IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 June 2023, accepted 22 June 2023, date of publication 28 June 2023, date of current version 6 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3290918

RESEARCH ARTICLE

Evolutionary Approach for Building, Exploring and Recommending Complex Items With Application in Nutritional Interventions

BARTOLOME ORTIZ-VISO, ANDREA MORALES-GARZÓN, MARIA J. MARTIN-BAUTISTA, AND MARIA-AMPARO VILA

Department of Computer Science and Artificial Intelligence, University of Granada, 18071 Granada, Spain

Corresponding author: Bartolome Ortiz-Viso (bortiz@ugr.es)

This work was supported in part by the European Union (Stance4Health) under Grant 816303, in part by Ministerio de Ciencia e Innovación under Grant PID2021-123960OB-I00 funded by MCIN (Ministerio de Ciencia e Innovación)/AEI (Agencia estatal de Investigacion)/10.13039/501100011033 and by ERDF (European Regional Development Fund) A way of making Europe.
And in part under Grant TED2021-129402B-C21 funded by MCIN (Ministerio de Ciencia e Innovación)/AEI (Agencia estatal de Investigacion)/10.13039/501100011033 and, by the European Union NextGenerationEU/PRTR (Plan de Recuperación, Transformación y Resiliencia). This work is part of the doctoral thesis of Bartolome Ortiz-Viso conducted within the context of the ''Program of Information and Communication technologies'' at the University of Granada.

ABSTRACT Over the last few years, the ability of recommender systems to help us in different environments has been increasing. Several systems try to offer solutions in highly complex environments such as nutrition, housing, or traveling. In this paper, we present a recommendation system capable of using different input sources (data and knowledge-based) and producing a complex structured output. We have used an evolutionary approach to combine several unitary items within a flexible structure and have built an initial set of complex configurable items. Then, a content-based approach refines (in terms of preferences) these candidates to offer a final recommendation. We conclude with the application of this approach to the healthy diet recommendation problem, addressing its strengths in this domain.

INDEX TERMS Applied computing, complex recommendation systems, human–computer interaction, information retrieval, recommendation systems.

I. INTRODUCTION

Recommendation systems (hereinafter referred to as **RS**) are a type of information filtering system that seeks to predict the preference a user would give to a set of items [1]. In our daily lives, RS usually serve as tools to simplify our experience in technology-based services (such as e-commerce) by offering items to users based on their characteristics or specific requests. However, RS are used in a variety of areas, each with its own unique features that present new and different challenges.

RS have two main characteristics that define the ''recommendation problem'': the data used (both raw and processed) and the recommendation model.

When describing the data, RS use two sources of information to a greater or lesser extent: users and items. These sources can be combined or used in different sub-processes of the system. Users can be defined as entities (not only people) that request recommendations. From these users, depending on our recommendation strategy, we will obtain information about them, their search requirements, or their historical interactions with the items. Secondly, the items are the elements that we are going to recommend to our users. Again, we can have a wide variety of information about them, such as their intrinsic characteristics, the tags that describe them, or the type of users that rate them positively.

In addition to the aforementioned categories, there exists another significant category known as a knowledge-based data source. This particular category is not universally present in all recommendation strategies [2], and we shall make reference to it whenever we utilize data sources that do not align with either the Users or Items category. Examples of such data sources include contextual information, expert opinions, and temporal decisions. In numerous scenarios, incorporating this

The associate editor coordinating the review of this manuscript and approving it for publication was Pasquale De Meo.

additional knowledge is crucial for providing high-quality recommendations, particularly in situations where multiple factors are at play or can influence the user's life and choices. In essence, we employ these sources when we believe that gaining a broader and more expert perspective on the problem at hand enhances the final recommendation.

The manner in which we incorporate these three data sources into the recommendation process serves as a broad categorization of the type of recommendation we are formulating and the strategies we can employ. Consequently, employing item characteristics to anticipate user preferences, as in the case of content-based filtering, differs from utilizing the preferences of a large user base to predict the preferences of a new user, which is a technique referred to as collaborative filtering.

However, as we have seen, the new opportunities presented by the processing of large amounts of different data (using technologies such as deep learning [3]) have led to the creation of increasingly hybrid systems. These systems are able to encode and use a more complex representation of user, contextual, and item data, leading us to a research landscape where many systems operate in a hybrid way.

These changes affect not only the algorithms and recommender strategies but also the problems that we can solve through these systems.

While much of the research focuses on what we might call simple items, access to more data allows us to work on more challenging everyday problems. To exemplify this difference, we can compare the problem of recommending a unique dish for dinner versus recommending a weekly diet [4].

If we wanted to recommend a single dish, we would have a well-defined and bounded item with a finite set of characteristics. We may have a lot of information about a specific dish, both objective (cooking, nutritional value) and subjective (many people like it, it is popular, it is very spicy). However, the item is defined and bounded. To recommend a diet, on the other hand, we must consider not only the factors mentioned above but also the food benefits for the user, the combinations of ingredients we can offer, the combinations that can negatively affect the user, and even the elements we should recommend, even if they do not align with the culinary preferences of our user.

Therefore, the problem becomes more complex, partly due to the possible internal structure of the recommendation (viewing a diet as a structure built by different dishes) and partly due to the significant impact it can have on the user's life (a single dinner versus a daily diet over time).

These two characteristics (complexity, structure) define the types of problems we intend to address in this work and motivate the recommendation system we propose, along with its main features.

After the introduction, in Section II, we will provide a formal definition of the terms describing our problem, along with examples of these terms. Section III will describe the theoretical details of our recommender system, outlining its different components. Next, we will explain how to apply our approach to a real-world application and discuss their benefits in Section IV. Finally, we will conclude the article with Section V, where we will discuss some considerations regarding our model and outline future research directions.

II. RELATED WORK: STRUCTURED RECOMMENDATIONS IN COMPLEX SCENARIOS

As mentioned in the introduction, this study and the recommendation model proposed focus on specific problems that fulfill this scenario: we want to recommend a structured object from a set of simpler items and we want to deal with multiple objectives that have different levels of precision. We are going to refer to this two fundamental characteristics as structured recommendation and multiobjetive recommendation.

- **Multiobjetive recommendation**: Multi-objective recommendation involves generating personalized recommendations that consider multiple conflicting objectives or criteria simultaneously. This area has become quite relevant, since we have now bigger amount of data to produce the recommendations process. But despite its use on several areas [5], [6], some of these works focus on specific aspects of the recommendation process (diversity, novelty), that is not our case. We are going to deal with scenarios from real-world applications. We call this scenario "complex scenarios".
  Complex scenarios force us to deal with the sources of constraint: knowledge-based constraints, expert-advise constraint and user-based rules (or user preferences). The complexity of the input is often determined by the problem and the data we can access. Examples of application areas facing such situations are those related to tourism or housing choice [7], [8], [9]. But it is in health recommender systems [10] where we found natural examples of multiple constraint from multiple sources. As examples, it is already common to have systems to help us decide what to eat [11] (more on this topic in Section IV), how we should exercise [12], [13] or what to do when we are ill [14].
- **Structured recommendation**:
  If the items we recommend have a structure (and this structure is involved in the recommendation process) our system will be a structured recommender system.
  This type of system is often explored in the scientific literature through sequential or set recommender systems in multiple areas (fashionable sets of [15], [16], and [17] or cultural travel [18], [19]).
  However, in this paper, we will refer to a structured recommendation when we recommend a set of interrelated items, following a design/structure that can be customized according to the user. It should be noted that both the structure, and the set of items in it, as a whole, are the output of the system and is presented as such to the user. The system output does not constitute a

recommendation sequence in time, with multiple items one after another.

Although this approach could affect the adaptability of the output, it gives our system a greater ability to incorporate more relationships and rules. Furthermore, the system can be re-evaluated if the user or the situation requires modifications. As a consequence, any part of the structure can be modified, to a certain limit.

III. EVOLUTIONARY STRUCTURED RECOMMENDATION ALGORITHM

In this Section, we are going to present our proposed algorithm (see Figure 1) for complex structured recommendations. During the text, some brief examples will appear to clarify the concepts discussed. We will expand these applications upon Section IV.

It is worth noting that definitions in the next subsections can be adapted to fulfill the specific requirements of our problem. This flexibility is necessary due to the different nature of the problems approachable with this model. We offer a secondary figure 3 where all variables depicted in this section can be easily consulted.

A. SOURCES OF DATA AND MAIN OBJECTIVE OF THE STUDY

One of the crucial parts of any recommendation process is the acquisition of the necessary data. In our case, the data sources needed may vary in nature, extent and characteristics, but they always correspond to three essential elements in our system.

The first two, as in any recommendation, are the space of Items to recommend and the space of Users. We denote the spaces of these elements by $\mathcal{I}$ and $\mathcal{U}$ respectively. The elements of each of these spaces belong to p-dimensional, q-dimensional space and are encoded by an array of parameters, $\{i_1, \dots, i_p\} = \mathcal{I}$ and $\{u_1, \dots, u_q\} = \mathcal{U}$. Moreover, we need an additional source of information that contain expert knowledge $\mathcal{K}$ about our problem. This last source of data can affect any phase of the recommendation, often formulated as a set of parameters or constraints on the set $\mathcal{I}$ denoted by $\mathcal{K} = \{k_1, \dots, k_l\}$. In both $\mathcal{U}$, $\mathcal{K}$ we have all the information related to the structure of our future recommendation and the relationships between items.

