Computer Science and Information Systems 19(2):979–1005 https://doi.org/10.2298/CSIS220215011O

A Novel Hybrid Recommender System Approach for Student Academic Advising Named COHRS, Supported by Case-based Reasoning and Ontology

Charbel Obeid[1], Christine Lahoud[2], Hicham El Khoury[3], and Pierre-Antoine Champin[4]

[1] LIRIS, Caude Bernard University Lyon 1
France
cobeid@liris.cnrs.fr
[2] French University in Egypt
Egypt
Christine.lahoud@ufe.edu.eg
[3] LaRRIS, Lebanese University
Lebanon
hkhoury@ul.edu.lb
[4] LIRIS, Caude Bernard University Lyon 1
France
Pierre-antoine.champin@univ-lyon1.fr

Abstract. The recent development of the World Wide Web, information, and communications technology have transformed the world and moved us into the data era resulting in an overload of data analysis. Students at high school use, most of the time, the internet as a tool to search for universities/colleges, university's majors, and career paths that match their interests. However, selecting higher education choices such as a university major is a massive decision for students leading them, to surf the internet for long periods in search of needed information. Therefore, the purpose of this study is to assist high school students through a hybrid recommender system (RS) that provides personalized recommendations related to their interests. To reach this purpose we proposed a novel hybrid RS approach named (COHRS) that incorporates the Knowledge base (KB) and Collaborative Filtering (CF) recommender techniques. This hybrid RS approach is supported by the Case-based Reasoning (CBR) system and Ontology. Hundreds of queries were processed by our hybrid RS approach. The experiments show the high accuracy of COHRS based on two criteria namely the "accuracy of retrieving the most similar cases" and the "accuracy of generating personalized recommendations". The evaluation results show the percentage of accuracy of COHRS based on many experiments as follows: 98 percent accuracy for "retrieving the most similar cases" and 95 percent accuracy for "generating personalized recommendations".

Keywords: Knowledge base, Collaborative Filtering, Hybrid Recommender System, Case-based Reasoning, Ontology.

1. Introduction

Academic advising at most high schools is limited in its ability to help students in identifying appropriate educational pathways. For example, choosing a university and a universtiy major is a challenging task rife with anxiety that gets students confused. (Cuseo, J., 2003) (V.N., 2007) revealed that 20 to 50 percent of students in the United States start their university journey with an undecided major and 50 to 75 percent of learners in higher education have changed their major at least once before graduation. This suggests that students' career choices are unclear upon university admission and enrollment. Besides (Tett et al., 2017), (Siri et al., 2016) outlined the transition from high school to university by socio-cultural perspectives affected by personal factors and the learning environment, comprising students' previous experiences.

Therefore, students at high school need assistance to match their interests with the available universities and field of study programs. Moreover, students need to filter, prioritize, and efficiently get adequate information to overcome the web information overload issues.

The purpose of this study is to propose a novel hybrid system approach that guide high school students toward higher education and career choices that match their interests and preferences.

Recommender systems (RSs) are software that provide recommendations to active users [27]. These systems address information overload and help users to take decisions suitable to their interests. For example, when using a RS, users will have the option to select which product to buy, which movie to watch, or which article to read. Such software is commonly used when the volume of data outperforms the user's capability to analyze it.

Existing RSs can be essential tools in guiding high school students when searching for appropriate universities/colleges, and university majors that align with their aspiring careers. However, these systems have many limitations such as Cold-start, Sparsity, Graysheep, and Scalability problems [5]. For example, RSs that are based on the CF technique use users' ratings instead of supplementary knowledge of the products and users to generate recommendations [17]. Besides, in e-learning RSs, limitations occur when recommending specific choices of educational materials. One of the biggest limitations is that e-learning RSs have a weak capability to generate personalized recommendations. These limitations happen because of the variety in the studying style and education level of the learners [26].

Our contributions in this study are summarized as follows: Firstly, guiding high school students in making the right decision when selecting a university/college, university major, and career path. Secondly, proposing a novel KB hybrid RS approach supported by the CBR and ontology techniques. This hybridization strategy helps to generate recommendations based on users' knowledge and ratings. In our system, the CBR is integrated into the KB recommender engine to support the recommendation process and generate recommendations based on the prior university graduates' knowledge. Also, the ontology is integrated into the KB recommender system to represent the schools, higher education domain, career domain, and students' profiles models. Thirdly, enabling RSs to process high dimensional datasets that encompasses heterogeneous data types. Finally, overcoming and reducing the limitations of traditional RSs.

This study is organized as follows: an overview of the recommendation techniques, similarity metrics, neighborhood-based CF algorithms, evaluation metrics, and related works are discussed in section 2. The proposed hybrid RS approach and the study experiments are presented in section 3. Finally, the conclusion and future work are discussed in section 4.

2. Background

In the following section, an overview of the recommendation techniques, similarity metrics, neighborhood-based CF algorithms, evaluation metrics and related works are presented.

2.1. Basic Recommendation Techniques

Many recommendation techniques were implemented to provide recommendations to users. Techniques such as Demographic-based [34], Knowledge-based [6] (Constraint-based [10], Case-based reasoning [15], Ontology-based [41]), Content-based filtering [37], Collaborative Filtering [20] (Memory-based [39,3], Model-based [31]), and hybrid RSs [8,16] are widely used in various domain. The following figure illustrates the taxonomy of the most popular RSs' techniques.

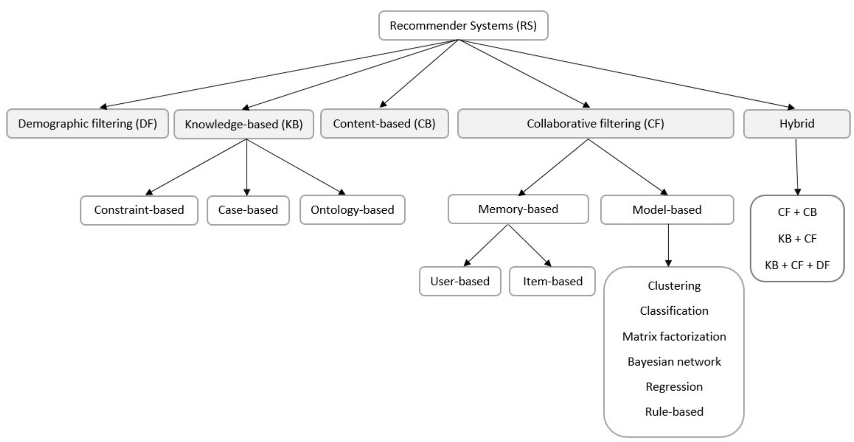


Fig. 1. Recommender systems' taxonomy

The CF recommender system is the most popular and successful approach implemented in the recommendation domain. This technique is based on the notion that if some users have the same preferences in their history, they will share mutual preferences in the future [12]. The CF technique integrates users' preferences, interests, and actions to suggest products to users based on the match between users' profiles. The CF algorithm encompasses two essential types namely the Model-based and Memory-based techniques. The memory-based computes the similarity between users based on users' activities, ratings, or selected items to generate appropriate recommendations. Memory-based integrates users and items' dataset to generate predictions. Besides, model-based calculates the similarities between users and/or items, then saves them as a model, and then implements the saved similarity values to generate recommendations. The Model-based implements several algorithms such as clustering algorithm, matrix factorizations, Bayesian network or regressions

Nevertheless, the CF technique has many limitations and problems such as the Cold-start for new users, Scalability, Sparsity, Grey-sheep problems [8]. Additionally, CF technique faces many limitations such as treating heterogeneous data types and high dimensional datasets.

- Cold-start: the CF technique needs past users' history such as users' activities and ratings to generate precise recommendations. Cold-start problems occur when the dataset does not include sufficient ratings and preferences.
- Scalability: an enormous community of users and products exists in several of the environments that the CF systems make a recommendation in. Hence, great computation power is necessary to compute recommendations.
- Sparsity: this problem occurs when the items and users' matrix table is widely sparse. In this case, the precision of the recommendations will decrease since past users could not rate all the available products in the system.
- Grey-sheep problem: This problem is caused by odd recommendations since the user may have other features that do not match with any other user or community of users [7]. An example of a grey-sheep issue is when a user neither agrees nor disagrees with any user or group of users.
- Treating heterogeneous data types limitation: basic recommender filtering techniques have no capability to treat heterogeneous data types. Here comes the role of the hybrid recommender systems that can handle and compute heterogeneous data.
- Treating high dimensional datasets limitation: basic recommender filtering techniques have no capability to deal with high dimensional datasets. High dimensional datasets encompasses high number of attributes. This limitation can be addressed by decreasing the number of attributes in the dataset or using a Hybrid RS that can handle large datasets.

