal aspects in the selection process.

1) Recommender System: The first step aims to take into consideration the user's preferences, previous experiences and expertise level during the selection process. In our approach, we use the collaborative filtering algorithm also known as k-NN collaborative filtering. This recommendation algorithm bases its predictions on previous users experiences and their profiles. The main assumption behind this method is that other users ratings can be selected and aggregated so that a reasonable prediction of the active user's preferences is deduced.

To recommend the IaaS provider meeting the user's profile we proceed as follows:

- First, we select the users profiles which have the same or higher expertise level than the active user "A". For example, if "A" has the expertise level intermediate, then, from our knowledge base, we select a first list named "list1" of users profiles which are intermediate or expert and their rated experiences.
- Second, among the high rated previous experiences of list1, we select those which are based on the favorite providers of "A" in order to create a second list named "list2".
- Third, among these experiences, "A" can refine list2 by identifying experiences that have similar applications to his application's profile. We obtain list3. Indeed, we aim by these three steps verifying if "A" favorite providers can be suitable for "A" application profile. Otherwise, we skip the second step to apply the third step on list1.
- Then, a rating $R_{(A,f_i)}$ is calculated for each one of candidate providers f_i of list3. $R_{(A,f_i)}$ is calculated as bellow:

$$R_{(A,f_i)} = \frac{\sum_{j=1}^{n} w_{(A,j)}(v_{j,f_i} - \overline{v_j})}{\sum_{j=1}^{n} |w_{(A,j)}|}$$

  where n is the number of identified users' profiles of list3, $w_{(A,j)}$ is the similarity between the profile of "A" and the identified users profiles j of list3, v_{j,f_i} is the rate given by the user j to the provider f_i, $\overline{v_j}$ is the rating's average given by the user j to the favorites providers of "A". We calculate similarity between "A" and the identified users using cosine similarity.

$$w_{(A,j)} = \frac{\sum_{k=1}^{n} v_{A,k} * v_{j,k}}{\sqrt{\sum_{k=1}^{n} v_{A,k}^2 \sum_{k=1}^{n} v_{j,k}^2}},$$

  where the sum on k is the set of providers for which "A" and the selected users in list 3 both assigned a rating, $v_{j,k}$ is the rate given by the user j to the provider k.
- Finally, we propose to "A", the set of providers sorted according to the rate calculated, thus the active user can select one provider.

2) Multi-Criteria Decision Making: Once the IaaS provider is selected, the second step consists on determining suitable IaaS services. In fact, several and conflicting criteria have to be taken into account when making a service selection decision. No single service exceeds all other services in all criteria but each service may be better in terms of some of the criteria. Since users have to decide which service to select amongst several candidates services with respect to different criteria, we consider IaaS Service selection as a MCDM problem. Among MCDM methods, we use the SAW method also known as weighted linear combination or scoring methods. It is based on the weighted average of different criteria. The purpose of using SAW method in our approach is to respond precisely to the application's profile.

The user introduces computing requirements (e.g., virtual Central Processing Unit (vCPU)), storage requirements (e.g., hard drive's size), network requirements (e.g., throughput and bandwidth). The user inserts also the QoS required (e.g., response time and Availability) and the pricing model (e.g., on demand, reserved, bidding).

To be able to apply the SAW algorithm, we need to formalize our decision problem. For that, we define a decision matrix related to the user. In parallel an analogous decision matrix is defined for the IaaS provider selected in the first step. The decision matrix is a combination of service descriptor vectors. Each service descriptor vector represents the performance of a service under a particular criteria. These criteria represent functional and non-functional requirements for the user. For each criterion, the user adds a weight to represent the importance of this criterion. Table III demonstrates an extract form of the decision matrix related to Azure Microsoft [18].