The final items recommended are not those elements of our initial data source $\mathcal{I}$. Items in $\mathcal{I}$ are just a lower conceptual unit with no structure and no relationship between its parts. In some problems, we may obtain a rich source of data that contains complex objects. But in reality, this is far from usual, and not realistic at all dealing with personalized problems. From now on we will denote the elements of $\mathcal{I}$ as sub-items in order to differentiate them from the definitive Items.

Thus, our first goal is the generation of the definitive Items which built our final recommendation. In the next subsection we elaborate further on their generation, but we identify here that the set of these items obeys the form $\mathcal{L}(\mathcal{I}, \mathcal{U}, \mathcal{K}) = \{\{1_i, \dots, a_i\}, \dots, \{1_i, \dots, b_i\}\}$. This form represents sets of elements of $\mathcal{I}$ that are dependent on the sub-items which will form it, the constraints proposed by the user and the characteristics proposed by other knowledge-based sources. For simplicity, we will denote it as $\check{\mathcal{I}} = \mathcal{L}(\mathcal{I}, \mathcal{U}, \mathcal{K})$ from now on. Note that this set may not be a subset of the set of parts of $\mathcal{I}$, up to this point, unless otherwise stated, repetitions of elements of $\mathcal{I}$ could occur in it.

With these sets we can now define the objective of our system: Given an initial set $\mathcal{I} = \{1_i, \dots, n_i\}$ we want to obtain a subset of structured combinations of elements of $\mathcal{I}$ denoted $\check{\mathcal{I}}$ for each user from the set $\mathcal{U} = \{1_u, \dots, m_u\}$, so that it optimizes the needs and some preferences of each user as well as the parameters from $\mathcal{K} = \{k_1, \dots, k_l\}$.

To better understand this definition, we can think of a classic recommendation problem such as personalized nutrition. In this case, we would have the recipes as the elements of $\mathcal{I}$ with their respective quantities of ingredients. $\mathcal{U}$ would be the set of users of the application, each one with different characteristics such as weight, age, or what pathologies they may suffer from. In addition, $\mathcal{K}$ would be the set of all the parameters that a healthy diet needs to satisfy (percentage of kilocalories, fat, or seasonality, amongst others). The objects that will appear in $\check{\mathcal{I}}$ will be groups of recipes (associated with a meal structure) depending on the user and the nutritional information we have.

B. CREATION OF THE ITEM SPACE

The final item space $\check{\mathcal{I}}$ is the space in which the logic of our recommender system operates. Obtaining a space of this complexity can be a hard and time-consuming task due to difficulties in finding and processing useful databases. For this reason, our system uses data with a lower conceptual level as bricks to build the final dataset.

This approach will favour the adaptation of the space $\check{\mathcal{I}}$ to the initial information and to those parameters of $\mathcal{K}$ and $\mathcal{U}$ involved in this part.

This task works through an evolutionary algorithm whose objective is to obtain a subset of elements from $\check{\mathcal{I}}$ that satisfies $\{u_{1+x}, \dots, u_{q+x}\}$ y $\{k_{1+z}, \dots, k_{l-z}\}$.

Note that these parameters, by notation, are a sub-set of the total parameters. It is worth remarking this, since some of them may be used elsewhere in the system or with other functions.

Within these parameters, there are two distinct types:

- On the one hand, we will denote $\mathcal{E}(\mathcal{U}, \mathcal{K})$ as those parameters that intervene in the generation of the structure. Although they affect characteristics such as convergence or the speed of the algorithm, they do not intervene directly in the evaluation of that candidate. We can think of them as parameters that only admit two values, whether they are in the final item or not, and therefore we will eliminate any element that does not have them or, if necessary, we will mark them as a condition to be fulfilled before evaluating the fitness of these objects. This step can also be considered as a preprocessing step if the constraint affect solely to the items that are

TABLE 1. Summary of the variables used in the model.

Variables	Explanation	Notation	Nutrition application
Preliminary Items	Items that will form the structured recommended object	I	Recipes
User	Users' characteristics from those that will request a recommendation	U	Set of user characteristics
Additional data sources	Additional information (knowledge/expert - based)	K	Nutrient levels, number of dishes
Structured final item to recommend	Final items build by the simpler items with an structure	$\check{\mathcal{I}} = \mathcal{L}(\mathcal{I},\mathcal{U},\mathcal{K})$	Daily/weekly menu
parameters that intervene in the generation of the structure	How the structure is going to be? What elements are not going to be selected?	$\mathcal{E}(\mathcal{U},\mathcal{K})$	Selected courses, number of certain ingredient, allergies
parameters of the genetic algorithm	parameters that affect how the genetic algorithm perform	$\mathcal{V}(\mathcal{U},\mathcal{K})$	Number of generations, threshold of convergence, scores...
Pool of eligible candidates	Sub set of definitive items where our genetic algorithm search for optimization	$\mathcal{H} \subset \mathcal{I}$	Menus that fit the restrictions

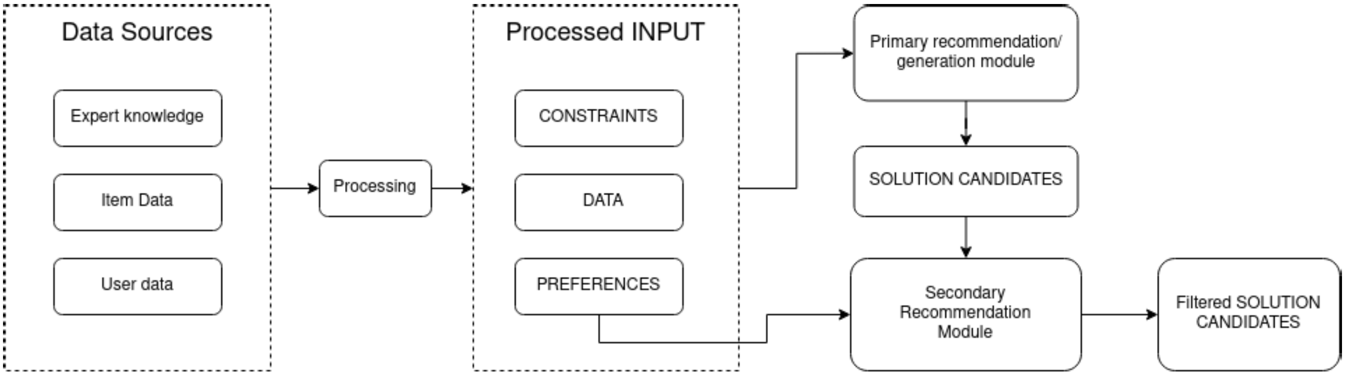

FIGURE 1. Basic scheme of our recommender system. Data sources denote unprocessed raw information. Processed input constitute the final form of the data used in the system. Both preliminary (genetic algorithm) and secondary (content-based) recommendation modules are presented.

considered in the creation of the bundles. However, these constraints can also affect to the whole structure. For example: certain elements are sufficiently spaced within the recommendation or the proportion of them follows a rule (as on per bundle).

- On the other hand, denoted as $\mathcal{V}(\mathcal{U},\mathcal{K})$ are those parameters on which the fitness functions of the genetic algorithm act and which admit adjustments and variations during the generation process.

During the first part of the generation step, we will randomly create elements of $\check{\mathcal{I}}$. This set, which we will call $\mathcal{H} \subset \check{\mathcal{I}}$, verifies $\mathcal{E}(\mathcal{U},\mathcal{K})$ by construction. We then need to describe the fitness function that evaluates which changes of the elements bring us closer to satisfying the parameters in $\mathcal{V}(\mathcal{U},\mathcal{K})$.

For example: in a dietary advice recommender, this step should produce a menu by filling in the selected structure (such as breakfast, lunch, and dinner). This structure is part of the parameters $\mathcal{E}(\mathcal{U},\mathcal{K})$. In this set, we can also have information about allergens or intolerances, which rule out which elements of $\mathcal{I}$ cannot be within the structure.

In $\mathcal{V}(\mathcal{U},\mathcal{K})$ we have characteristics such as "expected total purchase price", "seasonality", "amount of an ingredient", or "amounts of different nutrients", all of them are that the recommendation should have. However their amount have different ranges of acceptability, i.e. total kilocalory consumption in the day may be more or less than 100 kilocalories without affecting the user if the daily average stays close to the objective.

1) FITNESS FUNCTION

In this Section we will focus on those parameters that appear in the set $\mathcal{V}(\mathcal{U},\mathcal{K})$. We denote $\mathcal{V}_\mathcal{U}$ as the set of parameters from the user personalization and $\mathcal{V}_\mathcal{K}$ as the set of parameters from alternative data sources. Then we define the fitness function of our genetic algorithm as:

$$\begin{cases} \mathcal{F}(x) = W_{\mathcal{V}_\mathcal{U}} f_\mathcal{U}(x) + W_{\mathcal{V}_\mathcal{K}} f_\mathcal{K}(x) \\ \text{for all } x \in \mathcal{H} \subset \check{\mathcal{I}} \end{cases} \tag{1}$$

where $W_{\mathcal{V}_\mathcal{U}}$ is the weight associated to user's parameters and $W_{\mathcal{V}_\mathcal{K}}$ is the weight associated to extra data sources and expert knowledge that is taken into account.

