Knowledge-Based Systems 251 (2022) 109216

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Exploring post-hoc agnostic models for explainable cooking recipe recommendations

Raciel Yera [a], Ahmad A. Alzahrani [b], Luis Martínez [a,*]

[a] Computer Science Department, University of Jaén, Jaén, Spain
[b] Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia

ARTICLE INFO

Article history:
Received 26 October 2021
Received in revised form 2 June 2022
Accepted 3 June 2022
Available online 17 June 2022

Keywords:
Explainable recommendation
Cooking recipes
Post-hoc explanation
Trustworthiness

ABSTRACT

The need of increasing trustworthiness and transparency in artificial intelligence (AI)-based systems that adhere ethical principles of respect for human autonomy, prevention of harm, fairness, and explainability; has boosting the development of systems that incorporate such issues as a key component. Recommender systems (RSs) are included in such AI-based systems, because they use intelligent algorithms for providing the most suitable items to active users according to other users' preferences. The RSs success is based on how much customers trust on the system, therefore recommendation explainability has become a crucial dimension for RSs adoption in real-world scenarios. Among the different successful applications of RS, it is remarkable the recent and exponential importance of recommendations for health and wellness areas. Hence, this paper aims at exploring, adapting and applying explanations for nutrition/recipes recommendations, that not only explain why the recommendation is enjoyable but also, it is aware of how healthy is the recommendation. Among the different methodologies to explain recommendations, this paper is focused on post-hoc explainability approaches and its adaptation, application and evaluation for nutrition/recipes recommendation. Eventually, it is included a comprehensive experimental study for characterizing the strengths and weaknesses of such explainability approaches in the recipe recommendation context.

1. Introduction

Recommender systems (RSs) are AI-based systems focused on providing users with items that best fit their preferences and needs, in an overloaded search space of possible options [1,2]. RSs have been widely used in a wide range of domains, such as e-commerce, e-learning, e-health, e-business, wellness, and so on [3–7].

Two main paradigms have driven the development of RSs. At first, content-based recommendation [8] focuses on representing user and item profiles through the same feature space closely related to item characteristics. On the other hand, collaborative filtering-based recommendation [9,10] focuses on the preference of similar users, to generate suggestions for the active one. Recently, successful recommendation approaches have been focused on the incorporation into this scenario, of computational intelligence techniques such as matrix factorization or deep learning-based for predicting the users' unknown preferences and generating the recommendation lists [2,11]. However, such techniques have an important drawback related to their black-box behavior, generating a lack of transparency that affects their trustworthiness.

Furthermore, within these successful recommendation approaches the major goal of RSs research has been the improvement of the accuracy of the recommendation algorithms [12]. However, recently several authors have pointed out that beyond the accuracy improvement, it is very important the capacity to explain the recommendation results [13]. Such facts have been raised among others by the European Union Ethics Guidelines for a Trustworthy Artificial Intelligence, boosting the development and use of artificial intelligence systems in a way that adheres to the ethical principles of respect for human autonomy, including explainability.[1]

Nowadays, explainable recommendation is a key dimension in highly-risk domains such as e-health and e-business in order to facilitate the final decision made by the active user in a transparent and trustworthy way [14]. Furthermore, it is becoming a must in other low risky but very popular domains such as e-commerce and e-learning [14].

* Corresponding author.
E-mail addresses: ryera@ujaen.es (R. Yera), aalzahrani8@kau.edu.sa (A.A. Alzahrani), martin@ujaen.es (L. Martínez).

[1] https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai

https://doi.org/10.1016/j.knosys.2022.109216

Specifically, explainable recommendations have been mainly focused on the development of intrinsic models, centered on explaining the recommendation method; or post-hoc models, which are focused on explaining the recommendation results [14]. Most of research was focused on the use of intrinsic models, in which appears an important shortcoming such as the strong dependency of the associated recommendation approach, being coupled with such an approach, the final recommendation effectiveness. On the other hand, post-hoc explanation approaches have recently received an increasing attention, considering they are completely independent of the main recommendation algorithm [14–16], and therefore they have the potential to be used across a greater diversity of domains in RSs.