TABLE III. EXTRACT OF DECISION MATRIX FOR MICROSOFT AZURE (VIRTUAL MACHINE)

Service	VCPU	RAM	Hard Drive's size	Cost
A0	1	0,75 GB	19 GB	$0,02/h
A1	1	1,75 GB	224 GB	$0,08/h
A2	2	3,5 GB	489 GB	$0,16/h
A3	4	7 GB	999 GB	$0,32/h
A4	8	14 GB	2039 GB	$0,64/h
A5	2	14 GB	489 GB	$0,35/h
A6	4	28 GB	999 GB	$0,71/h

The SAW algorithm is based on the calculation of one score to each alternative (an alternative in our case is an IaaS service offered by the selected IaaS provider). According to the following SAW formula, the alternative score is calculated as (A_i)=$\sum w_j v_{ij}$, where w_j is the alternative's weight i according to criterion j and v_{ij} its performance. The alternative with the highest score will be suggested. By applying this formula, the recommended IaaS service will automatically be the most performing service, because it has the highest performing values in the decision matrix (highest number of vCPU, largest hard drive's size, highest cost, etc.). However, this does not entirely meet the user's requirements, because, he/she must not necessarily select the most performing IaaS service which will evidently have the highest cost. Whereas, he/she should select the service which meets exactly his/her requirements in order to pay the minimum possible cost. To solve this, we proceed as follows:

- First, we create the decision matrix representing the application's profile by gathering user's functional and non-functional requirements. Then, we determine for each service descriptor vector, the absolute value of the difference between its criteria performance and those of the service descriptor vector related to the IaaS provider. In this way, we will have significant values. In fact, low criteria values mean that they accurately match the user's requirements.
- Second, we calculate the score for each alternative using SAW algorithm. Yet, to be able to do so, we need to modify each criterion's weight to get significant results. Indeed, we have previously mentioned that a low criterion's value means that it may interest the user, if this criterion has a high weight, the multiplication of its weight by its value gives a low score. Therefore, this alternative will be considered as unimportant, yet this is not the case. To solve this problem we take the dual of each weight, meaning that, the subtraction of 1 by the weight's value given by the user. The weight values are between 0 and 1. Consequently, one low weight value indicates a major importance of a given criterion. Thus, we can calculate the score for each alternative using the SAW algorithm. The most relevant alternative (IaaS service) will incontrovertibly have the lowest score.

To illustrate this, we propose our personalized SAW Algorithm 1. We suppose that the cloud user has introduced his decision matrix $UserMat[i][j]$ as well as the weights of each criterion $Weight[j]$. In addition, we suppose that we have the decision matrix $ProvMat[i][j]$ containing IaaS services offered by the IaaS provider. In the decision matrix $UserMat$, $UserMat[i][j]$ represent the IaaS service i under the criterion j.

$$UserMat = \begin{bmatrix} u_{00} & \dots & u_{0n} \\ \vdots & \ddots & \vdots \\ u_{n0} & \dots & u_{nm} \end{bmatrix}$$

Our personalized SAW algorithm gives as output, the index i representing the adequate cloud service i in the decision matrix.

Algorithm 1 Personalized SAW Algorithm

Require: $Weight[i] \neq 0$
  $Min = 0$
  for $int\ i\ from\ 0\ to\ n$ **do**
    for $int\ j\ from\ 0\ to\ n$ **do**
      $Sub[i][j] = abs(ProvMat[i][j] - UserMat[i][j])$
    end for
  end for
  for $int\ j\ from\ 0\ to\ m$ **do**
    $DualWeight[j] = 1 - Weight[j]$
  end for
  for $int\ i\ from\ 0\ to\ n$ **do**
    $Score[i] = 0$
    for $int\ j\ from\ 0\ to\ m$ **do**
      $Score[i] = Score[i] + Sub[i][j] * DualWeight[j]$
    end for
  end for
  for $int\ i\ from\ 0\ to\ n$ **do**
    if $Score[i] < Min$ **then**
      $Min \leftarrow Score[i]$
      $Index \leftarrow i$
    end if
  end for
  return i

IV. IaaSSelAss: a Framework for IaaS Selection Assistant

In order to implement our proposed approach, we develop the framework IaaSSelAss. We have used Eclipse Modeling Framework, Java Platform Enterprise Edition (JEE) and Mahout eclipse framework [19]. IaaSSelAss guides cloud users step by step in the selection process and proposes IaaS providers and services with adequacy percentage according to applications and users profile. IaaSSelAss has been designed to support different IaaS providers such as Amazon, Google and Azure Microsoft. We demonstrate the effectiveness of our framework through an evaluation simulation.