The functions $f_{\mathcal{U}}$ and $f_{\mathcal{K}}$ are defined:

$$\begin{cases} f_{\mathcal{U}}(x) = \sum\limits_{\forall u \in \mathcal{V}_{\mathcal{U}}} w_u g_u(x) \\ f_{\mathcal{K}}(x) = \sum\limits_{\forall k \in \mathcal{V}_{\mathcal{K}}} w_k g_k(x) \\ \text{for all } x \in \mathcal{H} \subset \check{\mathcal{I}} \end{cases} \tag{2}$$

with

$$\begin{aligned} g_u(x) &= (\frac{v(optimal_u)}{optimal_u})(u - optimal_u)^2 \\ g_k(x) &= (\frac{v(optimal_k)}{optimal_k})(k - optimal_k)^2 \end{aligned} \tag{3}$$

in this case $optimal_{k,u}$ represents the optimal value to be reached by k, u. In addition, $v(optimal_{k,u})$ is defined as :

$$v(optimal_{k,u}) = optimal_{k,u} * C/(k, u_{range})^2 \tag{4}$$

where k, u_{range} are the specific range of acceptance for k, u and C is the global optimum of the criterion (if we establish that our convergence criterion is acceptable when the value is less than 10, this function changes the scale of our acceptance function, in order to punish solutions that fall outside this interval). We treated this set of constraints as a set of different ranges of values. Inside them, the most optimal (with the smallest score) value will be the midpoint of that range. These ranges are a translation on what is defined by user's or expert's opinion.

For simplicity, we have omitted those parameters that act as modifiers, but, for completeness, we develop them here:

Some parameters in $\mathcal{K}$ o $\mathcal{U}$, do not generate modifications directly in $x \in \mathcal{I}$, they will affect other parameters as $g'_k : g(f_k(x))$, $g'_u : g(f_u(x))$ for $x \in \mathcal{H}$. An example of these parameters in the nutrition application would be a reduction on the limits of the daily cholesterol consumption due to the existence of a pathology.

In the discussion below, it is assumed that these parameters are applied in the functions and that, by construction, they have the form described above.

Thanks to the fitness function we can summarize the optimization problem describe in this subsection as follows:

$$\begin{cases} Min_{x \in \mathcal{H}} \| \mathcal{F}(x) = W_{\mathcal{V}_{\mathcal{U}}} f_{\mathcal{U}}(x) + W_{\mathcal{V}_{\mathcal{K}}} f_{\mathcal{K}}(x) \| \\ \text{where } \mathcal{H} \subset \check{\mathcal{I}} \end{cases} \tag{5}$$

2) OTHER FUNCTIONS ASSOCIATED TO THE GENERATION PROCESS

The rest of the associated functions that make up the evolutionary algorithm are often associated with different characteristics of the generation process. Thus, the classical crossover, mutation and selection functions are determined according to our objective concerning the variation of the solutions, the avoidance of local minima, and the structure of the elements of $\check{\mathcal{I}}$. In our case they will be defined as follows:

- **Selection function:** The selection functions are based on the fitness function. These functions select those objects from $\check{\mathcal{I}}$ with a higher score to be the candidates to expand their genome in the next generation. In situations where we are interested in a large variety, such as those presented in the paper, we have opted for a probabilistic function based on subsets of n elements. In our case, this function is able to produce high diversity. At the same time, it is capable of generating solutions with huge potential in the next steps (which can be obtained later with crossover and mutations).
- **Crossover function:** In our applications, the objects that make up the chromosome are directly related to the structure of the elements that are recommended. Thus, during the crossover, elements of $\mathcal{I}$ that are in a similar position relative to each other (and relative to each of their structures) that are part of $\check{\mathcal{I}}$ can be swapped. The frequency and arrangement in which they do so will depend on what features we are looking for. In our work, we have opted for a classical crossover that splits two solutions into two different sections and exchanges one part with the other.
- **Mutation function:** Mutation functions alter elements of $\mathcal{I}$ within the structure. They are therefore an interesting parameter to adjust in terms of their form and action. These functions can act on concrete or general subsets of the final structure. For example, in a nutritional recommendation system, the structure has a series of subsets related to each meal and filled with elements of our base dataset $\mathcal{I}$. These elements also have a quantity associated with them. The mutation, in this case, could be done (with a given probability) on a block of food, half of the structure, a particular meal, or a particular size.

C. SOLUTION REFINEMENT

The above procedure has allowed us to create a set of items adjusted to the selected parameters from the user and other additional sources of information. Furthermore, this set of solutions has a selected structure and it items preserve the relationships imposed during the generation process.

This set is also highly diverse. Its variability is not intrinsic to the model and is produced by the genetic algorithm. Parameters such us the stopping condition, the number of generations, the initialization of the population, the mutation rate or the types of mutation, can affect and generate different combinations. These features can also prevent the convergence of the algorithm to its global optimum, offering local sub-optimalities that can be refined in later phases and are sufficiently different from the previous ones.

We also make special emphasis on the initialization of the algorithm, as it can offer another customizable parameter and relegates much of the weight to the initial dataset. One of the downsides of these approaches is that it forces us to have a sufficiently well-annotated and diverse initial dataset or at least varied in those characteristics where we want to have huge variability.

At this point, thanks to all the previous conditions, we have a set of solutions that satisfy a series of imposed restrictions (to a greater or lesser extent, but always between the marked limits). This allows us to obtain a space of solutions or a subset of it, on which we can apply more specific algorithms of **RS**, but without depending on many constraints.

In the nutritional case, the solutions obtained so far meet the main characteristics associated with the healthiness of the model and the user's objectives, as these directly affect the final amount of nutrients. We can now focus on selecting from these alternatives, the one that best aligns with their preferences and tastes.

This improvement, using the same measures, will be done in three steps as described in Figure 2. First, we will have an evaluation phase where we analyze those elements that conform to our solutions. Through them, we will be able to know if any of the elements of $\mathcal{I}$ are marked by the user as non-preferential (or close to a non-preferential item). Notice that this may happen in this phase, since we are talking about preferences that may not be fulfilled for the sake of a recommendation adapted to the user's needs and constraints.

Once we have analyzed those elements of our structure that have room for improvement, we will look for elements in our database that are similar in the desired characteristics ($\mathcal{V}(\mathcal{U}, \mathcal{K})$), but that are not marked as non-preferred by the user (either because they are marked as liked or because they lack a negative evaluation).

We make as many changes as possible (or up to a specific changing rate) without altering our initial structure and move on to the last evaluation.

Finally, we select one of the generated and refined recommendations. We compute the scores associated with the user's preferences (at the conceptual levels we have chosen) to give a weighted average of the final score of the menu. An application example can be found in Figure 4.

D. CONTENT BASED SYSTEM

For both the evaluation and refinement steps, a content-based system is the natural step to take. This is due to several characteristics:

- First, the objects we are recommending already have a high degree of conceptualization. They are composite objects with a lot of data and relationships between them, so an approach based on exploiting precisely that information is very relevant.
- The content-based system is robust to cold start, which may be the case of several items in our database.
- Finally, we must not forget that the starting dataset can be very diverse (or the system has diversity conditions when it is generated), which means that, if we generate the initial population on which to apply the second phase, we will find unique objects, which would cause a user/item matrix to be very sparse.

These characteristics, while relevant, can be addressed with other types of systems, or even with hybrid systems. On the other hand, in order to use other methods such as collaborative approaches, we would need another source of information in our system (i.e. a user database and their interactions). We discuss this type of improvements in Section V.

1) SIMILARITY MEASURES

This section is highly variable depending on the area of application. However, as we have already mentioned, due to the large amount of information we have on the products, several similarity measures associated with different aspects can be established.

Each element that makes up our final items has a set of characteristics, denoted as $i_1, \ldots, i_p$. These characteristics can range from specific values to multivalued or Boolean labels (possesses a quality or does not possess a quality).

This fact allows us to transform these elements into vectors of a multidimensional space on which to apply different metrics. These metrics can be evaluated from the user's own profile, or by collecting those elements that the user has rated positively. Which one we apply will depend on our objectives and the nature of our data.

In particular, given that we are working with items created from the combination of another set, the characteristics we use can be a very high dimensional space. Therefore, the similarity metrics used in natural language processing, adapted to this kind of similar nature, are applicable in our approach.

Moreover, as we have been pointing out throughout the text, the presence of bi-valued or Boolean variables may also be common. This is why if we assume that k is the number of features of the elements of $\mathcal{I}$ appearing in the sub-set of $\mathcal{H}$ selected for the recommendation process, we highlight two similarity measures:

- For those variables that encode information of a different nature, operations such as weighted cosine similarity provide a metric that works optimally with large amounts of data even though the data are sparse:

$$\begin{cases} Sim_{Cos}(x, y) = \dfrac{\sum_{i_j} x_{i_j} w_{i_j} y_{i_j}}{\sqrt{\sum_{i_j} w_{i_j} x_{i_j}^2}\sqrt{\sum_{i_j} w_{i_j} y_{i_j}^2}} \\ i_j \text{ define the j-th parameter of x or y in } \mathcal{I} \end{cases} \quad (6)$$

  where the value w_{i_j} of those selected can represent different motivations (explained later in one of the applications).
- For those scores that are evaluated with a characteristic array composed of bi-valued variables, we can also use Jaccard's Similarity, especially useful if we seek to design a metric where we detect items whose composition is very similar (whether it is a desirable quality or not). We could transform some parameters into a binary set or use a generalized version of the Jaccard index for real valued vectors:

$$\begin{cases} Sim_{Jacc}(x, y) = J(x, y) = \dfrac{\sum_i \min\left(\mathrm{x}_i,\ \mathrm{y}_i\right)}{\sum_i \max\left(\mathrm{x}_i,\ \mathrm{y}_i\right)} \\ i_j \text{ define the j-th parameter of x or y in } \mathcal{I} \end{cases} \quad (7)$$

Once again, we highlight that other measures of similarity or distance are easily usable, and this will depend on the