The CB technique works with the data provided by the user. Users' data is collected either explicitly by rating or implicitly by clicking a hyperlink. The CB algorithm function is to find products with the same content to suggest to the active users. The CB recommendations are based on what the active user liked. This recommender system compares the user's items ratings with items he or she did not rate and then computes the similarities. Based on that, the recommender system recommends the appropriate items, which are similar to the rated ones [32].

The KB recommender system generates appropriate recommendations based on explicit and implicit knowledge about the users and items. This technique integrates knowledge such as users' characteristics, preferences, interests, or needs [6]. KB recommender systems deal with the cases in which ratings are not used for the recommendations. Therefore, ontology is essential in the KB system to overcome the cold-start issues. Ontology is a KB technique, which does not take into consideration users and items past information. KB techniques are good examples to hybrid them with different recommendation techniques such as CF and CB.

DF recommender system purpose is to cluster the users based on their personal features in order to suggest appropriate recommendations. The DF recommendations are based on users' demographic data such as age, gender, education, occupation, address (city, country), etc. [18,19]. The importance of such system is that it overcome the new user issue of the CF technique since they do not require user ratings. Also, it is easy to preprocess the data since it does not require domain knowledge. In DF, it is easy to identify similar users since new user must register and enter his/her demographic data to the system. In DF, users having the same demographic characteristics may also have similar preferences or tastes.

Combining two or more recommender techniques for a specific domain is an approach named "Hybrid RS". The hybridization method is commonly used for improving recommendation accuracy and overcoming the traditional RSs' problems and limitations.

Burke [8] classified the hybrid RS techniques into seven hybridization strategies namely (weighted, switching, mixed, feature combination, feature augmentation, cascade, and meta-level). In Weighted, the values of two or more RS are collected to generate a single recommendation. In Switching, the system navigates between the hybrid recommendations systems taking into account the running case. In Mixed, the output of two or more recommendation techniques are generated simultaneously. For example, CF rank (3) + CB rank (2) Combined rank (5). In Feature combination, the features from different sources are integrated into a single RS technique. In Cascade, the running recommendation technique refines the output of a second recommender system. In Feature augmentation, the output of one recommendation technique is integrated as input attributes into a second recommender system. In Meta-level, he model learned by one recommender is integrated as input into another recommender system.

In our work, we implemented the Feature augmentation strategy that combines the CF and KB techniques in a uniform system.

2.2. The Similarity Metrics, Neighborhood-based CF Algorithms, and Evaluation Metrics

Usually, people count on recommendations given by other people that are linked to different domains or products. Thus, RSs offer users the capability to count on the preferences or interests of large communities. In order to generate personalized recommendations, a RS makes some similarity evaluations on the users' preferences or interests and chooses which recommendations match users' tastes. So, what is the similarity between two items? In all situations, a full similarity is an absence of differences. Therefore, similarity metrics in a RS are about matching products or users that are most similar.

In this study, we aimed to implement and evaluate many RS algorithms and similarity metrics in order to generate better and accurate recommendations.

The Similarity Metrics: RS uses similarity metrics that are implemented in machine learning [14]. Most similarity metrics are correlated with vector space approaches. The applied Mahout Java library [1] has many similarity algorithms, which are,

- Euclidean Distance Similarity
- Pearson Correlation Coefficient Similarity
- Spearman Correlation Coefficient Similarity
- City Block Similarity
- Uncentered Cosine Similarity

The Euclidean Distance is the most common among all the distance measures. This distance is a straight-line distance between two vectors. The EuclideanDistanceSimilarity technique in mahout [1] java library calculates the similarity between two users X and Y. This technique considers items as dimensions and preferences as points along those dimensions. The distance is calculated using all items where both users have a similar preference for that item. It the square root of the sum of the squares of differences in position along each dimension. The similarity could be computed as 1 / (1 + distance) and the distance is mapped between (0, 1]. The distance between two points with coordinates (x, y) and (a, b) is given by

$$dist((x,y),(a,b)) = \sqrt{(x-a)^2 + (y-b)^2} \tag{1}$$

In Euclidean distance, the value of the distance is smaller when users are more similar. The larger the distance value is, the smaller the distance is. Thus, the closer the distance, the greater the similarity [28].

The PearsonCorrelationSimilarity is based on the Pearson correlation. The values for users A and B are calculated as follows:

- SumA2: the sum of the square of all A's preference values.
- SumB2: the sum of the square of all B's preference values.
- sumAB: the sum of the product of A and B's preference value for all items for which both A and B express a preference.

To calculate the correlation the following formula is used: sumAB / sqrt(sumA2 * sumB2).

$$sim(a,b) = \frac{\sum_p \in P(r_{a,p} - \overline{r_a})(r_{b,p} - \overline{r_b})}{\sqrt{\sum_p \in P(r_{a,p} - \overline{r_a})^2}\sqrt{\sum_p \in P(r_{b,p} - \overline{r_b})^2}} \tag{2}$$

a and b represents two users or items, p represents an item, $r_{a,p}$ and $r_{b,p}$ represent the user ratings from a and b for p, and average ratings of r_a and r_b are, for the item or user a and b [28]. Here the Pearson correlation coefficient is equal to the covariance of the two variables divided by the standard deviation of the two variables. The results range between [-1, 1], the larger the absolute value, the stronger the correlation, and the negative correlation has little significance for the recommendation.

The SpearmanCorrelationSimilarity is like the PearsonCorrelationSimilarity. However, the SpearmanCorrelation compares the relative ranking of preference values instead of preference values themselves. Each user's preferences are sorted and then assigned a rank as their preference value, with 1 being assigned to the least preferred item. The equation for Spearman Correlation Similarity is given in equation (3):

$$w(a,b) = \frac{\sum_{i=1}^{n}(rank_{a,i} - \overline{rank_a})(rank_{b,i} - \overline{rank_b})}{\sigma a * \sigma b} \tag{3}$$

The calculation in Spearman Correlation Similarity is very slow and there is a lot of sorting. Its results range between [-1.0, 1.0], 1.0 when there is a total match, -1.0 when there is no match.

The City block distance [13] also referred to as Manhattan distance. It calculates the distance between two points, a and b, with k dimensions. The City block distance is computed like following:

$$\sum_{i=1}^{n} |a_i - b_i| \tag{4}$$

The City block distance result should be greater than or equal to 0. The result for identical points should be equal to 0 and greater than 0 for the points that express little similarity.

The UncenteredCosineSimilarity is an implementation of cosine similarity. Its result is the cosine of the angle formed between two vectors. The correlation between two points, a and b, with k dimensions is computed as:

$$Similarity = \frac{\sum_{i=1}^{n} a_i * b_i}{\sqrt{\sum_{i=1}^{n} a_i^2} * \sqrt{\sum_{i=1}^{n} b_i^2}} \tag{5}$$

This correlation ranges from (+1 to -1). The highest correlation is equal to +1 and the dissimilar points have a correlation equal to -1.

The Neighborhood-based CF Algorithms: The two types of neighborhood-based CF algorithms are the User-based CF and Item-based CF. The difference between the User-based CF and the Item-based CF is that User-based takes the rows of ratings matrix and Item-based takes the columns of ratings matrix for similarity measurement. In User-based, the item's recommendation rating for a user is calculated depending on those items' ratings by other similar users. The ratings are predicted using the ratings of neighboring users. In User-based, the Neighborhoods are defined by similarities among users. In Item-based, the item's rating is predicted based on how similar items have been rated by that user. The ratings are predicted using the user's own ratings on neighboring items. In Item-based, the Neighborhoods are defined by similarities among items.

The Evaluation Metrics: Many researchers found several evaluation metrics to evaluate the quality of the prediction. Prediction accuracy metrics find values that show how much the prediction is close to the real preference. The evaluation metrics help to assess the precision of the RS recommendations by comparing the predicted ratings with the rating of the active user. There are many prediction accuracy metrics used for testing the prediction accuracy of the used algorithms such as the Mean Absolute Error (MAE) [2] and Root Mean Squared Error (RMSE) [2]. In our graduates' dataset context, MAE and RMSE will assess how well the RS can predict a user's rating for a course/career.