The idea of merging RS and MCDM techniques in a structured approach based on two well defined steps as explained in Section IV, provides satisfactory results. In this section, we conduct simulations on 25 real users some of them are PhD students. These simulations show that our approach prove to be efficient rather than using RS and MCDM techniques each independently. We define the simulations' conditions as follows:

- Supported IaaS provider: Amazon, Google, Microsoft Azure
- Number of users: 25
- Number of items (IaaS services): 30
- Active user's profile: User profile 2 defined in Table IV
- Active user's application profile: Active user application profile 2 defined in Table IV
- The non-functional requirements are defined as follows:
  - Pricing model: Per hour
  - Resource Location: US

We define in Table V the decision matrix " $ProvMat[][]$ " used by the personalized SAW algorithm of our approach. Table V contains 5 configuration models of Virtual Machines instances provided by Amazon [20]. Each value in Table V is verified and identified from cloud provider's official web site. Although the number of users and items is relatively small compared to commercial RS, it proves to be sufficient for the purpose of these simulations.

To compare our framework IaaSSelAss to RS techniques (CF algorithm), we omit the step two of our approach, which is the use of personalized SAW algorithm and we rely only on the Collaborative Filtering algorithm in order to create a simple recommender System. The metrics used to evaluate our approach with the use of Rs are the Root-Mean Square Error (RMSE) and The Normalized Discounted Cumulative Gain (NDCG).

The RMSE is a metric widely used to evaluate predicted ratings [21]. It represents the sample standard deviation of the differences between predicted values and expected values. RMSE is the square root of the average of squared errors.

$$RMSE = \frac{\sqrt{\sum_{i=1}^{n}(p_{A,i} - \widehat{p}_{A,i})^2}}{N}$$

where $p_(A,i)$ is a predicted value by user "A" for item i, $\widehat{p}_{A,i}$ is the expected value of user "A" for item i, and N is the number of predicted values. In order to be able to calculate RMSE values, we assume that users introduce their expected rating values.

The Normalized Discounted Cumulative Gain (NDCG) is a measure of ranking quality. NDCG is defined as

$$NDCG_N = \frac{DCG_N}{IDCG_N}$$

where DCG_N and $IDCG_N$ are the Discounted Cumulative Gain (DCG) of top-N items of a predicted ranking and the ideal ranking, respectively. DCG_N is calculated by

$$DCG_N = \sum_{i=1}^{N} \frac{2^{(rel_i)} - 1}{log_2(i+1)}$$

where rel_i is the value of the item at position i of a ranking and $IDCG_N$ is calculated by

$$IDCG_N = \sum_{i=1}^{REL} \frac{2^{(rel_i)} - 1}{log_2(i+1)}$$

where REL represents the list of relevant items (ratings $\geq 0,5$). The value of NDCG is between 0 and 1, where a larger value means a better ranking, and 1 implies the ideal ranking.

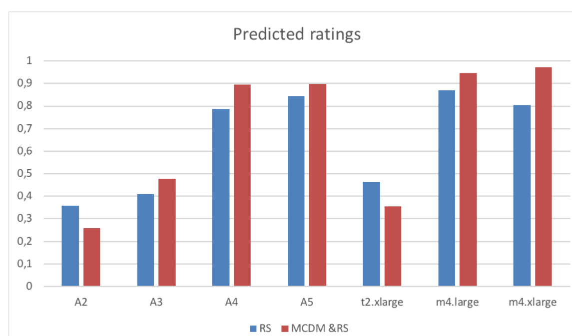

Figure 2. Predicted Ratings

As illustrated in Figure 2 for user 2, the predicted ratings are 0,8403, 0,8053, 0,7872, respectively for AWS instances m4.large, m4.xlarge presented in Table III, and Azure instance A4 presented in Table V. For clarity and visibility purposes, we did not display all instances' predicted ratings of Tables III and V. The scores given by the personalized SAW algorithm of our approach for the same instances are respectively 0,9476, 0,9734, 0,8954. When conducting the CF approach, we obtained 0,042 and 0,571 as RMSE and NDCG average, respectively. However, the RS & MCDM approach gave us 0,031 and 0,73 as RMSE and NDCG