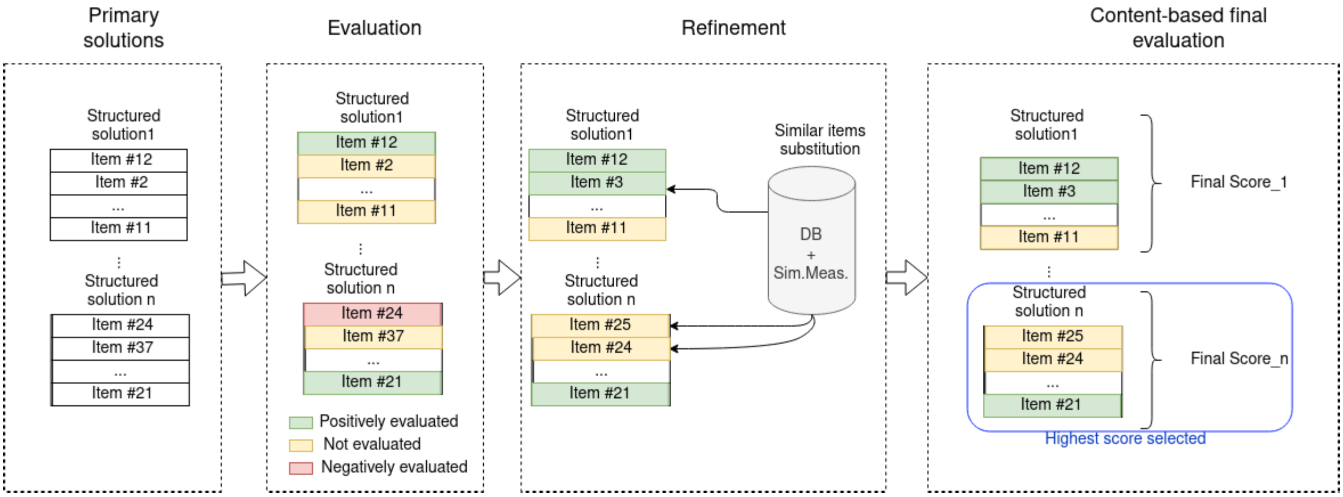

FIGURE 2. Solution refinement process.

nature of our data and objectives whether we use one or the other. Lastly this evaluation can be taken into account with the fitness evaluation from the first part as another important weighted factor. A final score, weighted sum of all the score used, is set as the final threshold to surpass.

IV. APPLICATIONS

In this Section we offer a complete walkthrought of a possible application of the theoretical recommender system develop on this topic. After this example, will briefly discuss other possible scenarios.

The field of personalized nutrition, which is experiencing great growth [20], [21], and we believe can benefit greatly from our approach.

A. DIETARY LONG-TERM ADVICE

Recommending a long-term diet is a highly complex but critical important problem. This is mainly due to some differences between this environment and similar ones that only focus on single and simpler recommendations.

An ideal long-term food recommendation system must be flexible enough to deal with the trade-off between personalized food preference/interest and nutritional/health requirements. We must take into account food preferences since it is unrealistic to think that subjects will drastically change their diet or tastes based on our suggestions. However, pathologies, allergies or other medical conditions that directly affect the subject's health (such as intestinal microbiota [22] or genomics [23]) must be taken into account.

Furthermore, we need to correctly understand what role the intrinsic characteristics of food play, and how they affect the predisposition of a user to choose a dish. That is to say, we may choose a quite different dish to eat for a midday dessert or a main dish for dinner.

Finally, is essential to understand how physical activity, pathology and user are related, and how they impact diet choices. Whether our objective is to change from one physical situation to another (weight loss) or to palliate any deficiency produced by a pathology (anemia), the system must be able to offer different recommendations that deal with the different objectives and states of users.

Earlier works on these topics mostly focus on dealing with user preferences when recommending (as for example in a restaurant). Then, works where health plays a leading role appears [24]. Along them, another relevant corpus on how to extract information and ontology's from recipes, textual and visual data references [25] evolve and they become another central part of these food recommendation systems.

Finally, in recent years, more recommender systems try to deal with users preferences and nutritional values, given the importance of a healthy nutrition in our lives [26]. Then the objective shift on how to deal with multiple constrain and preferences. Works like [27] combine several parameter while classifying ingredients. More recently [28] focus on certain health issues to recommend food for CKD patients, but in return they focus on fulfill only specific parameters for them. Evolutionary algorithms has also been used with good results. Works like [13] and [29] began to explore the possibility of using evolutionary algorithms to produces bundles or sets to be recommended (and extra source of recommendation with physical exercises).

However, our approach specifically focus on recipes, which give a context to the ingredients. On this behalf recent approaches has already use a multiobjetive optimization approach [30], which again state that evolutionary algorithm are useful for the recommendation of food. However, while this system learn a recipe pattern, most plans and nutritional intervention follows a more standard approach. Our system is more focused in this type of scenarios, as we have full control on the relations that we obtain when creating the initial population of menus. Moreover, this also improve the time performance of the algorithm. At the same time we can use the system to explore which set con parameters can not produce feasible combinations, that can offer the experts a starting point for completing the recipe database or let the user know if their pattern can be problematic to fullfill (for example if a user is physically active but eat small isolated portions of food).

Therefore, the nutritional application of our recommendation system is an application based on evolutionary algorithms, which are used for the creation and refinement of daily menus capable of incorporating constraints and rules at different levels of restriction. These rules can be specified at the precision level of different nutrients and are robust against user choices.

A graphic summary of this application can be found on Figure 3.

1) SOURCES OF DATA

Personalized nutrition problems usually have a set of different data sources. In our case we are going to follow the notation described in Section III, classifying them into three categories:

- $\mathcal{I}$: The space of items will be created based on the [31] and [32] databases that have data about recipes and their nutritional values. Most of the time, the recommendation processes use ingredients to give nutritional advice [20]. But we do not understand food in our daily lives like this. For that reason we define our primary sub-item space as a set of different recipes. Recipes also have extra information about its seasonality or cooking methods that can be used in the recommendation. Thus, recipes in $\mathcal{I}$ will form the menu that will be recommended to users.
- $\mathcal{U}$: The user space stores all the users of the application with the characteristics that have a direct impact on the system. The user's preferences as well as physical and health status are part of these parameters.
  Other parameters such as the usual quantities of certain recipes consumed by the user, the structure of his menu or their fitness goals must be taken into account during the generation process.
  Finally, in this category, we will also find parameters that define the user at a biometric level, such as BMI, weight or height, necessary to calculate the energy expenditure throughout the day.
- $\mathcal{K}$: Expert knowledge will be the most diverse set of parameters in this problem. Within this set, we will have information about the choices the user can make regarding some of its parameters (i.e. the different menu structures supported). These parameters also encode the nutritional values associated with the healthy amount of each nutrient. These amounts will be affected according to the user's condition, physical activity or pathologies. Any additional information from nutrition experts will be stored in this set.

2) CREATION OF THE ITEM SPACE AND FIRST GENERATION

Once we have defined the initial items, we proceed with the creation of the final item space. This first part is already an element of the recommendation process, since some user characteristics $\mathcal{U}$, as well as the constraints of $\mathcal{K}$, are taken into account in this creation.

From $\mathcal{U}$ we collect all the parameters that build a biomedical profile of our user (with data such as BMI, weight, height or the amount of body fat).

On the other hand, with respect to $\mathcal{K}$ we collect two types of constraints. Those that directly relate the healthy amount of nutrients the user should have, according to their data. And those that intervene in the menu generation: type of cuisine, seasonality, possible menu structures.

It is particularly relevant to point out how the structure of the menu is selected. In this case the final structure of the recommendation depends both on the user and on the parameters of $\mathcal{K}$. In this model we make use of expert knowledge and propose common food patterns. This generates a first set of structures that we can use. However, in order to make the system flexible, it is up to the user to choose which of these structures is the most appropriate for their daily diet.

All the parameters selected are included in the final item space considered by our system. This space is made up of daily menus composed of dishes, which follow a structure and relationships set by the user and the expert knowledge.

Finally, the first module, configured as stated in Section III, produces a sub-set of the processed item space. This generation process tries to find which generated items are closer to our objective, defined by the fitness evaluation. The size of the initial sub-set of the item database and the mutation functions of the genetic algorithm allow us to ensure some variability in the results.

3) SOLUTION REFINEMENT AND RECOMMENDATION

At this point we have a structure filled with sub-items from $\mathcal{I}$ that verify the main constraints selected for the first part of our recommendation process.

From here we move on to the second phase where we refine these solutions according to a more classical approach, based on (mainly) user preferences. A graphic summary can be found in Figure 4.

This procedure consists of two parts. On the one hand, the personalization of the generated menus (corresponding to Evaluation and Refinement in Figure 4) and on the other hand, the choice of the menus that best fit the user's preferences (corresponding to Content-based final evaluation in Figure 4).