Formally, post-hoc methods are centered on explaining the recommendation results, focused on coupling any main recommendation method with a white-box explainable method that facilitates the explanation of the main approach [14]. Key research works in this direction have been developed by Peake and Wang [15], presenting an early post-hoc approach for individual RS, that extracts explanations from latent factor-based recommendation systems by training rule mining models on the output of a matrix factorization black-box model. Furthermore, Nóbrega and Marinho [17] introduced the generation of locally interpretable model-agnostic explanations (LIME-RS) which is focused on discovering the top-n item features that better explain the individual recommendations delivered by the factorization machine method. LIME-RS was later extended by Chanson et al. [18], being focused on improving the sampling process around the recommendation instance to explain, to finally learn a proper explanation model. In a different direction, SHapley Additive exPlanation values (SHAP) have been raised as post-hoc RS explanation model [19], being focused on computing the average of the marginal contributions of each feature value to the model prediction across all permutations. Recently, Shmaryahu et al. [16] have pointed out the use of simple-to-explain content-based and collaborative filtering-based explanation approaches, for explaining complex recommendation methods such as matrix factorization-based collaborative filtering. Here, it must be pointed out that post-hoc explanations do not precisely reflect the computation used by the underlying recommendation model, but they commonly present rationale, plausible, and valuable information for the user [20].

The need of improving trustworthiness and transparency about recommendations in general and health and wellness recommendation in particular, in addition to the features of post-hoc approaches, drives this paper at exploring the use and adaptation of post-hoc explanation approaches in cooking recipe recommendations [21–24]. Recipe recommendation has become an important RS domain, gaining popularity in the last few years together with other domains such as movies, books, or travel packages [4]. Particularly, most of current proposals for recipe recommendation are based on complex models that lack of transparency and do not provide any add value regarding trustworthiness [6,23]. Furthermore, recipe recommendation is a domain in which it is necessary to take into account both the suitability of suggested items (enjoyability) and their appropriateness from the nutritional viewpoint [6,23]. Hence, it is necessary the contextualization and adaptation of post-hoc recommendation explanation approaches to recipe recommendation. Consequently, this work is then focused on exploring, adapting and applying post-hoc recommendation explanation models in this domain, to characterize their explanation ability in relation to the preference-aware and health-aware viewpoints.

Specifically, our goal is to explain why the recommended cooking recipes are enjoyable, as well as controlling how the incorporation of the nutrition-aware criteria affects such explanation capability.

Our paper is then chasing the following objectives:

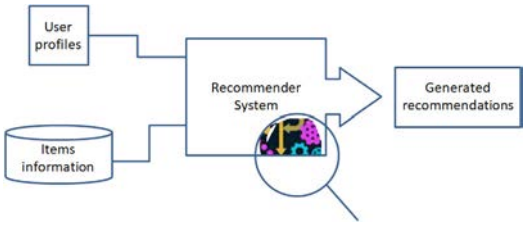


Fig. 1. Basic scheme of a recommendation approach.

- The contextualization and adaptation of post-hoc recommendation explanation approaches to the recipe recommendation domain.
- The incorporation of nutritional knowledge domain into the explainable recipe recommendation approaches.
- The evaluation of the contextualized explanation approaches with and without incorporating the nutritional knowledge domain, by using measures specifically focused on explanation capabilities.

The paper is structured as follows. Section 2 incorporates the necessary background for the proposal understanding, including recommender systems, the recipe recommendation domain, and explainable recommendations. Section 3 contextualizes and adapts several recommendation explanation approaches to the cooking recipe domain. Furthermore, Section 4 develops an extensive evaluation protocol for evaluating and comparing the proposed explanation approaches, which also explores the effect of considering the nutritional information in the final recommendation generation, in relation to the explanation capability. Section 5 concludes the paper, pointing out future works.

2. Background

This section briefly presents the necessary background for this research proposal. It includes basics on recommender systems, the recipe recommendation domain, and explaining recommendations.

2.1. Recommender systems

Recommender systems are tools focused on providing users with the information that best fits their preferences and needs in a search space overloaded with possible options [25] (Fig. 1).

Beyond the current diversity of proposals for the development of recommendation technologies, they have been mainly based on two paradigms: content-based recommendation, or collaborative filtering-based recommendation [1]. Such paradigms are detailed briefly in the next subsections.

2.1.1. Content-based recommendation

Content-based recommender systems [8] assume that items are characterized by a set of attributes, that are used as the key for recommendation generation. Such systems are based on the fact that items with similar attributes, receive similar preferences by the same user.