TABLE IV. USERS AND APPLICATIONS PROFILES

User	User profile			Application profile									
	Favorite provider	Expertise level	Previous experiences	Functional requirements								QoS	
				Compute			Storage		Network			Response time ms	Availability
				vCPU	Clock speed GHz	CPU events/s	RAM GB	Hard drive's size GB	Bandwidth $\mathrm{Gbit\,s^{-1}}$	Throughput $\mathrm{Mbit\,s^{-1}}$	Latency ms		
Weights User1	-	-	-	0,5	0,3	0,1	0,4	0,6	0,6	0,1	0,3	0,5	0,3
Values User1	-	Beginner	-	2	2	$v \geq 50$	$v \geq 2$	> 50	1	$10 < v \leq 50$	-	$100 < v \leq 900$	95%
Weights User2	-	-	-	0,4	0,2	0,4	0,6	0,4	0,5	0,3	0,2	0,7	0,2
Values User2	Amazon	Intermediate	2	4	3	47	8	80	2	35	70	700	99%

TABLE V. AMAZON DECISION MATRIX [20]

Model	vCPU	Clock speed GHz	CPU events/s	RAM GB	Hard drive GB	Bandwidth $\mathrm{Gbit\,s^{-1}}$	Throughput $\mathrm{Mbit\,s^{-1}}$	Latency ms	Response time ms	Availability
t2.nano	1	3,3	37	0,5	8	-	-	200	1600	99%
t2.medium	2	3,3	42	4	≥ 30	-	-	200	1600	99%
t2.xlarge	4	3,3	42	6	≥ 100	0,7	45	120	1000	99%
m4.large	4	2,4	57	8	≥ 500	1	62	100	700	99%
m4.xlarge	8	2,4	57	16	-	2	125	100	700	99%

average (Figure 3). So, in terms of RMSE (i.e., 0,042 vs.0,031), the merging of MCDM & RS performs better than RS only. In terms of NDCG (i.e., 0,571 vs. 0,73), RS & MCDM present better result than the CF approach.

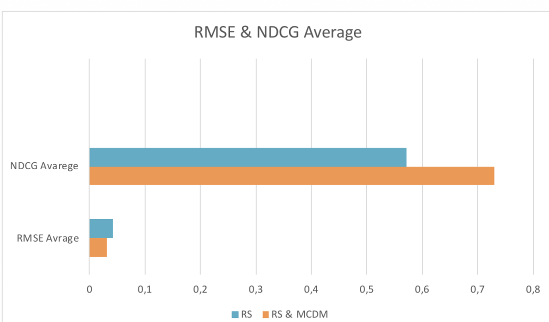

Figure 3. RMSE & NDCG Average

It is worth pointing that the use of CF algorithm only obliges us to calculate predicted ratings for all items in our knowledge base which can be time consuming. However, by applying the step one of our approach we can reduce the number of candidate services by providing only services related to the selected IaaS provider. In addition, the selection of IaaS services using CF algorithm will be associated with previous users experiences in our knowledge base. Although, we identify the most similar users their application profiles must be more or less different to the active user application profile. Consequently, the predicted IaaS services are less accurate. In conclusion, this simulation shows that our approach performs better than using RS only.

V. CONCLUSION

This paper investigates the challenges of selecting appropriate cloud infrastructure and services. We proposed a new hybrid approach that transforms the IaaS services selection from an ad-hoc task that involves manually reading the provider documentations to a structured and guided process. Our solution aims to involve users in the selection process and takes into consideration their personal preferences and their previous experiences in addition to the functional requirements of their applications. Thus, our approach proposes relevant IaaS services responding users expectations. Although we believe that the framework IaaSSelAss supporting our approach leaves scope for a range of enhancements, yet it provides suitable results. For our ongoing works, we are focusing on reducing the complexity of introducing the application's profile, such as CPU clock speed, throughput, etc. In fact, we aim to deduce these requirements from real-world scenarios and experiences, such as the capacity of a server to respond to a given number of users per hour with a required latency between request and response. In addition, we are working on integrating other cloud service models like Platform as a Service(PaaS) or Software as a Service (SaaS). In our approach, we select IaaS services according to a single selected provider, thus, we can extend it to support a Multi-cloud services selection.

REFERENCES

[1] P. Mel and T. Grance, "The nist definition of cloud computing," *NIST Special Publication*, pp. 800–145, 2011, Available at: http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf [accessed: 2018-05-23].