Firstly, given the dishes that are associated with our menu (i.e. the elements of $\mathcal{I}$ that appear in $\mathcal{H}$), we can choose those that are negatively valued by the user. These items must be replaced by a similar dish. For them, we will use euclidean similarity, looking for those dishes that are very similar nutritionally speaking, but that are marked as indifferent or liked

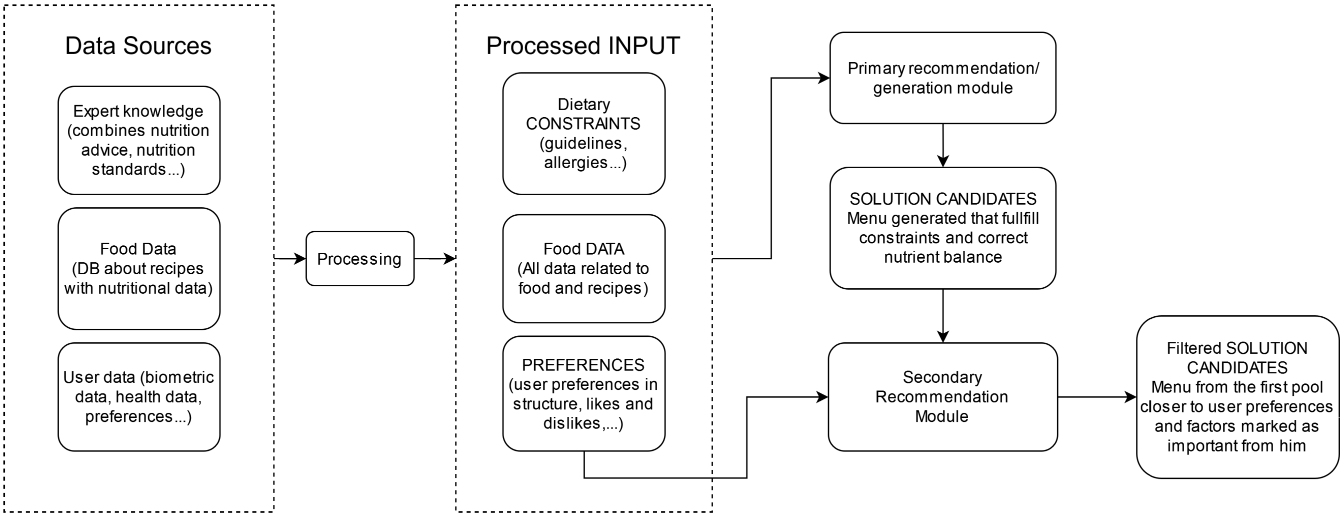
Data Sources
Expert knowledge (combines nutrition advice, nutrition standards...)
Food Data (DB about recipes with nutritional data)
User data (biometric data, health data, preferences...)
Processing
Processed INPUT
Dietary CONSTRAINTS (guidelines, allergies...)
Food DATA (All data related to food and recipes)
PREFERENCES (user preferences in structure, likes and dislikes,...)
Primary recommendation/ generation module
SOLUTION CANDIDATES Menu generated that fullfill constraints and correct nutrient balance
Secondary Recommendation Module
Filtered SOLUTION CANDIDATES Menu from the first pool closer to user preferences and factors marked as important from him

FIGURE 3. Flowchart describing the recommendation process for the nutritional case.

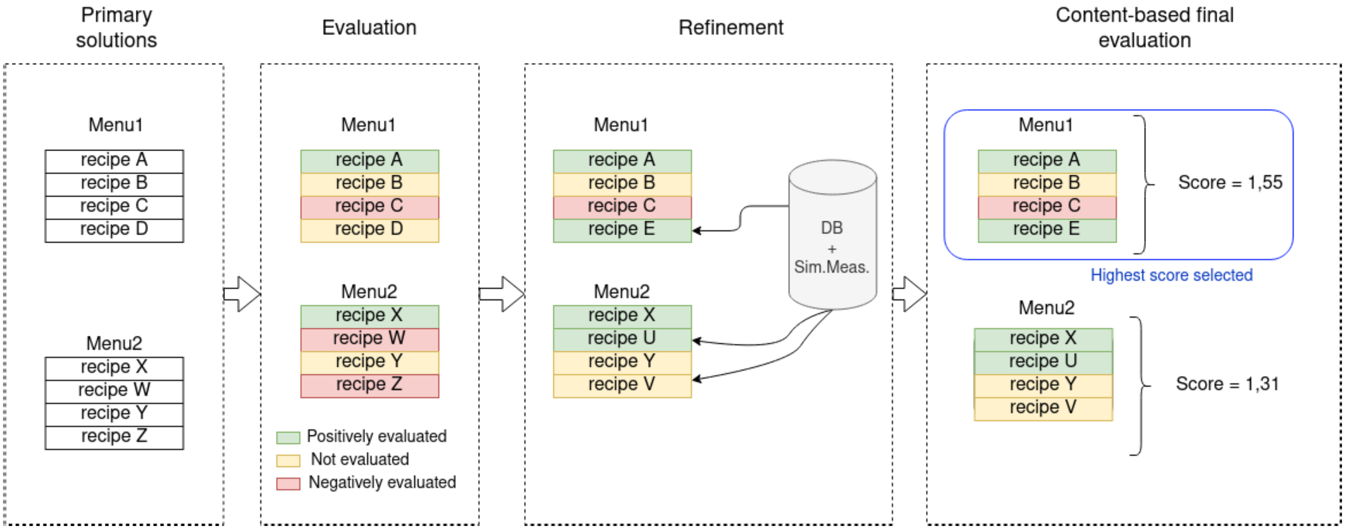

FIGURE 4. Solution refinement phase for the nutritional application.

by the user rather than disliked. It is possible that we are unable to find a realistic improvement in this structure. If this is the case, we still keep the solution for the last phase.

Finally, from the set of secondary recommendations, we will evaluate which is closer to the user's tastes (using the similarities define in section III-D1). In this case we recover some information from $\mathcal{I}$ to develop three types of evaluated and weighted scores:

1) Dish category: Those menus that have a higher frequency of labels common to the dishes valued positively by the user: meats, salads, fruit, etc. We will compare the set of tags from a menu with the set of tags from the user and derive a distance from it.
2) Fitness functions: Fitness of the menus evaluated. We will use a cosine similarity metric to compare set of normalize nutrients.

The weighting of these characteristics not only reflects the relative importance of each of the similarities compared to the rest, but can also be related to other types of information. In the nutritional case, similarity could be scored more positively if it is found in a main dish within the menu structure, as there is a correlation between the overall quantity and significance of the main dish compared to smaller side dishes or desserts. Another option could be to assign extra value to the fitness function evaluations.

4) EXPERIMENTAL OFFLINE EVALUATION

For the first evaluation of the model we followed a offline test based on different user profiles. Those where selected in the Stance4Health project. The Stance4Health project (Smart Technologies for personalized Nutrition and Consumer Engagement) (S4H) is a project funded by European

TABLE 2. Example menu for user.

Course	Dish	Name
Breakfast	Main	Skimmed milk with wholemeal bread
Mid-morning snack	Main	Skimmed yogurt and toast with jam
Lunch	First	Vegetable lasagna
Lunch	Main	Artichokes with Iberian Serrano Ham
Lunch	Dessert	Watermelon
Dinner	Main	Brown rice with leek
Dinner	Dessert	Flat peach

Union's Horizon 2020 research and innovation program. Our main focus was to show that this approach can achieve reasonable results for a realistic user in a nutrition intervention. The ability to produce healthy menus has already been validated by nutritionist in [33] for different micronutrients. However In this section we will show that the model can obtain healthy menus from different configuration, showing its robustness to different patterns, sizes or restrictions.

Moreover we will offer novel insights on the secondary recommendation module. For this task we choose a standard healthy male human, with a normal IMC, moderately active which result in a General Metabolic Rate of 2000 kcal a day. For this user we have design the following scenarios, along with and explanation on which situation they can be useful:

- S0:User does not take anything in the morning and single recipes for lunch and dinner. Its portions are standard portions.(Adapt to user behaviour)
- S1:User make 3 meals a day
- S2: User make 4 meals a day
- S3; User make 5 meals a day (Calorie surplus)
- s4: User demand fish as the main ingredient of lunch and dinner (ability to adapt to certain type of dish, this is necessary to provide not daily, but weekly recommendation, where we can incorporate Mediterranean patterns in the diet)
- s5: User demand meat as the main ingredient of lunch and dinner (ability to adapt to certain type of dish, this is necessary to provide not daily, but weekly recommendation, where we can incorporate Mediterranean patterns in the diet)
- s6: User is allergic to milk
- s7: User demand higher portion of dishes (have a good appetite, vegan diets)
- s8: User demand the smallest quantity possible. (Troubles eating, inability to cook)

An example of the output menu can be seen in figure 2.

We divided the tests in different stages, one for the nutritional adjustments of the menu (which is the main focus of the first part) And one for the second part, centered in the content based recommendation system.

For the evaluation of the genetic evolution process we run a set of 30 different menus modifying the algorithmic-centered parameters, those are: number of generations, fitness functions, initial population, type of crossover, probability of crossovers. Based on the first set of tests, we chose the parameters described in table 3 for the next part of the evaluation. For the threshold we base our results on the scores of the nutritional validation [33].

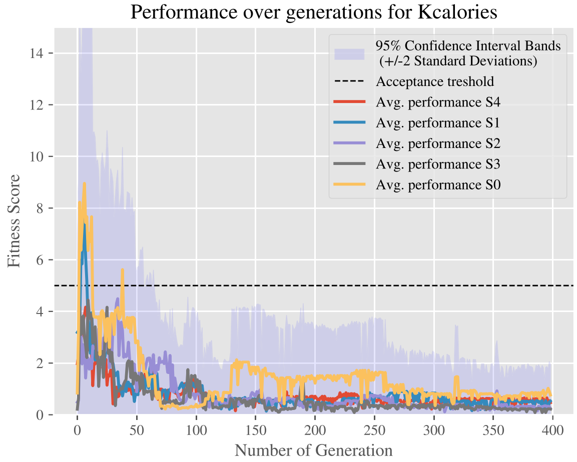

FIGURE 5. GA convergence scoring only the Kilocalories levels in the scenarios S0,S1,S2,S3,S4.