The MAE metric evaluates the accuracy of an algorithm by comparing the value of predictions against the actual user's ratings for the user-item pairs in the test dataset. For each rating prediction pair, their absolute error is calculated. After summing up these pairs and dividing them by the total number of rating-prediction pairs, Mean Absolute Error can be found. It is the most commonly used and can be interpreted easily. The equation of Mean Absolute Error is given in equation (6).

$$MAE = \frac{\sum_{i}^{n} |r_i - e_i|}{n} \tag{6}$$

The RMSE is calculated by finding the square root of the average squared deviations of a user's estimated rating and actual rating. Once rating-prediction difference is calculated, the power of 2 is taken. After summing them up and dividing them by the total number f rating-prediction pairs and taking square root of it, Root Mean Square Error can be found. The equation of Root Mean Square Error is given in equation (7).

$$RMSE = \sqrt{\frac{\sum_i^n (r_i - e_i)^2}{n}} \tag{7}$$

Where in both formulas for MAE and RMSE n is the total number of items, i is the current item, ri is the actual rating a user expressed for i, and ei is the RS's estimated rating a user has for i. The smaller RMSE and MAE are, the more accurate a RS. This is because RMSE and MAE will calculate smaller values if the deviations between actual and predicted ratings are smaller. By using evaluation metrics, prediction accuracy and efficiency of the CF methods can be calculated and compared.

The Evaluation Algorithms: To evaluate the accuracy of our recommender system, two evaluation algorithms were implemented namely the HoldOutEvaluator and SameSplitEvaluator.

- The HoldOutEvaluator method splits the case-base into two sets, one used for testing where each case is used as a query, and another that acts as a normal case-base. This process is performed several times.
- The SameSplitEvaluator method splits the case-base into two sets, one used for testing where each case is used as a query, and another that acts as a normal case-base. This method is different from the other evaluator because the split is stored in a file that can be used in following evaluations. This way, the same set is used as queries for each evaluation. The generateSplit() method does the initial random split and saves the query set in a file. Later, the HoldOutfromFile() method uses that file to load the queries set and perform the evaluation.

These two evaluation algorithms helped us to evaluate the accuracy of our proposed hybrid recommender system that is supported by the ontology and CBR system. In the following section, the related works are presented.

2.3. Related Work

Researchers have proposed several approaches for building RSs, which offer recommendations to users based on specific criteria that match their interests. For instance, Jihane Karim et al. [23] proposed a hybrid method for a generic and personalized CBR-based RS. The authors used a generic ontology to represent the essential knowledge needed throughout the reasoning task. Moreover, they proposed a hybridization technique that incorporates CBR and CF to enhance recommendations. The preliminary experiments for this system were performed using restaurant datasets. Additionally, several hybrid strategies with CF and DF techniques have been proposed in many studies [22,38,18]. This type of hybridization can minimize the limitations of the CF technique. Eventually, few hybridization approaches have been proposed with a combination of three filtering techniques such as CF, KB, and DF. For instance, the authors in this paper [21] proposed a hybridization strategy consisting of three core techniques namely the DF, semantic KB, and CF. The goal of this RS is to enhance the visitor's experience in visiting museums and tourist places. The demographic approach is first used to overcome the CF cold-start issue and the semantic approach is then activated to provide recommendations to the user semantically close to those he/she has previously appreciated. Finally, the collaborative approach is used to recommend to active user works previously liked by similar users. In addition, several hybridization models have been implemented with the combination of the CF and KB to improve the accuracy of recommendations. The possibility of combining CF and KB techniques is introduced in [9]. This hybridization approach has the ability to overcome CF limitations such as the problem of new users or items. As an example, Chavarriaga et al. [11] proposed a CF and KB technique to suggest learning resources or activities. This approach helps learners to reach a high competency ranking by using an online course platform. Moreover, Jhon K. Tarus et al. [25] proposed a KB hybrid RS supported by sequential pattern mining (SPM) and ontology. This hybrid system provides recommendations of e-learning resources to learners. In this system, the ontology is integrated to represent the domain knowledge about the learner and learning resources. The role of the SPM algorithm is to determine the learners' sequential learning patterns. As well, Mohammad I. [33] designed and implemented a hybrid RS named OPCR that incorporates the CB and CF filtering supported by an ontology to overcome the user Cold-start problem. This system incorporates all information about courses and helping students to choose courses towards their career goals. Besides, Hsu Mei-Hua [30] presented an online-personalized English learning RS. This system is capable of suggesting students with reading lessons that fit their interests. This hybrid RS incorporates the CB, CF, and data mining techniques to study students' reading data and computes recommender scores. Besides, Rodriguez et al. [36] presented a student-centered Learning Object (LO) RS based on a hybridization approach that incorporates the CB, CF, and KB techniques. The LOs that are adapted to the learner model/profile are retrieved from the LO databases by implementing the saved descriptive metadata of the objects. Also, Tarus et al. [26] proposed an approach that combines the CF and KB supported by ontology to suggest personalized learning materials to online learners. In this system, the ontology is used to represent the learner characteristics while CF predicts ratings and provide recommendations. Furthermore, researchers have proposed several advanced approaches for building RSs based on cognitive models, sentiment, and affective analysis. For instance, this study [40] demonstrates that the analysis models of human cognition grips promise for the design of recommender mechanisms. Besides, Ha et al. [19] designed a multi-level sentiment network visualization mechanism based on emotional words in the movie domain. The proposed approach has been integrated into a RS to recommend movies with similar emotions to the watched ones. As well, the authors in this paper [24] studied sentiment and affective analysis in a RS of blogs. The blog's recommendations are based on the association between the sentiment and affective analysis of the blog and text content submitted by the users. In addition, this paper [29] presented an educational RS based on affective computing. The main aim of this RS is to discover educational resources based on emotion detection. Our general study of the RS techniques can be summed up by the advantages and drawbacks of the hybridization approaches shown in the following table.

Hybridization model	**Advantages**	**Drawbacks**
KB + Memory-based CF	- Reduce the cold-start problem of CF - Reduce the sparsity issue of CF - The accuracy of the recommendations of this hybridization outperforms the memory-based CF predictions. - Fast replying when user's preferences are modified.	- It is not scalable for large datasets. - It needs knowledge engineering.
KB + Model-based CF	- The accuracy of the recommendations of this hybridization outperforms the model-based CF predictions. - Fast replying when user's preferences are modified. - It has a scalability feature.	- The hybridization of the memory-based CF with the KB provided better results than this system. - It needs knowledge engineering.
DF + Memory-based CF	- Reduce the cold-start problem. - The accuracy of the recommendations of this hybridization outperforms the memory-based CF predictions.	- It is hard to acquire demographic data. - It is not scalable for large datasets.
DF + Model-based CF	- The accuracy of the recommendations of this hybridization outperforms the model-based CF predictions. - It has a scalability feature.	- The hybridization of the memory-based CF with the DF provided better results than this system. - It is hard to acquire demographic data.

Table 1. Comparison of different hybridization techniques

In summary, our approach differs from the previously mentioned approaches in the sense that it integrates the user-based CF and KB techniques that are supported by the CBR and ontology. This approach is named CBR and ontology-based hybrid recommender system (COHRS). The CBR and ontology knowledge are integrated into this hybrid system to overcome the issues and limitations of traditional RS. We integrated the ontology engineering to model the knowledge acquired from different resources such students' demographic data, interests, schools, universities/colleges, university majors, and career domain.

By incorporating the CBR and ontology into COHRS the following CF issues and limitations have been addressed:

– Grey-sheep problem: this problem is caused by odd recommendations since the user may have other features that do not match with any other user or community of users

[7]. An example of a grey-sheep issue is when a user neither agrees nor disagrees with any user or group of users.

– Treating heterogeneous data types limitation: basic recommender filtering techniques have no capability to treat heterogeneous data types. Here comes the role of the hybrid recommender systems that can handle and compute heterogeneous data.
– Treating high dimensional datasets limitation: basic recommender filtering techniques have no capability to deal with high dimensional datasets. High dimensional datasets encompasses high number of attributes. This limitation can be addressed by decreasing the number of attributes in the dataset or using a Hybrid RS that can handle large datasets.

Moreover, our proposed system is specialized in the field of guiding high school students toward higher education choices. No study has been conducted previously that describes the higher education domain with a hybridization strategy that combines the KB, CF, ontology, and CBR techniques. The following two tables present a comparison between COHRS and other hybrid RSs.

Hybrid RS Name	**Supported By CBR**	**Supported by Ontology**	**Applied to high dimensional dataset**
COHRS	X	X	X
Chavarriaga et al. Hybrid RS [11]			
Mohammad I. [33] OPCR		X	
Jhon K. Tarus et al. Hybrid RS [25]		X	
Hsu Mei-Hua Hybrid RS [30]			
Rodriguez et al. Hybrid RS [36]			
Tarus et al. Hybrid RS [26]		X	
Jihane Karim et al. [23]	X	X	

Table 2. The Core Techniques Implemented by Each Hybrid RS

Table 2 describes the core techniques implemented by each hybrid RS. Whereas, table 3 illustrates the hybridization strategy, key feature, targeted users and targeted domain of the compared hybrid RSs.