Content-based recommendation usually comprises four steps: (1) Item profiling, (2) User profiling, (3) User–item utility calculation, and (4) Recommendation [26]. Being the items represented by a set of attributes that characterize them, user profiles are calculated by using the profiles linked to the items preferred by the associated user, being based on the same item feature space. Subsequently, the user–item matching calculation is usually represented by information retrieval-related functions such

as cosine or Jaccard measures [1,11]. These matching values are finally used for retrieving the items with higher utility as the top n recommendation list for each user.

Recently, Pérez-Almaguer et al. [26] extend individual content-based recommender systems to be used in the group recommendation problem, by proposing several approaches adapted to such scenario. In this way, they propose content-based group recommendation approaches based on recommendation aggregation and ranking, on recommendation aggregation and user–item matching values, and on the aggregation of the user profiles.

2.1.2. Collaborative filtering-based recommendation

On the other hand, collaborative filtering is focused on using the preferences of similar users, as a starting point for generating recommendation to the active one [9]. Collaborative filtering is usually divided into memory-based and model-based methods [1]. Specifically, memory-based collaborative filtering [27] implements this paradigm in a direct way, by using a similarity function for identifying the top k users who are more similar to the active one. In a second stage, it aggregates the preferences of such neighbor users for calculating the preferences of the active user and generates then the top n recommendation list.

In contrast, model-based collaborative filtering does not develop a direct calculation of the similarity values between users [28]. Instead, it usually builds a user–item model that comprises all the preferences information in a reduced feature space, and allows a direct calculation of the user rating over certain items. The building of the compact user–item model, usually removes some data disturbance, leading therefore to an improvement in the recommendation accuracy compared to memory-based approaches [28].

A key approach in model-based collaborative filtering is the matrix factorization approach, popularized by Koren et al. in the context of the Netflix Prize [29]. The matrix factorization approach maps users and items into a joint latent factor space of dimensionality f, where user–item interactions are modeled as inner products in that space. Each item i is then associated with a vector $q_i \in R^f$, and each user u is associated with a vector $p_u \in R^f$. For item i, q_i measures the extent to which the item possesses those latent factors. Similarly, for a given user u, the elements of p_u measure the interest of u in items i, represented by the same factor space. The resulting dot product $(q_i)^T p_u$, represents the interaction between user u and item i, which can be interpreted as the user's interest in the item's characteristics. In this context, this is regarded as the approximates user u's rating of item i, r_{ui}, leading to the estimate $\tilde{r}_{ui} = q_i^T p_u$.

In this framework, the goal is to compute the factor vectors $q_i, p_u \in R^f$, which is usually done by minimizing the regularized squared error on the set of known ratings:

$$min_{q^*,p^*} \sum_{(u,i)\in\kappa} (r_{ui} - q_i^T p_u)^2 + \lambda(\|q_i\|^2 + \|p_u\|^2) \quad (1)$$

where κ is the set of the (u, i) pairs for which r_{ui} is known.

This basic framework has been extended into more complex model-based collaborative filtering methods, such as methods based on matrix factorization incorporating temporal dynamics [30], non-negative matrix factorization [31], or matrix factorization-based list-wise learning [32]. Furthermore, recently neural collaborative filtering methods [33] have been also developed as a successful paradigm for building effective model-based recommendation systems.

Beyond its success in achieving a high recommendation accuracy, an important shortcoming of model-based approaches is their lack of transparency [1], because they are black-box models where the final users are not aware of their working principle (see Fig. 1), affecting in this way the trust on the delivered recommendations.

In this way, post-hoc explanation models [14] are currently used to complement model-based recommendation methods in order to provide such methods with explainability capabilities. Section 2.3 presents some notions on post-hoc explanations models. Furthermore, the main goal of this work is exploring the performance of post-hoc recommendation explanation models, coupled with model-based collaborative filtering, in the particular recipe recommendation domain.

2.2. The recipe recommendation domain

Recipe recommendation is a domain that has historically received comparatively less attention in RS than other areas related to leisure and entertainment [34,35]. However, in the last few years it has been identified as a domain with great importance considering it incorporates a relevant health-aware component [35].