With this set of values we then run the primary recommendation system throught 50 different days in each scenario. In this process we start using the most basic fitness function declare in III taking only the data from the kcalories. We re run this experiments using a more advance definition of the fitness function taking into account kcalories and macronutrient levels. Finally, we produce the last batch of experiments using the most advanced fitness function. As we can see in figures 5-10 along 200 steps in the generation process, all fitness functions stay below their acceptance threshold, but the variability of data force us to reach between 250 and 300 generations to stay below the accepted level in all situations. These evaluations use all the three possible fitness functions built with the nutrient levels. It is worth noting that we are using only the last one for evaluating the population, where all the parameters are encoded as the formula presented in (5). The two upper plots represent how these less complex fitness functions built as Eq. (5) but with less nutritional parameters decay as well as the most complex one is used.

On the second part of the recommendation we proceed to change the menus according to the preferences in food types the user give us. Every recipe in the dataset has two set of characteristics: one with a general pyramid-type of food and other ontological related to that one related to the type of dish (over 35 different categories). We use that information and the user preferences to calculate the distance between the user preferences in terms of food types and the user disfavors in term of the menu food types. For that we use a Jaccard similarity type as boolean vector of length 35 (has this food type, do not have it) and evaluate to both types of preferences.

TABLE 3. Summary of the values used in the genetic algorithm part.

GA parameter	N^{O} of Generations	Initial population size	treshold	Tourtnament type	Probability of mutation	Crossover
Value	300	15	300	Elitism 3 random selection	0.5	Half menu exchange

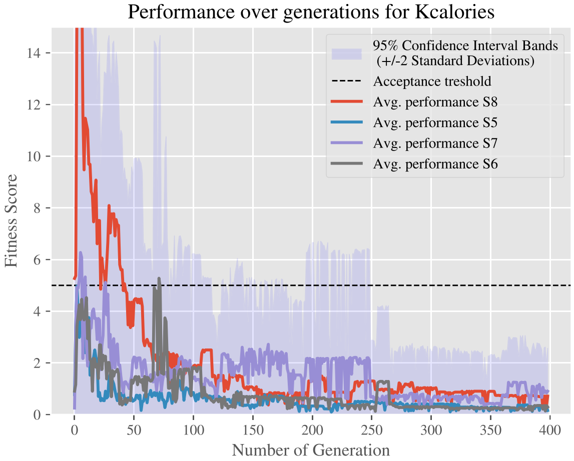

FIGURE 6. GA convergence scoring only the Kilocalories levels in the scenarios S5,S6,S7,S8.

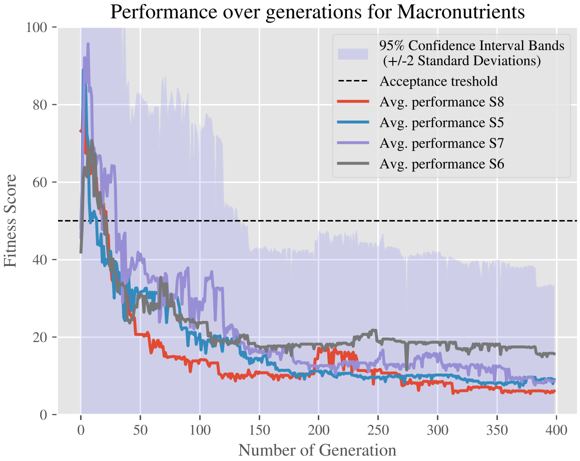

FIGURE 8. GA convergence scoring Kilocalories and macronutrient levels in the scenarios S5,S6,S7,S8.

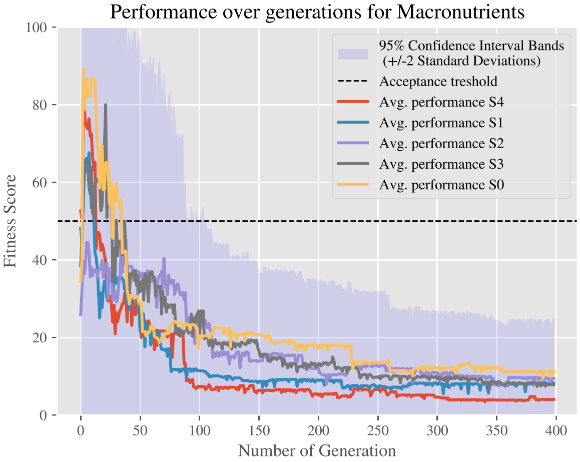

FIGURE 7. GA convergence scoring Kilocalories and macronutrient levels in the scenarios S0,S1,S2,S3,S4.

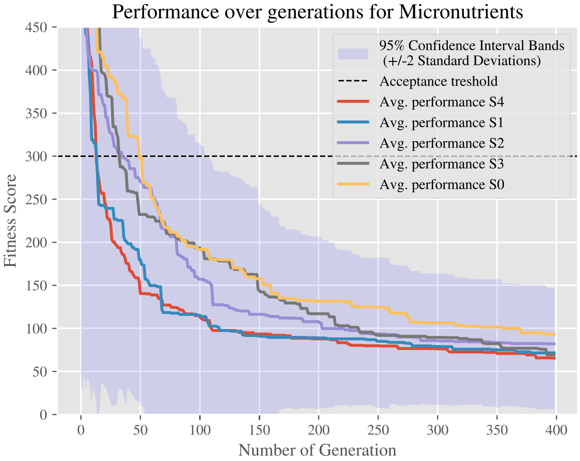

FIGURE 9. GA convergence scoring the most complex function (Kcal, macro nutrients and micronutrients) in the scenarios S0,S1,S2,S3,S4.

Ideally the distance of like categories will decrease while the distance to the disliked types will increase.

However we still have to take into account the nutritional score. Throughout the preference metrics we will add up the fitness score function to penalize those menu changes that diverge too much from the recommendation (due to the absence of similar dishes that match the preferences).

We perform this second part with a random assignation of like and dislike dishes, and two more oriented one: only like meat and no vegetables and the opposite: plant-based diet with little to no-meat. The variability of the distance from the liked-by-user pattern and disliked-by-user pattern is shown in Table 4. Moreover a kcal and all nutrient oscillations are also shown.

For this section we force at least one change in the menu for all the menus created before. However two or more changes were discarded if they disrupt the healthy level of nutrients. On overall we get an decrease in the like metrics while a increase in the distance form the dislike options. Along this changes, most of the nutritional level stayed close to the original ones and the objective one.

V. DISCUSSIONS AND RESEARCH LINES

There are several remarks to be made about this type of system. We want to highlight some of them along with the research directions that would improve our results.

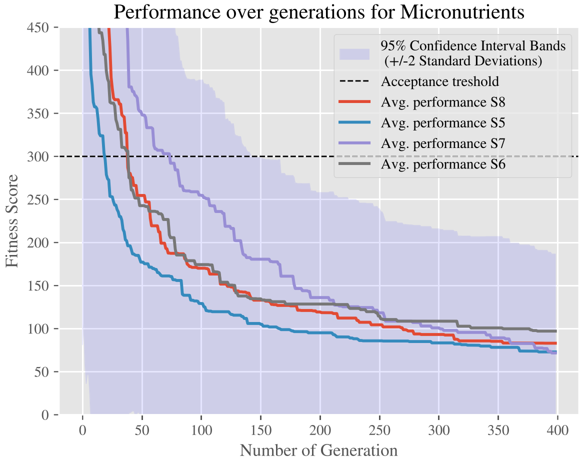

FIGURE 10. GA convergence scoring the most complex function (Kcal, macro nutrients and micronutrients) for the scenarios S5,S6,S7,S8.

A. DATA SOURCES AND EXPECTED OUTCOME

Having a sufficiently large dataset can prove highly advantageous for establishing an initial set of recipes, denoted as $\mathcal{I}$, that is substantial enough to offer a significant range of choices to users. However, further research is still needed to explore the impact of various factors on the recommendation process. Specifically, we posit that investigating the optimal number of recipes required, the number of user recipes that can be incorporated into the original dataset, and the types of recipes to be added to the initial dataset are pivotal considerations. Such investigations are crucial not only for narrowing down the search space but also for presenting users with options that closely align with their preferences.

Furthermore, constructing complete dataset with recipes and multiple nutrients information would facilitate the comprehensive evaluation of menus from a broader perspective. By considering multiple nutrients, such a system would enable the assessment of whether a set of recipes can yield healthy values across various menu configurations. This type of system would also prove valuable in nutritional intervention scenarios like Stance4Health, where users are more inclined to adopt complete recipes as opposed to isolated ingredients. From the expert standpoint, this system would empower the generation of comprehensive menus based on user requirements, preferences, and nutrient levels.

B. EVALUATION AND USER ADHERENCE

One of the main concerns in recommender system is the evaluation process. Finding an optimal way to measure how correct are our recommendations, and how users may perceive them, is a crucial step. In addition how we translate this feedback into the recommendation process is also a relevant task. For example, upgrading our system to give more weight to users' preferences may please the user, but at the same time it could affect to the different constraints and rules that are significantly knowledge-based as dietary patterns or nutrient levels.

TABLE 4. Value oscillation after the secondary module. An conservative strategy was adopted and only small variation of the nutrients where allowed.