Hybrid RS Name	Hybridization Strategy	Used for	Targeted users	Targeted domain
COHRS	KB + User-based CF	Guiding high school student toward higher education choices such as university and field of study.	High school students	Higher education
Chavarriaga et al. Hybrid RS [11]	KB + Item-based CF	Recommending activities and resources that help students in achieving competence levels throughout an online course.	Online learners	E-learning
Mohammad I. [33] OPCR	CB + Item-based CF	Recommending personalized courses that match student's personal needs.	University students	Higher education
Jhon K. Tarus et al. Hybrid RS [25]	KB + Item-based CF	Generating recommendations of e-learning resources to learners.	Online learners	E-learning
Hsu Mei-Hua Hybrid RS [30]	CB + Item-based CF	Recommending students with English reading lessons that fit their interests.	Online learners	E-learning
Rodriguez et al. Hybrid RS [36]	CB + CF + KB	Providing learners with appropriate recommendations adapted to their preferences and bringing LOs closer than expected.	Online learners	E-learning
Tarus et al. Hybrid RS [26]	CF + KB	Suggesting personalized learning materials to online learners.	Online learners	E-learning
Jihane Karim et al. [23]	KB + Item-based CF	Recommending personalized items to customers	Customer	Restaurants Domain

Table 3. The Hybridization Strategy, Key Feature, Targeted Users and Targeted Domain

In the following section, the architecture of the proposed hybrid RS, similarity metrics, neighborhood-based CF algorithms, evaluation metrics, and experimental procedure are presented.

3. The Proposed Approach

The proposed hybrid system incorporates four core techniques (KB, user-based CF, CBR and ontology). This hybrid RS focuses on recommending universities/colleges, university majors, and career choices.

3.1. The Implementation Phases

This hybrid approach involves four main implementation phases: (1) the data acquisition phase, (2) the data-preprocessing phase, (3) the ontology design phase, and (4) the hybrid recommendation phase.

Fig. 2. The Four Main Implementation Phases

The Data Acquisition Phase: Our study focuses on analyzing university graduates trajectories and finding solutions to assist high school students to take appropriate decisions toward higher education choices. However, this study requires special types of data to be used in the analysis phase. Unfortunately, the required data is not available anywhere, since it is related to university graduates educational trajectories. In addition, it is very difficult to acquire it online and users are reluctant to disclose it. Data can be acquired in two ways: implicitly or explicitly. Explicit data are acquired from user ratings; for example, after listening to a song or watching a movie, and the implicit data are acquired form purchase history, search engine searches, or users/items' knowledge. In our case, we worked on gathering the required data through the explicit method. Therefore, we disseminated an online survey that includes more than 50 questions. The survey purpose is to reach university graduates and collect information about their educational trajectories, interests, current career occupation, etc. The dissemination process of the survey covered the Lebanese university graduates.

Our survey was published online in a period of 6 months and it involved questions of heterogeneous data types such as nominal, ordinal, numerical and open-ended. Ordinal data take their values in an ordered finite set. For example, a survey may ask the user to provide feedback on the service he/she received in a restaurant. The quality of service is ranked as (1) Not at all Satisfied, (2) Partly Satisfied, (3) Satisfied, (4) More than Satisfied and (5) Very Satisfied. The larger the set of values, the more informative the data. Nominal data names somewhat without assigning it to an order in relation to other numbered items of data. For example, ”acting”, ”camping” or “cycling” classification for each user’s hobbies. Numerical attributes with continuous values that are represented by numbers and have most of the characteristics of numbers. Open-ended questions are questions that ask an applicant to answer in their natural language. They require a longer response. Thus, open-ended questions provide more information than a simple yes or no answer.

Our survey collected a real-world dataset that includes about 1000 university graduate applications and approximately 20,000 high school course ratings. This real-world dataset has varied data such as demographic data, interests, education and career knowledge, and ratings. In our hybrid system the collected demographic data and domain knowledge are used in the KB system in order to overcome the limitations of traditional RSs while ratings are used in the CF system. A collection of question types was used in this survey such as multiple-choice, Likert scale, and open-ended questions. For example, the answers for “How would you rate your high school grades on the following (Biology, Chemistry, Physics, and Mathematics. . .)?” are Very Good, Good, and Poor/Not concerned. Similarly, answer options for “If you already changed your university major, why did you change it?” are badly advised, Lack of understanding, you were uninterested in courses, and you had new interests...

The survey was organized into six main sections namely the survey description, graduate personal information, graduate high school or vocational school information, graduate first attended university information, graduate interests and career information, and graduate current university major information. The following are some samples of questions copied from the survey sections:

– Graduate personal information section: What is your Gender? (Male or Female); Select the work of your father
– Graduate high school or vocational school information section: What high school did you attend? (High School or Private School); What high school subject did you like best?
– Graduate first attended university information section: What was your university major?; How effective was the teaching within your major at the university? (Very Effective, Somewhat Effective or Not So Effective)
– Graduate interests and career information section: What kind of job/career interests you?; Is your current job related to your university major?
– Graduate current university major information section: What degree are you currently pursuing? (Bachelor Degree, Master Degree, Doctoral Degree or Other); How many times (if any) did you change your major at university (current or before)?

The Data-Preprocessing Phase: Inappropriate and redundant information or unreliable and noisy data exist in all dataset. Thus, analyzing data that has not been carefully refined can generate inaccurate results. Here comes the role of data preprocessing that involves many important steps and techniques. Therefore, data preprocessing is an essential phase in the data mining process and machine learning projects. In our work, we applied the following data preprocessing techniques in order to clean and refine the acquired data from the online survey.

– Data Quality Evaluation: data must be checked for missing, inconsistent and duplicate values.

- Dataset Dimensionality Reduction: significant real-world datasets have a great number of attributes (features). Therefore, the dimensionality reduction technique's purpose is to reduce the number of features in order to make the processing of the data more tractable. Reducing the dimensionality of a dataset is done by defining new features which are an arrangement of the original features.
- Attribute Sampling: sampling is picking a subset of the dataset that we are studying. Analyzing the whole dataset can be too expensive considering the time and memory constraints. Implementing a sampling algorithm can aid in reducing the size of the dataset to a level where the analyst can use a better machine learning algorithm.

In order to initialize the data-preprocessing phase, we extracted the required data as a CSV file from the online survey. The following figure illustrates the process of filling, extracting, cleaning, refining and preparing the desired dataset using many data preprocessing methods and tools such as Weka, WordNet, and Levenshtein distance.

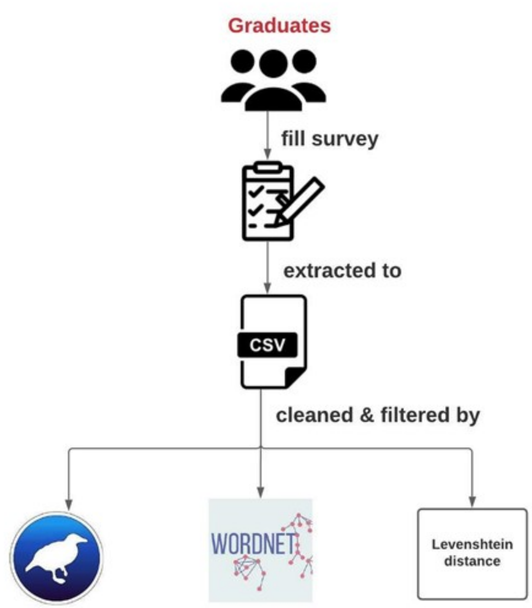


Fig. 3. Data pre-processing

As mentioned in the data acquisition phase, our survey contains more than 50 questions which led us to a large number of features. Therfore, we were obligated to reduce the dimensionality of our dataset. Thus, we used Weka InfoGainAttributeEval technique to perform feature selection by calculating the information gain for each feature for the output variable. The entry values range from 0 (means no information) to 1 (means maximum information). The features that give more information will get a higher information gain value and can be chosen, whereas those that do not show much information will get a lower score and can be ignored in the analysis process.