[2] M. Eisa, M. Younas, K. Basu, and H. Zhu, "Trends and directions in cloud service selection," *IEEE Symposium on Service-Oriented System Engineering*, 2016, ISBN: 978-1-5090-2253-3.

[3] A. J. Ruby, B. W. Aisha, and C. P. Subash, "Comparison of multi criteria decision making algorithms for ranking cloud renderfarm services," *Indian Journal of Science and Technology*, vol. 9, p. 31, 2016.

[4] S. Soltani, K. Elgazzar, and P. Martin, "Quaram service recommender: a platform for iaas service selection," *International Conference on Utility and Cloud Computing*, pp. 422–425, 2016, ISBN: 978-1-4503-4616-0.

[5] J. Lu, D. Wu, and G. Zhang, "Recommender system application developments: A survey," *Decision Support Systems*, vol. 74, pp. 12–32, 2015.

[6] M. Zhang, R. Ranjan, S. Nepal, M. Menzel, and A. Haller, "A declarative recommender system for cloud infrastructure services selection," *GECON*, vol. 7714, pp. 102–113, 2012.

[7] T. Zain, M. Aslam, M. Imran, and Martinez-Enriquez, "Cloud service recommender system using clustering," *Electrical Engineering, Computing Science and Automatic Control (CCE)*, vol. 47, pp. 777–780, 2014.

[8] L. Sun, H. Dong, F. Khadeer, Hussain, O. K. Hussain, and E. Chang, "Cloud service selection: State-of-the-art and future research directions," *Journal of Network and Computer Applications*, vol. 45, pp. 134–150, 2014.

[9] Z. Rehman, F. Hussain, and O. Hussain, "Towards multi-criteria cloud service selection," *International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, 2012, ISBN: 978-1-61284-733-7.

[10] C. JatothG and U. Fiore, "Evaluating the efficiency of cloud services using modified data envelopment analysis and modified super-efficiency data envelopment analysis," *Soft Computing, Springer-Verlag Berlin Heidelberg*, vol. 7221-7234, p. 21, 2017.

[11] F. Aqlan, A. Ahmed, O. Ashour, A. Shamsan, and M. M. Hamasha, "An approach for rush order acceptance decisions using simulation and multi-attribute utility theory," *European Journal of Industrial Engineering,*, 2017, ISSN: 1751-5254.

[12] M. Whaiduzzaman, A. Gani, N. B. Anuar, M. Shiraz, M. N. Haque, and I. T. Haque, "Cloud service selection using multicriteria decision analysis," *The Scientific World Journal*, vol. 2014, p. 10, 2014.

[13] L. D. Ngan and R. Kanagasabai, "Owl-s based semantic cloud service broker," *International conference on web services (ICWS)*, pp. 560–567, 2013, ISBN: 978-1-4673-2131-0.

[14] F. K. Hussain, Z. ur Rehman, and O. K. Hussain, "Multi-criteria iaas service selection based on qos history," *International Conference on Advanced Information Networking and Applications*, 2014, ISSN: 1550-445X.

[15] Z. Li, L. OBrien, H. Zhang, and R. Cai, "On the conceptualization of performance evaluation of iaas services," *IEEE Transactions on Services Computing*, vol. 7, pp. 628 – 641, 2014.

[16] Q. Yu, "Cloudrec: a framework for personalized service recommendation in the cloud," *Knowledge and Information Systems*, vol. 43, pp. 417–443, 2014.

[17] C. Qu and R. Buyya, "Oa cloud trust evaluation system using hierarchical fuzzy inference system for service selection," *International Conference on Advanced Information Networking and Applications*, 2014, ISBN: 978-1-4799-3630-4.

[18] "Microsoft Azure," 2018, URL: https://azure.microsoft.com/en-us/pricing/details/cloud-services [accessed: 2018-05-23].

[19] "Mahout Apache," 2018, URL: http://mahout.apache.org/ [accessed: 2018-05-23].

[20] "Amazon Instance Types," 2018, URL: https://aws.amazon.com/ec2/instance-types/?nc1=h_ls [accessed: 2018-05-23].

[21] Z. Zheng, X. Wu, Y. Zhang, M. R.lyu, and J. Wang, "Qos ranking prediction for cloud services," *European Journal of Industrial Engineering,*, vol. 24, pp. 1213–1222, 2013.