Randomly recipes based diet		
Nutrient	**Unit**	**Avg. Variation**
Dislike distance	-	+0.04
like distance	-	0.0
kcal	kcal	-13.761230220317449
prot	gr	-0.4787886729870081
carb	gr	-2.3930042006395977
fat	gr	5.5323632772495435
fiber	gr	0.6314800815290088
vitamin c	mgr	23.309959776641275
iron	mgr	1.2869436552125038
calcium	mgr	221.9211939960594
vitamin a	mcg	605.2216795035298
vitamin b6	mg	0.12096667466822812
vitamin d	mg	0.8831782318351888
vitamin e	mg	-0.04191276760525309
vitamin k	mcg	-26.967997170903345
phosphorus	mg	248.23147039009345
iodine	mcg	-13.199232041648225
Meat based diet		
Nutrient	**Unit**	**Avg. Variation**
Dislike distance	-	+0.03
like distance	-	-0.02
kcal	kcal	-20.76980993867777
prot	gr	-1.2853054948674079
carb	gr	-11.349871501578805
fat	gr	2.15362956783667
fiber	gr	-0.3597185261571551
vitamin c	mg	4.818489899065986
iron	mg	-0.5324358173383796
calcium	mg	24.34613141594754
vitamin a	mcg	136.43323729369698
vitamin b6	mg	-0.0005443298859073575
vitamin d	mg	1.3463118304824875
vitamin e	mg	0.5697441805741829
vitamin k	mcg	11.294386024089308
phosphorus	mg	-28.59306844771512
iodine	mcg	2.04775629037603
Plant based diet		
Nutrient	**Unit**	**Avg. Variation**
Dislike distance	-	+0.020
like distance	-	-0.023
kcal	kcal	-27.643320500722336
prot	gr	0.6367802296466921
carb	gr	-13.33946494350169
fat	gr	2.285943875284525
fiber	gr	0.7682935597238089
vitamin c	mg	37.6217961349622
iron	mg	0.580691558837975
calcium	mg	15.47815102283235
vitamin a	mcg	-60.78130001410057
vitamin b6	mg	-0.02462815755091788
vitamin d	mg	0.5734166322332718
vitamin e	mg	2.1383934085201814
vitamin k	mcg	-3.7408984417080737
phosphorus	mg	-0.9323320301525607
iodine	mcg	9.284131809164709

Moreover, bundle recommendation and sequence recommendation produce new specific challenges. When evaluating a set of objects, we may have a single item in the bundle that could transform our decent recommendation into a bad one. Or perhaps a slightly positive collection of products is better rated by the user than a set where one item is rated very positively, but the rest are unknown. Evaluating these factors can make a big difference to the final result.

Some studies already suggest the possibility of evaluating how constraints can affect the optimization of recommendations [6], [28], [34]. In our case, we believe that our system has a great potential to study how different rules are correlated, if they can produce acceptable outputs or even if we need certain types of simple items in our database that enables the later creation of complex items that fulfill our necessities.

This evaluation is not only important for the performance of the system. In addition, the complexity of the recommendations makes it difficult to compare the system with other similar systems. This problem itself also opens another line of interesting research that we would like to explore in near future.

Finally, it is worth noting that some scenarios presented in Section IV are meant to be used continuously in time. Specifically in these situations, a positive evaluation will be key to maximizing user's adherence to the system, which may be needed for long-lasting changes.

C. INTERPRETABILITY OF THE SYSTEM

In Section III we defined $\mathcal{E}(\mathcal{U}, \mathcal{K})$ and $\mathcal{V}(\mathcal{U}, \mathcal{K})$. These two sets of parameters encode the two categories that should be differentiated when attempting interpretability strategies:

- The fixed constraint that will mostly obey very strict rules and admit only small variations
- Those characteristics that function as parameters or rules that admit modification by the user.

Within these two categories, a significant portion of the information can be classified into the user-based and item-based explanations, as proposed in Zhang et al. [35]. These categories, along with feature-based explanations, have the potential to provide substantial information to the user, aiding their decision-making process.

Certain application areas may involve subjective factors for the user. Hence, it becomes crucial to evaluate which aspects the user deems most important in order to enhance acceptance and adherence to our recommendations.

Identifying the pertinent information to display, considering factors such as the user's prior knowledge, and determining the optimal way to present it, pose significant and intriguing challenges within our system. The complexity arises from the extensive conceptual load and intricate interrelationships that must be considered during the recommendation process. However, addressing this challenge will be a pivotal aspect in improving the interpretability of the system for the user.

D. ETHICS AND FAIRNESS

As we have been pointing out throughout the survey, two of the main (defining) characteristics of **RS** in complex scenarios are complexity and high impact. Complexity because of the large number of relationships and structures involved, and impact, because of the ability of these systems to significantly affect people's lives.

The characteristics of these systems, and the areas where they are applied, make an excellent case for delving into the repercussions of **RS** on our society. And therefore, how they affect us and how they should affect us. With the contributions to Ethics and Fairness becoming more and more relevant [36], and looking at the technologies described in this study, the analysis of the ethics and impact of RS will gradually become more and more important.

This study will be fundamental to ensure that the synergies between these systems and our lives are made with the greatest possible guarantees for the user, but also for all the stakeholders involved [37]. It is also worth noting that these complex situations often lead to ethical considerations. A clear example of this will be how to offer a recommendation that may contain aspects the user could (in some degree) dislike, due to its benefits in the long term.

E. HYBRIDIZATION

In addition to content-based filtering we could use other types of filtering during the recommender process. However, in most cases we would have to deal with additional information that is added to the system.

Specifically collaborative filtering can be a good starting point to hybridize our system and get the benefits of it, when we have a sufficiently large mass of users. For this we would have to decide which model to use, both in the collaborative section and in the hybridization process. It should be noted that some advantages of the collaborative approach, such as serendipity, are to some extent covered by the randomness of the first phase of our recommendation.

Other advantages of this hybridization also depend on the problem we want to solve. Areas such as nutrition have more important constraints to meet and do not need a greater capacity for "surprise" to satisfy the user. On the contrary, when we work with problems where we apply less significance to the items and more to the structure, this type of hybridization may offer more desirable features. This would be the case for podcasts.

Overall, we can hybridize our system with collaborative filtering and other recommendation approaches depending on our objectives, such as those based on sessions or reinforcement, which allow us to capture useful contextual information for the user. These add-ons will probably allow us to apply our approach to more problems, but further research on them is required.

VI. CONCLUSION

Given the remarkable success and widespread adoption of Recommendation Systems in our society, their application to complex scenarios is expected to continue expanding. This growth can be attributed, in part, to the incorporation of various sources of conceptual, relational, and contextual information into these systems, along with advancements in algorithms capable of effectively integrating such information.

These advancements in understanding and incorporating information empower us to develop systems that can assist us in making decisions of high complexity in our daily lives. In particular, we have proposed a system that combines two computational processes, each operating on information of different nature: preferences and constraints. This approach enables us to construct recommended items from simpler components, thereby introducing a structured framework with relationships and order among sub-items. Furthermore, the secondary recommendation module allows us to dynamically adjust the recommendations to align with the user's preferences or requirements.

The manner in which our system incorporates and processes information is particularly well-suited for tackling complex problems such as nutritional recommendation. However, with appropriate modifications, it can be adapted to similar situations where simple items, user preferences, and domain experts define the constraints and preferences that need to be fulfilled.

ACKNOWLEDGMENT

This work is part of the doctoral thesis of Bartolome Ortiz-Viso conducted within the context of the ''Program of Information and Communication technologies'' at the University of Granada.

REFERENCES

[1] F. Ricci, L. Rokach, and B. Shapira, ''Recommender systems: Introduction and challenges,'' in *Recommender Systems Handbook*. New York, NY, USA: Springer, 2015, pp. 1–34.

[2] R. Burke, ''Knowledge-based recommender systems,'' in *Encyclopedia of Library and Information Systems*, vol. 69. New York, NY, USA: Taylor & Francis, 2000, pp. 175–186.

[3] S. Zhang, L. Yao, A. Sun, and Y. Tay, ''Deep learning based recommender system: A survey and new perspectives,'' *ACM Comput. Surv.*, vol. 52, no. 1, pp. 1–38, Jan. 2020.

[4] T. N. T. Tran, M. Atas, A. Felfernig, and M. Stettinger, ''An overview of recommender systems in the healthy food domain,'' *J. Intell. Inf. Syst.*, vol. 50, no. 3, pp. 501–526, Jun. 2018.

[5] Q. Shambour, ''A deep learning based algorithm for multi-criteria recommender systems,'' *Knowl.-Based Syst.*, vol. 211, Jan. 2021, Art. no. 106545.

[6] Y. Zheng and D. Wang, ''Multi-criteria ranking: Next generation of multi-criteria recommendation framework,'' *IEEE Access*, vol. 10, pp. 90715–90725, 2022.

[7] P. Chavan, B. Thoms, and J. Isaacs, ''A recommender system for healthy food choices: Building a hybrid model for recipe recommendations using big data sets,'' in *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, 2021, p. 3774.

[8] D. A. Rohani, A. Q. Lopategui, N. Tuxen, M. Faurholt-Jepsen, L. V. Kessing, and J. E. Bardram, ''MUBS: A personalized recommender system for behavioral activation in mental health,'' in *Proc. CHI Conf. Human Factors Comput. Syst.*, Apr. 2020, pp. 1–13.

[9] L. Esmaeili, S. Mardani, S. A. H. Golpayegani, and Z. Z. Madar, ''A novel tourism recommender system in the context of social commerce,'' *Expert Syst. Appl.*, vol. 149, Jul. 2020, Art. no. 113301.

[10] J. Su, Y. Guan, Y. Li, W. Chen, H. Lv, and Y. Yan, ''Do recommender systems function in the health domain: A system review,'' 2020, *arXiv:2007.13058*.

[11] V. S. Vairale and S. Shukla, ''Recommendation of food items for thyroid patients using content-based KNN method,'' in *Data Science and Security*. Singapore: Springer, 2020, pp. 71–77.

[12] L. R. Ferretto, E. A. Bellei, D. Biduski, L. C. P. Bin, M. M. Moro, C. R. Cervi, and A. C. B. De Marchi, ''A physical activity recommender system for patients with arterial hypertension,'' *IEEE Access*, vol. 8, pp. 61656–61664, 2020.