Additionally, we implemented the WordNet that is a huge lexical database of many languages lexical. It is formed of sets of synonyms or synsets, which are groups of Nouns, Adjectives, Verbs, or Adverbs. The synonyms are linked based on lexical relationships, such as hyponym, hypernym, antonym, etc. This lexical database is available online for free download and usage. WordNet's structure enables this tool to deal with numerous tasks in NLP and computational linguistics such as information and document retrieval, improve search engine returns, automated document and text classification, Word sense disambiguation, machine translation, online lexical dictionary, etc. WordNet is usable from the R language to compute linguistic and text mining processing. As explained before, our survey contains open-ended questions. Thus, the graduates answered these questions in natural language and they expressed the same information differently. For instance, the term bike could be expressed as bike, bicycle, motorcycle, wheel, and cycle. In order to regroup our data, we used the WordNet database to find the synonym of the terms and strings that were entered by the graduates in their natural language. Getting the synonyms is required in the data prepressing phase in order to find the meaning of the terms and strings that describe the same object and then unify it in one common term. For example, the "IT manager" and "Information Technology manager" represent the same career domain. However, the clustering techniques will consider the "IT Manager" and "Information Technology Manager" as two different strings. Therefore, unifying the terms or strings into one common term can help the clustering technique to consider it as a same object and cluster it in the same group. A second example, the terms teacher and instructor share the same meaning but they are considered as two different terms in the clustering process. Nevertheless, when the synonym of the two terms is retrieved using WordNet and then unify it in one common term, the clustering technique will cluster it in the same cluster.

Moreover, many misspelled terms and strings were found in our survey entered by the university graduates. Therefore, we implemented the Levenshtein distance in order to compute the match between correct and incorrect terms and strings. The Levenshtein string metric was proposed by Vladimir Levenshtein. The Levenshtein distance measures the dissimilarity between two sequences. The distance between two strings is the least number of single-character alterations needed to transform one string into the other. This metric is applicable in sequence matching and spell checking. To clean our data and avoid loss information we used the Levenshtein distance. This method allows us to clean our dataset based on a reference dataset by computing the similarity between a source column from our dataset and a target column from the reference dataset that contains the correct terms and strings.

Furthermore, records of duplicate data should be deleted from the dataset before the analysis phase starts. Therefore, the Python dropduplicates function was applied to drop duplicate records from our survey data. Moreover, the multi-answer questions were split, using "," as a delimiter. Besides, we used the python Series.str.contains and Series.string.split functions to find specific terms and split each row in the series based a delimited.

The Ontology Design Phase: This phase represents the ontology that forms the model, individuals, and provides a semantic description of the education domain and career domain knowledge. The CBR recommender systems take advantage of this domain knowledge to obtain accurate results. The student and graduate's profile, school, higher education, and careers domains were described in an ontology using the Protégé OWL editor. This ontology is integrated into the KB recommender system in order to increase the accuracy of the recommendations. This ontology encompasses two main segments; the first segment is the Graduates that describes all the graduates' instances in the knowledge base such as their career interests, preferred courses, country, hobbies, etc. The second segment is the rest of the concepts in the ontology structure that describes all the attribute concepts of the student and education domain. Figure 4 illustrates the graph of our ontology. This figure represents the depth of the subclass hierarchy, which aids in the computation of the similarity measure.

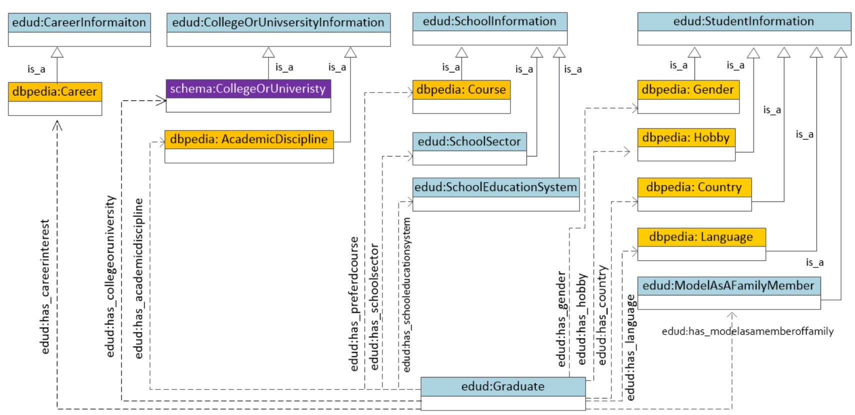


Fig. 4. The graph of the ontology design

The Hybrid Recommendation Engine Phase: Once the data and ontology are prepared, the RS computes the similarities and provides recommendations to the active student. The engine of this hybrid RS incorporates two core recommender systems namely the KB and CF illustrated in figure 6. The main function of the CF system is to compute the users' ratings and find similarities in order to generate appropriate recommendations. Then the output of the CF recommender system will be integrated as a new feature into the KB recommender system in order to recommend personalized recommendations. In the KB system, the semantic similarity is computed through the ontology structure based on the hierarchical order between the ontology concepts. This collaboration strategy between the KB and CF recommender systems is based on the "Feature Augmentation" hybrid strategy [7].

Researchers categorized the RS hybridization into two main cases:

– The first case is the uniform in which one RS algorithm has better precision than another algorithm over the entire space of recommendation. For instance, the Cascade

strategy with the stronger RS given higher priority, the Feature augmentation strategy in which the weaker RS algorithm performs as an assistant contributing a small amount of info, and the Meta-level strategy in which the stronger algorithm generates a heavy representation that reinforces the performance of the weaker algorithm.
– The second case is the non-uniform in which two recommender algorithms have different powers in different parts of the space. In this case, the process will need to be able to employ the two-recommender algorithms at different times. For instance, the Switching strategy is a natural choice here and needs the system should be able to detect when one algorithm should be favored. The Mixed and Feature combination strategies permit output from both RS algorithms without applying a switching measure.

The following figure illustrates the Feature augmentation hybrid procedure:

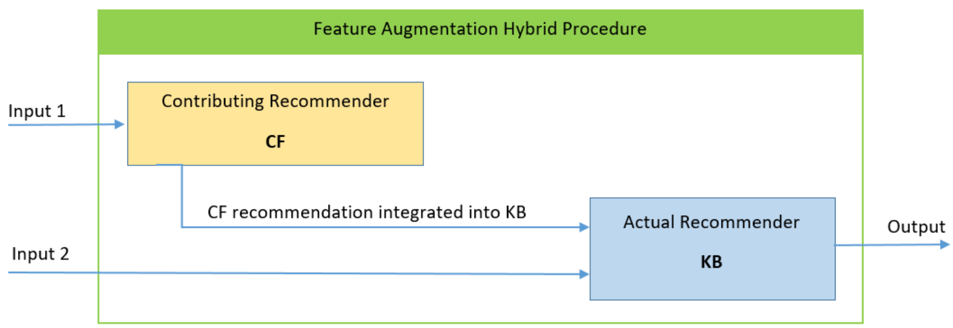


Fig. 5. Feature augmentation hybrid procedure

In our hybridization approach, we implemented the Feature augmentation technique because in our case the KB system is a stronger algorithm that is based on the domain knowledge and the CF is the weaker one that is based on the ratings. This approach enabled a contributing CF recommender to make a positive effect without interfering with the performance of the KB algorithm. Figure 6 shows the architecture of the proposed hybrid RS that integrates the User-based CF and KB algorithms. The proposed approach comprises 3 core modules described as follows:

– The first module illustrates the domain knowledge, which integrates the concepts and individuals of higher education, career, and students. The domain knowledge is formally represented in an ontology.
– The second module illustrates the hybrid RS engine, which incorporates the KB and User-based CF sub-systems. The CF system's role is to compute the k-most similar student and generate recommendations whereas the KB role is to generate the overall personalized recommendations to the active user based on ontology and CBR system.
– The third module illustrates the profile, rating, and query of the active student. In this module, the active user inputs his/her course/career's ratings and queries through a GUI in order to get recommendations. Course and career ratings are integrated into the CF system and the queries are integrated into the KB system.

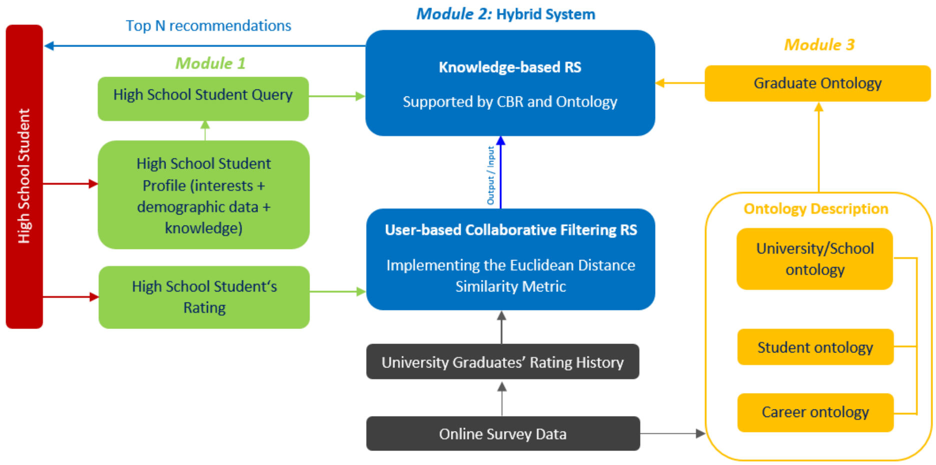


Fig. 6. The architecture of the proposed hybrid RS

These modules interconnect in a hybrid mechanism process and produce recommendations based on the active user's preference, interest, demographic data, and ratings. The following section presents the implementation and evaluation of the User-based CF and Item-based CF algorithms based on many similarity metrics, and the KB hybrid RS.