In this way, Trattner and Elsweiler [34] developed a survey that includes the more relevant research results associated to this recommendation problem. Here, such authors reported more than 15 research works focused on the use of traditional recommendation algorithms for recipe recommendation. It includes the use of memory-based methods [23,36], matrix factorization-based methods [37], learning to rank approaches [38], and so on. Furthermore, beyond the necessary user preferences, in several cases they work over heterogeneous data sources that include tags [37], image embedding [38], or text sentiments [39].

Table 1 presents previous works on recipe recommendation, identified as relevant by the research literature. Specifically, we have considered works focused on recipe recommendation that have been highlighted at the survey by Trattner and Elsweiler [34], as well as research linked to recent relevant venues on food recommendations, such as ACM UMAP and ACM RecSys main conference and workshops.

The analysis behind Table 1 suggests that recent works such as Ludwig et al. [40], Chen et al. [41] and Pecune et al. [42], have been focused on the use of black-box models for recipe recommendation, evidencing a better performance in relation to other recommendation techniques. Furthermore, it is also relevant the use of ingredients and nutritional information of recipes [40,41,44,46,47], as a data source for recommendation generation. Other group of works, such as Yang et al. [38] and Elsweiler et al. [44] also use images for characterizing recipes, being their use limited to some specific scenarios. It is also worthy to mention that some recent works are not exactly focused on recipe recommendation. Instead, they work on different problems, such as recipe completion [41], or healthier recipe replacement [44]. Finally, it is also necessary to note that there are only few works incorporating dietary constraints in recipe recommendation (e.g. Chen et al. [41], Yang et al. [38], Bianchini et al. [46]), and some of them are not directly focused on the recipe recommendation problem.

Previous analysis shows clearly the necessity of developing more transparent recipe recommendation approaches, that can explain the performance associate to recent successful black-box models in this direction. This goal can be currently reached, by taking into account the availability of relevant recipe data like ingredients and nutritional information, as discussed in this section. Finally, it is necessary to incorporate nutritional constraints as component of the recipe recommendations.

Therefore our proposal is focused at this direction, by discussing model-agnostic post-hoc explanation methods in the recipe recommendation domain.

Table 1
Relevant works on recipe recommendation. ALS—Alternating Least Square. BPR—Bayesian Personalized Ranking. MF—Matrix Factorization.LMF—Logistic Matrix Factorization. CB—Content based. CF—Collaborative filtering. LDA—Latent Dirichlet Allocation. WRMF—Weighted Matrix Factorization.AR—Association Rules. NB—Naive Bayes.SVD—Singular Value Decomposition.SLIM—Sparse Lineal Methods. RF—Random Forest.

Authors	Algorithms	Used information	Dietary constraint
Ludwig et al. [40]	MF, nutritional validity by post-filtering	User's nutritional requirements	No
Chen et al. [41]	Deep learning focused on recipe completion	Ingredients	Yes
Pecune et al. [42]	ALS,BPR, LMF	FSA health score	No
Khan et al. [43]	Ensemble topic modeling	Food features from text (e.g. ingredients, category, context)	No
Elsweiler et al. [44]	RF,NB, for predicting suitability of a healthier recipe replacement	Image, ingredients	No
Trattner and Elsweiler [23]	LDA,WRMF,AR SLIM,BPR,Mostpop User–ItemKNN	WHO-FSA Health score	No
Cheng, Rokicki and Herder [45]	BPR, Mostpop	City size	No
Yang et al. [38]	Learning to rank	Image embeddings	Yes
Bianchini et al. [46]	Content-based recommendation	Ingredients, recipe type, country	Yes
Ge et al. [37]	MF, CB	Tags	No
Trevisiol et al. [39]	UserKNN,CB	Text sentiment	No
Harvey et al. [47]	CB, CF, Logistic Reg., SVD-Hybrid	Ingredients	No
Forbes and Zhu [48]	MF	Ingredients	No
Freyne and Berkovsky [36]	UserKNN, CB, Hybrid	Ingredients	No

2.3. Building explainable recommendations

The necessity of explainable recommendation in real-world scenarios comes from the users' low understanding on why systems make decisions or exhibit certain behaviors [13,14]. Such inscrutability can hamper users' trust in the system, especially in contexts where the consequences are significant such as e-business, e-health, or software engineering, and lead to the rejection of the systems. An explanation for the delivered recommendations is then likely to make the information more useful to the user/group and has a stronger influence on their actions.