[13] H. Alcaraz-Herrera, J. Cartlidge, Z. Toumpakari, M. Western, and I. Palomares, ''EvoRecSys: Evolutionary framework for health and well-being recommender systems,'' *User Model. User-Adapted Interact.*, vol. 32, pp. 883–921, Jan. 2022.

[14] A. Mustaqeem, S. M. Anwar, and M. Majid, ''A modular cluster based collaborative recommender system for cardiac patients,'' *Artif. Intell. Med.*, vol. 102, Jan. 2020, Art. no. 101761.

[15] P. Usip, F. Osang, and S. Konyeha, ''An ontology-driven fashion recommender system for occasion-specific apparels,'' *J., Adv. Math. Comput. Sci.*, vol. 8, no. 1, pp. 67–76, 2020.

[16] R. Yin, K. Li, J. Lu, and G. Zhang, ''Enhancing fashion recommendation with visual compatibility relationship,'' in *Proc. World Wide Web Conf.*, May 2019, pp. 3434–3440.

[17] W. Kang, C. Fang, Z. Wang, and J. McAuley, ''Visually-aware fashion recommendation and design with generative image models,'' in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2017, pp. 207–216.

[18] R. Logesh, V. Subramaniyaswamy, and V. Vijayakumar, ''A personalised travel recommender system utilising social network profile and accurate GPS data,'' *Electron. Government, Int. J.*, vol. 14, no. 1, pp. 90–113, 2018.

[19] S. Jiang, X. Qian, T. Mei, and Y. Fu, ''Personalized travel sequence recommendation on multi-source big social media,'' *IEEE Trans. Big Data*, vol. 2, no. 1, pp. 43–56, Mar. 2016.

[20] T. Theodoridis, V. Solachidis, K. Dimitropoulos, L. Gymnopoulos, and P. Daras, ''A survey on AI nutrition recommender systems,'' in *Proc. 12th ACM Int. Conf. Pervasive Technol. Related Assistive Environ. (PETRA)*, Rhodes, Greece, 2019, pp. 540–546.

[21] S. Norouzi, A. K. Ghalibaf, S. Sistani, V. Banazadeh, F. Keykhaei, P. Zareishargh, F. Amiri, M. Nematy, and K. Etminani, ''A mobile application for managing diabetic patients' nutrition: A food recommender system,'' *Arch. Iranian Med.*, vol. 21, no. 10, pp. 466–472, 2018.

[22] M. D. Russo, P. Russo, J. Á. Rufián-Henares, D. Hinojosa-Nogueira, S. Pérez-Burillo, S. P. de la Cueva, S. Rohn, A. Fatouros, K. Douros, V. González-Vigil, D. Epstein, M. P. Francino, A. Siani, and F. Lauria, ''The Stance4Health project: Evaluating a smart personalised nutrition service for gut microbiota modulation in normal- and overweight adults and children with obesity, gluten-related disorders or allergy/intolerance to cow's milk,'' *Foods*, vol. 11, no. 10, p. 1480, May 2022.

[23] V. A. Mullins, W. Bresette, L. Johnstone, B. Hallmark, and F. H. Chilton, ''Genomics in personalized nutrition: Can you 'eat for your genes,''' *Nutrients*, vol. 12, no. 10, p. 3118, 2020.

[24] M. Ge, F. Ricci, and D. Massimo, ''Health-aware food recommender system,'' in *Proc. 9th ACM Conf. Recommender Syst.*, New York, NY, USA, Sep. 2015, pp. 333–334.

[25] S. Fang, Z. Shao, R. Mao, C. Fu, E. J. Delp, F. Zhu, D. A. Kerr, and C. J. Boushey, ''Single-view food portion estimation: Learning image-to-energy mappings using generative adversarial networks,'' in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 251–255.

[26] C. Iwendi, S. Khan, J. H. Anajemba, A. K. Bashir, and F. Noor, ''Realizing an efficient IoMT-assisted patient diet recommendation system through machine learning model,'' *IEEE Access*, vol. 8, pp. 28462–28474, 2020.

[27] R. Y. Toledo, A. A. Alzahrani, and L. Martínez, ''A food recommender system considering nutritional information and user preferences,'' *IEEE Access*, vol. 7, pp. 96695–96711, 2019.

[28] R. Shandilya, S. Sharma, and J. Wong, ''MATURE-food: Food recommender system for mandatory feature choices a system for enabling digital health,'' *Int. J. Inf. Manage. Data Insights*, vol. 2, no. 2, Nov. 2022, Art. no. 100090.

[29] H. Alcaraz-Herrera and I. Palomares, ''Evolutionary approach for 'healthy bundle' wellbeing recommendations,'' in *Proc. HealthRecSys RecSys*, 2019, pp. 18–23.

[30] J. Zhang, M. Li, W. Liu, S. Lauria, and X. Liu, ''Many-objective optimization meets recommendation systems: A food recommendation scenario,'' *Neurocomputing*, vol. 503, pp. 109–117, Sep. 2022.

[31] *I-Diet Food Composition Database, Updated From Original Version of G*, Gestion De Salud Nutricion Sl, Martín Peña FCD 1, Spain, 2019.

[32] D. Hinojosa-Nogueira, S. Pérez-Burillo, B. Navajas-Porras, B. Ortiz-Viso, S. P. de la Cueva, F. Lauria, A. Fatouros, K. N. Priftis, V. González-Vigil, and J. Á. Rufián-Henares, ''Development of an unified food composition database for the European project 'Stance4Health,''' *Nutrients*, vol. 13, no. 12, p. 4206, Nov. 2021.

[33] D. Hinojosa-Nogueira, B. Ortiz-Viso, B. Navajas-Porras, S. Pérez-Burillo, V. González-Vigil, S. P. de la Cueva, and J. Á. Rufián-Henares, ''Stance4Health nutritional APP: A path to personalized smart nutrition,'' *Nutrients*, vol. 15, no. 2, p. 276, Jan. 2023.

[34] M. Uta, A. Felfernig, and D. Helic, ''Constraint-aware recommendation of complex items,'' in *Proc. Joint KaRS ComplexRec Workshop (CEUR-WS)*, 2021, pp. 38–44.

[35] Y. Zhang and X. Chen, ''Explainable recommendation: A survey and new perspectives,'' 2018, *arXiv:1804.11192*.

[36] S. Lo Piano, ''Ethical principles in machine learning and artificial intelligence: Cases from the field and possible ways forward,'' *Hum. Social Sci. Commun.*, vol. 7, no. 1, pp. 1–7, Jun. 2020.

[37] S. Milano, M. Taddeo, and L. Floridi, ''Recommender systems and their ethical challenges,'' *AI Soc.*, vol. 35, no. 4, pp. 957–967, Dec. 2020.

BARTOLOME ORTIZ-VISO received the B.S. degree in mathematics from Granada University, Granada, Spain, in 2017, and the joint M.S. degree in mathematics and physics from the University of Granada and Castilla La Mancha, Spain, in 2018. He is currently pursuing the Ph.D. degree in computer science with the Center for Communication Research, Granada University. He is also a Predoctoral Researcher with the European Research Project (H2020) Stance4Health, developing nutritional recommendation systems focusing on overall health and microbiome data. His research interest includes the development of computational techniques that improve human situations. Specifically those that include applied mathematics and computer science with applications in biology and medicine.

ANDREA MORALES-GARZÓN received the B.Sc. degree in computer science and the M.Sc. degrees in computer engineering and data science from the University of Granada, Spain, in 2018, 2020, and 2022, respectively. She is currently pursuing the Ph.D. degree in computer science under the supervision of Prof. Maria J. Martín-Bautista. She is a member and a Predoctoral Fellow of the Intelligent Databases and Information Systems IDBIS Research Group, University of Granada. Her research interests include machine learning (ML) and natural language processing (NLP), with a main focus on the development of intelligent systems for domain adaptation (DA) applied to several scenarios.

MARIA J. MARTIN-BAUTISTA received the Ph.D. degree. She has been a Full Professor with the Department of Computer Science and Artificial Intelligence, University of Granada, Spain, since 2018. She has supervised several Ph.D. thesis and published more than 100 papers in high impact international journals and conferences. She has participated in more than 20 research and development projects and has supervised several research technology transfers with companies. Her current research interests include recommender systems, intelligent information systems, big data analytics in data, text and web mining, knowledge representation, and uncertainty. She is a member of the Intelligent Data Bases and Information Systems (IDBIS) research group. Furthermore, she has served as a program committee member for several international conferences.

MARIA-AMPARO VILA was an Associate Professor and an Analyst-Programmer with the Computer Centre, University of Granada, where she has been a Professor in computer science and artificial intelligence, since 1992. She has developed her research and teaching primarily in the area of databases and intelligent information systems and its main lines of research are the treatment of imprecision in information systems using fuzzy logic, knowledge discovery in databases using techniques soft computing-based, ubiquitous computing, and knowledge mobilization. She has been an advisor of 27 doctoral theses, she has been or is responsible for more than ten research projects and she has published numerous research papers, which highlights more than 100 papers in SCI journals. She has been responsible for the research group of the Andalusian ICT-113 approximate reasoning and artificial intelligence), from 1994 to 1997, and the Head of the Group TIC174 (Databases and Intelligent Information Systems) since its inception in 2000 until today. From the point of view of university management, she has been the Vice-Director of Organization with the School of Computer Science for three years (1994–1997) and the Director of the Department of Computer Science and Artificial Intelligence, University of Granada, from 1997 to 2004. She has also developed related tasks and quality assessment processes.