The Experimental Procedure A hybridization technique for basic RSs is needed to address the limitations and problems of some basic filtering approaches. Therefore, in this section, we implemented and evaluated the CF and KB algorithms in order to demonstrate the efficiency of our proposed KB hybrid RS. Thus, we experimented, evaluated, and tested the following three recommendation strategies:

- The User-based CF technique
- The Item-based CF technique
- The KB hybrid RS incorporated with the User-based CF technique and supported by the CBR and ontology.

The experimental study of the User-based and Item-based CF algorithms: In order to conduct the CF experiments, we extracted and refined a dataset from our online survey. This dataset encompasses 469 objects and 39 attributes. The objects represent the university graduates that have a job interest similar to their actual job. Besides, 39 attributes represent the items' ratings. This dataset contains about 11,000 ratings from 469 users on 39 items. All users in the dataset rated at least 20 items. The dataset involves correct real-world data that will ensure the accuracy of our experiments' returns. The correctness of our dataset is ensured by the way we disseminated and collected the survey entries. This survey was disseminated to university graduates that study in different disciplines and real employees that work in different domains. In addition, the data collection process involved face-to-face interviews to fill the intended survey. Since the selected dataset involve only graduates having a job interest similar to their actual job, we considered it a trusted real word dataset.

This experimental study divides the dataset into two sub-datasets. The first sub-dataset contains the training data and the second sub-dataset contains the testing data to test it. For each similarity metrics, evaluation has been implemented based on the MAE and RMSE. Since this experiment is based on item ratings, we implemented and evaluated the User-based and Item-based CF algorithms based on the Euclidean Distance Similarity, Pearson Correlation Similarity, Spearman Correlation Similarity, Uncentered Cosine Similarity, and City Block Similarity. The main function of the mentioned metrics is to find similarities between graduates based on their ratings. Then, the CF recommender system will recommend a career that is appropriate to the user's interests. In this experiment, some parameters have been determined such as the N neighborhood size, and training ratio of the experiment. In addition, the effects of different CF algorithms and similarity metrics were considered. The N neighborhood represents the nearest-neighbors to the object location. With user neighborhood, the RS can find the most similar user for the selected user. The training ratio represents the percentage of each user's preferences to use to produce recommendations; the rest are compared to estimated preference values to evaluate recommender performance.

To evaluate our CF recommender system, we implemented the mahout evaluation method [1]. The evaluate method evaluates the accuracy of RSs' recommendations. Applications will take a percentage of the preferences provided by the given DataModel as training data. This is classically most of the data, like 90 percent. This data is used to produce recommendations, and the rest of the data is compared against estimated preference values to see how much the recommender's predicted preferences match the user's real preferences. Precisely, for each user, this percentage of the user's ratings are used to produce recommendations, and for each user, the remaining preferences are compared against the user's real preferences. The return is a score representing how well the recommender's estimated preferences match real values. Lower scores mean a better match and 0 is a perfect match.

The experiment evaluation results based on different Neighborhood sizes: The size of the Neighbor can affect the prediction quality. By changing the number of neighbors, the sensitivity of the neighborhood is determined. In this section, the User-based and Item-based CF algorithms are evaluated and tested based on many similarity metrics and neighborhood sizes. The result of the experiments shows that the User-Based CF algorithm and the Euclidean Similarity metric have the lowest MAE equal to 0.45 and RMSE equal to 0.58, which means they predict better. All our experiments showed that the User-based CF algorithm and the Euclidean Similarity metric with Neighborhood size equal to 50 and Training Ratio equal to 0.8 have the lowest RMSE and MAE, which means they predict better. Therefore, we selected the Euclidean similarity metric as an appropriate technique for our CF recommender engine. The recommendations of the proposed User-based CF algorithm should be like:

- Recommended Academic Discipline: [Information Technology, value: 3.0]
- Recommended Academic Discipline: [Architecture and Construction, value:3.0]
- Recommended Academic Discipline: [Business, Management, and Administration, value: 3.0]
- Recommended Academic Discipline: [Finance, value: 3.0]

In the above recommendations, the term value represents the higher rate of the recommended item. The RMSE and MAE results are illustrated in figures 7 and 8.

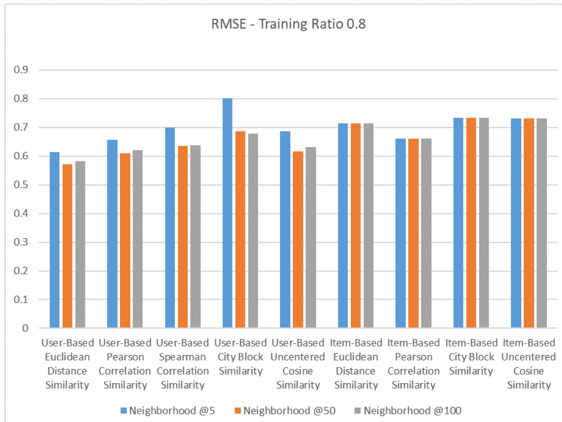


Fig. 7. RMSE for User-based and Item-based similarities with training ratio equal to 0.8

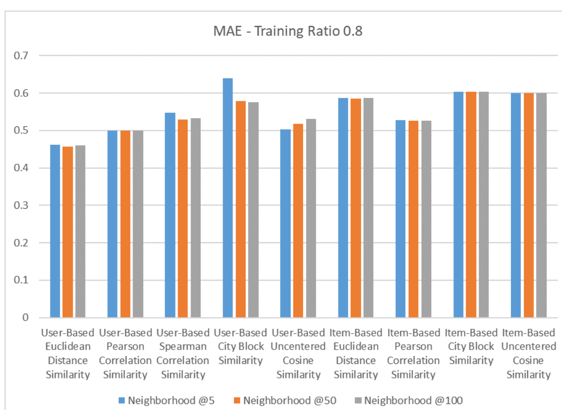
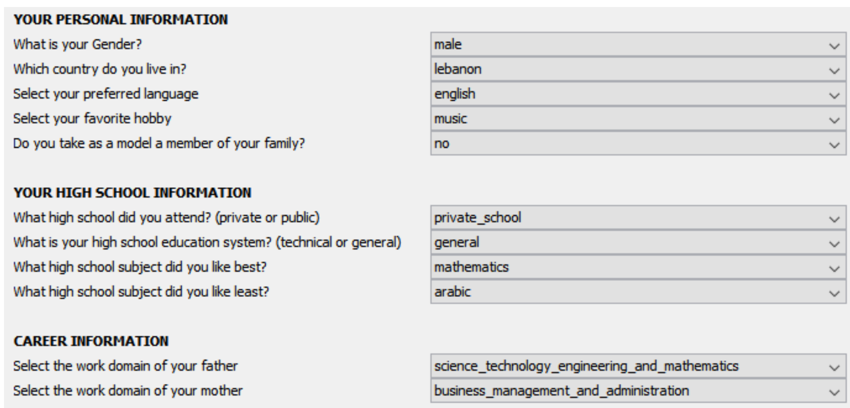


Fig. 8. MAE for User-based and Item-based similarities with training ratio equal to 0.8

At this point, the active student inputs his course ratings into the GUI of the system. This input permits the CF engine to generate recommendations based on the graduates' rating history. Then, the output of the CF system is integrated automatically as a new feature into the KB recommender system. This new feature will be used in the KB system as a support knowledge to the student interests' query. This query is submitted by the active student in order to feed the KB system. A query sample that describes a student's interests and demographic data is shown in Figure 9. Finally, the KB algorithm generates the top N recommendations based on the student's query.



YOUR PERSONAL INFORMATION	
What is your Gender?	male
Which country do you live in?	lebanon
Select your preferred language	english
Select your favorite hobby	music
Do you take as a model a member of your family?	no
YOUR HIGH SCHOOL INFORMATION	
What high school did you attend? (private or public)	private_school
What is your high school education system? (technical or general)	general
What high school subject did you like best?	mathematics
What high school subject did you like least?	arabic
CAREER INFORMATION	
Select the work domain of your father	science_technology_engineering_and_mathematics
Select the work domain of your mother	business_management_and_administration

Fig. 9. Query sample

Figure 10 and 11 show that graduate case 584 and graduate case 601 are the most similar cases to the current active student query. If we compare the active student's query with the recommendations' result we notice that case 584 is more similar to the active student than case 601. This reveals that case 584 is totally similar to active student query and case 601 is diffirent in the hobby, mother work and graduate interest career attributes.

In our proposed hybrid system, the Feature augmentation strategy enabled the recommender engine to incorporates two separate types of recommender algorithms in a way that the output of the first recommender is fed into the input of the second recommender. In addition, the Feature augmentation strategy improved the performance of the proposed hybrid system and made a significant contribution to the quality of recommendations.

Finally, Hundreds of queries were processed by our hybrid RS approach. The experiments show the accuracy of COHRS based on two criteria namely the "accuracy of retrieving the most similar cases" and the "accuracy of generating personalized recommendations". To evaluate the accuracy of COHRS approach, the HoldOutEvaluator and SameSplitEvaluator algorithms were implemented. More information about these two algorithms are presented in section 2.2.4. The evaluation results show the high accuracy of COHRS based on using several percent of the dataset for testing and performing the process several times through many cycles. The evaluation results show the percentage of accuracy of COHRS based on many experiments as follows: 98 percent accuracy for

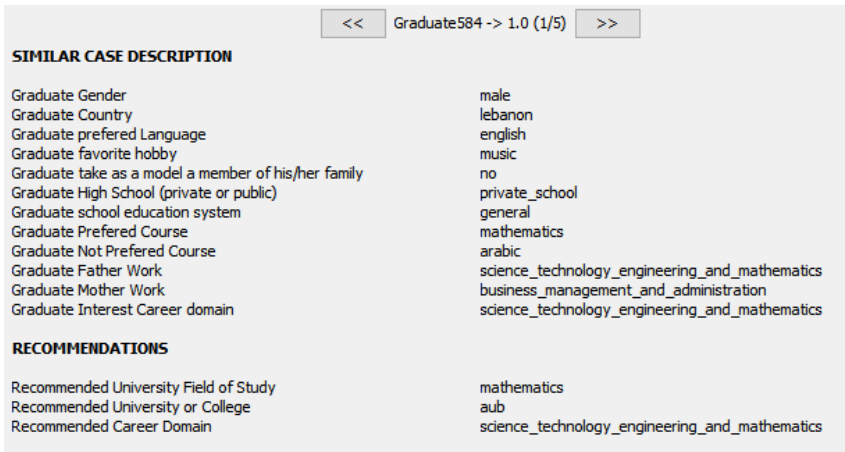


Fig. 10. Hybrid RS result (Top most similar case 584)

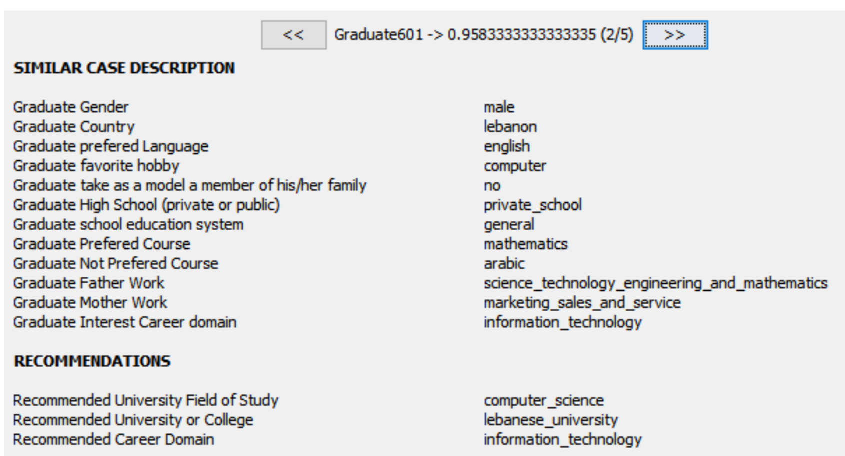


Fig. 11. Hybrid RS result (second most similar case 601)

"retrieving the most similar cases" and 95 percent accuracy for "generating personalized recommendations".

Besides, our analysis revealed that this hybridization approach is adequate to our high dimensional dataset that encompasses more than 50 heterogenious attributes. Furthermore, we noticed that the more knowledge we acquire the more effective the ontology-based hybrid RS could be. The novelty of our method focuses precisely on CBR and ontology-based hybrid RS within the higher education domain, of which to the best of our information, no research has been conducted using COHRS approach and presented this domain. Thus, we consider COHRS an effective approach for designing KB hybrid RS that support students in their higher education choices.

4. Conclusion

In this study, we proposed a novel hybrid RS approach named COHRS that incorporates the CF and KB recommendation techniques. This hybrid system is supported by CBR and ontology technologies. The purpose of this hybridization technique is to recommend to high school students appropriate universities/colleges, university majors, and career options. The recommendations of the proposed system are based on students' demographic data, course and career ratings, and the higher education domain. The experiments in this work enabled us to identify the appropriate similarity metrics, neighborhood size, and CF algorithm for our hybrid RS engine. Our contribution in this study is threefold: First, assisting high school students to find appropriate universities and university majors through a hybrid RS based on their interests and preferences. Second, proposing a novel hybridization approach that incorporates many recommendation algorithms. Third, minimizing the limitations of traditional RSs such as treating high dimensional datasets and heterogeneous data types. Additionally, solving problems that encounter most RSs such as the Gray-sheep problem. In future work, we will demonstrate the efficiency of the proposed hybrid RS approach by conducting more experiments and studying more RS algorithms and hybridization approaches.

References

1. Apache Mahout Essentials. Packt Publishing. ISBN:978-1-78355-499-7. (2015)
2. Badrul M. Sarwar, Joseph A. Konstan, Al Borchers, Jon Herlocker, Brad Miller, and John Riedl. 1998. Using filtering agents to improve prediction quality in the GroupLens research collaborative filtering system. In Proceedings of the ACM conference on Computer supported cooperative work - CSCW '98, 345–354. https://doi.org/10.1145/289444.289509. (1998)
3. Badrul Sarwar, George Karypis, Joseph Konstan,John Riedl.. Item-based Collaborative Filtering Recommendation Algorithms. (2001)
4. Basu, C., Hirsh, H. and Cohen W.. Recommendation as Classification: Using Social and Content-Based Information in Recommendation, in: Proc.the 15th National Conference on Artificial Intelligence, Madison WI 714-720. (1999)
5. Bobadilla, J., Ortega, F., Hernando, A., Bernal, J.. Acollaborative filtering approach to mitigate the new user cold start problem. Knowledge-Based Systems 26, 225-238. (2012)
6. Bruke, R.. Knowledge-based Recommender Systems. Encyclopedia of Library and Information Syst. 69(32). (2000)

7. Bruke, R.. Hyprid Recommender Systems. Survey and Experiments. User Modeling and User-Adapted Interaction 4:331-370. (2002)
8. Breese, J. S., Heckerman, D. and Kadie, C.. Empirical analysis of predictive algorithms for collaborative filtering. Computer Networks and ISDN Systems, pp. 43-52. (1998)
9. Burke, R.. Integrating knowledge-based and collaborative-filtering recommender systems. In Proceedings of the Workshop on AI and Electronic Commerce, pp. 69–72. (1999)
10. Burke, R.. Hybrid Web Recommender Systems. Springer Berlin Heidelberg, pp. 377 - 408. (2007)
11. Chavarriaga O., Florian-Gaviria B., Solarte O.. A Recommender System for Students Based on Social Knowledge and Assessment Data of Competences. In: Rensing C., de Freitas S., Ley T., Muñoz-Merino P.J. (eds) Open Learning and Teaching in Educational Communities. EC-TEL. Lecture Notes in Computer Science, vol 8719. Springer, Cham. (2014)
12. David Goldberg, David A. Nichols, Brian M. Oki, Douglas B. Terry.. Using collaborative filtering to weave an information TAPESTRY. Communications of the ACM 35(12):61-70. DOI: 10.1145/138859.138867.(1992)
13. Deeksha and S. Sahu, ”Finding similarity in articles using various clustering techniques,” 2017 6th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, pp. 344-347, doi: 10.1109/ICRITO.2017.8342449. (2017)
14. Desrosiers, C., Karypis, G.. A comprehensive survey of neighborhood-based recommendation methods, recommender systems handbook, (pp. 107–144). Berlin: Springer. (2011)
15. E. by David Leake. Case Based Reasoning. Experiences, Lessons and Future Directions. AAAI Press. MIT Press, USA. (1997)
16. F. S. Gohari, M. J. Tarokh.. Classification and Comparison of the Hybrid Collaborative Filtering Systems. Int. J. Res. Ind. Eng. 6 (2) (2017) 129-148. (2017)
17. G. Adomavicius, N. Manouselis, Y. Kwon.. Multi-criteria recommender systems in Recommender Systems Handbook, ed. by F. Ricci, L.Rokach,B. Shapira (eds.) (SpringerUS), pp. 769–803.(2011)
18. Gaurav Agarwal1, Himanshu Bahuguna2, Ajay Agarwal. Solving Cold-Start Problem in Recommender System Using User Demographic Attributes. International Journal on Emerging Technologies (Special Issue NCETST-2017) 8(1): 55-61. (2017)
19. Ha, H.; Han, H.; Mun, S.; Bae, S.; Lee, J.; Lee, K. . An Improved Study of Multilevel Semantic Network Visualization for Analyzing Sentiment Word of Movie Review Data. Appl. Sci. 2019, 9, 2419. (2019)
20. Herlocker, J.L., Konstan, J.A., Borchers, A. and Riedl, J.. An Algorithmic Framework for Performing Collaborative Filtering. Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 230-237. (1999)
21. Idir Benouaret. Un système de recommandation contextuel et composite pour la visite personnalisée de sites culturels. Autre [cs.OH]. Université de Technologie de Compiègne,. Français. ¡NNT : 2017COMP2332¿. (2017)
22. J.B. Schafer, D. Frankowski, J. Herlocker, S. Sen.. Collaborative filtering recommender systems. The Adaptive Web, 291-324. (2007)
23. Jihane Karim, Matthieu Manceny, Raja Chiky, Michel Manago, Marie-Aude Aufaure. Using Collaborative Filtering to Enhance Domain-Independent CBR Recommender's Personalization. IEEE 9th International Conference on Research Challenges in Information Science (RCIS), Athens, Greece. ⟨hal-02402483⟩. (2015)
24. João Pedro B. Ferreira, Franciscone L. A. Junior, Renata L. Rosa, and Demóstenes Z. Rodríguez. Evaluation of Sentiment and Affectivity Analysis in a Blog Recommendation System. In Proceedings of the XVI Brazilian Symposium on Human Factors in Computing Systems (IHC 2017). Association for Computing Machinery, New York, NY, USA, Article 25, 1–9. DOI:https://doi.org/10.1145/3160504.3160559. (2017)

25. John K. Tarus, Zhendong Niu, and Abdallah Yousif. A hybrid knowledge-based recommender system for e-learning based on ontology and sequential pattern mining. Future Generation Computer Systems 72: 37–48. https://doi.org/10.1016/j.future.2017.02.049. (2017)
26. John Tarus, Zhendong Niu, and Bakhti Khadidja. E-Learning Recommender System Based on Collaborative Filtering and Ontology. International Journal of Computer and Information Engineering 11, 2: 256–261. (2017)
27. Kantor, P. B., Rokach, L., Ricci, F., Shapira, B.. Recommender systems handbook. Springer, New York, Dordrecht, Heidelberg, London (Vol. I). ISBN 978-0-387-85819-7. DOI 10.1007/978-0-387-85820-3. (2011)
28. Keshav R, et al,. (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (3), 4782-4787. (2014)
29. LOPEZ, Maritza Bustos et al.. EmoRemSys: An educational recommender system by using emotions detection. RISTI [online]. n.17, pp.80-95. ISSN 1646-9895. http://dx.doi.org/10.17013/risti.17.80-95. (2016)
30. Mei-Hua Hsu. A personalized English learning recommender system for ESL students. Expert Systems with Applications 34, 1: 683–688. https://doi.org/10.1016/j.eswa.2006.10.004. (2008)
31. Minh-Phung Thi Do, Dung Van Nguyen, Loc Nguyen.. Model-based Approach for Collaborative Filtering. Ho Chi Minh city, Vietnam. (2010)
32. Mohammad Hamidi Esfahani, Farid Khosh Alhan. New Hybrid Recommendation System Based On C-Means Clustering Method". 5th conference of information Knowledge. IEEE. (2013)
33. Mohammed Essmat Ibrahim. An Ontology-based Hybrid Approach to Course Recommendation in Higher Education. 167. (2019)
34. Montaner, M., Lopez, B. and De la Rosa J.L.. A Taxonomyof Recommender Agents on the Internet, Artificial Intelligence Review, Kluwer Academic Publisher, 285 –330. (2003)
35. Owen S., Anil R., Dunning T. and Friedman E.: Mahout In Action. Manning Publications Co. ISBN 978-1-9351-8268-9. (2012)
36. Paula A. Rodríguez, Demetrio A. Ovalle, and Néstor D. Duque. A Student-Centered Hybrid Recommender System to Provide Relevant Learning Objects from Repositories. In Learning and Collaboration Technologies (Lecture Notes in Computer Science), 291–300. https://doi.org/10.1007/978-3-319-20609-7-28. (2015)
37. Pazzani M.J., Billsus D.. Content-Based Recommendation Systems. In: Brusilovsky P., Kobsa A., Nejdl W. (eds) The Adaptive Web. Lecture Notes in Computer Science, vol 4321. Springer, Berlin, Heidelberg. (2007)
38. Ruchika, Singh, Ajay Vikram, Sharma Dolly. Evaluation Criteria for Measuring the Performance of Recommender Systems, IEEE. (2015)
39. Sarik Ghazarian, Mohammad Ali Nematbakhsh.. Enhancing memory-based collaborative filtering for group recommender systems. DOI: 10.1016/j.eswa.2014.11.042. (2015)
40. Simone Kopeinik . Applying Cognitive Learner Models for Recommender Systems in Sparse Data Learning Environments. SIGIR Forum 51, 3 (December 2017), 165. DOI:https://doi.org/10.1145/3190580.3190608. (2018)
41. Z. Bahramiana, R. Ali Abbaspoura. An ontology-based tourism recommender system based on spreading activation model. Doi:10.5194/isprsarchives-XL-1-W5-83-2015. (2015)

Charbel Obeid is a Computer Science lecturer at AUL University - Lebanon. He received his Ph.D. degree in Computer Science from Claude Bernard University Lyon 1 - France. His research interests include artificial intelligence, data mining, recommender systems, Semantic Web, and ICT in education. He is the author of two journal papers and one conference paper. Also, he works as a technical engineer at 6SS a company that provides security solutions powered by Milestone XProtect.

Christine Lahoud is an Assistant Professor and the coordinator of the faculty of Engineering at the French University in Egypt, Cairo-Egypt. She obtained a Ph.D. degree in Computer Engineering from the University of Technology of Belfort-Montbeliard (2013), France. She joined then the department of Computer Engineering at the University of Reims in France and then Galatasaray University in Istanbul, as an assistant professor. She is currently doing research on recommendation systems. Her research interests include recommendation systems, knowledge management, semantic web, and artificial intelligence. She has been a member of the Program Committee of several conferences. She organized several workshops on Knowledge management as KARE 2011, EKM (2014, 2016, 2018), and a IJCEELL special issue on Knowledge Management Technologies in Education (2020).

Hicham El Khoury is an associate professor at the Lebanese University and a member of the LaRRIS Lab. He got his master's degree in Modeling from the Lebanese University and doctoral degree in Computer Science and Telecommunications from Paul Sabatier – Toulouse III. His research interests include Network Security Management Information, AI, clustering techniques, e-learning, EdTech, and ICT in Education. He is an author or co-author of 5 journal papers, 13 conference and workshop papers. He is also an ICT and Services Consultant at Lebanon-CERD (Center for Educational Research and Development) since 2017 and UK Lebanon Tech Hub since 2021.

Pierre-Antoine Champin is an associate professor at Université Claude Bernard Lyon 1, and a W3C fellow. His research interests include data and knowledge interoperability at web-scale, and the design of intelligent interactive systems. He has been representing Université de Lyon in various W3C working groups (Media annotation, RDF, LDP). He is an author or co-author of one book chapter, eleven journal papers, 71 conference and workshop papers, and four W3C standards. He is also a member of program committees in major Web and Semantic Web conferences (WWW, ISWC, ESWC).

Received: Jun 18, 2021; Accepted: February 27, 2022.