In a recent survey, Zhang and Chen [14] have revised more than 170 works on recommendation explainability, pointing out a two-dimensional classification according to: (i) the computational model used for recommendation (neighbor-based, matrix factorization, deep learning, topic modeling, graph-based, knowledge-based, rule mining, and post-hoc), and (ii) the information/style of the generated explanations (relevant user/item, user/item features, textual sentences, social explanation etc.).

Overall, explainable recommendations have been mainly focused on the development of: (1) model-intrinsic explanation approaches, and (2) model-agnostic explanation approaches, also called post-hoc approaches. The next subsection will be focused on briefly discussing such approaches.

2.3.1. Model-intrinsic explanation approaches

The model-intrinsic explanation approach develops interpretable models whose decision mechanism is transparent and thus, they can naturally provide explanations for the model's decision.

Intrinsic explanation contexts are usually coupled to some specific recommendation techniques, considering the nature of such explanation type. Therefore, most of works in this direction are focused on extending factorization-based, topic modeling, graph-based, deep learning, or knowledge-based approaches, to devise interpretable models that increase transparency, leading then to the explainability of the recommendation results [14].

Focused on this goal, Zhang et al. [49] proposed Explicit Factor Models, for recommending products that perform well on the user's favorite features, aligning then each latent dimension of matrix factorization with an explicit feature. Such alignment makes trackable the factorization and the prediction procedures, allowing the generation of explicit explanations. Wang et al. [50] also proposed a tree-enhanced embedding model for explainable recommendation to combine the generalization ability of embedding-based models and the explainability of tree-based models. In a different direction, several researchers have recently leveraged deep learning and representation learning for explainable recommendations [51]. Furthermore, knowledge graphs have also helped to explain black-box recommendation models. Herein, Zhang et al. [14] proposed an end-to-end joint learning framework to combine the advantages of embedding-based recommendation models and path-based recommendation models, for explaining the generated suggestions.

A recognized limitation of model-intrinsic explanation approaches is that they are necessary coupled with the specific associated model. This lack of flexibility makes difficult their generalization to more complex recommendation scenarios such as the cooking recipe domain. For such reasons, at this stage this explanation paradigm was not selected to be studied in the current research work.

2.3.2. Model-agnostic explanation approaches

The model-agnostic explanation approach, also called the post-hoc approach, allows to the decision mechanism to be a black box. It then develops a model that generates explanations after a recommendation has been made. Therefore, they are centered on explaining the recommendation results and not the recommendation process [14]. In post-hoc explanations in recommender

systems, recommendations and explanations are obtained from different models. An explanation model (independent from the recommendation mechanism) provides explanations for the main recommendation model after the recommendations have been provided (thus "post-hoc").

On the other hand, in typical applications the recommendation mechanism is composed by several components and therefore may be too complex to explain. In such complex contexts, such as recipe recommendations, post-hoc explanation can become a suitable approach for explaining recommendations [14].

In this way, post-hoc explanation models are currently a growing research trend in explainable recommendation. Peake and Wang [15] presented a pioneer post-hoc approach for individual RS, that extracts explanations from latent factor recommendation systems by training rule mining models on the output of a matrix factorization black-box model.

Furthermore, Singh and Anand [52] focused on post-hoc explanation of learning to rank algorithms. Here the authors based the ranking explainability on an interpretable feature space, reached in a model-agnostic way. McInerney et al. [53] also developed a bandit approach to explainable recommendation. They assume that users would respond to explanations differently and dynamically, and based on such issue, develop an exploitation–exploration bandit-based approach to find the best explanation orderings for each user. Eventually, Cheng et al. [20] also proposed an explanation method named FIA (Fast Influence Analysis), which helps to understanding the prediction of trained latent factor models by tracing back to the training data with influence functions.

Focusing on a machine learning context, Ribeiro et al. [54] proposed the development of Local Interpretable Model-agnostic Explanation models (LIME), which adopts sparse linear models to approximate a black-box classifier around a sample. Such linear model can thus explain which sample features contributed in a high extent to its predicted label. Using the LIME framework, Nóbrega and Marinho [17] introduced the generation of locally interpretable model-agnostic explanations for recommender systems (LIME-RS) focused on discovering the top-n item features that better explain the individual recommendations delivered by a factorization machine method. Furthermore, Chanson et al. [18] introduced LIRE, an improved LIME framework for recommender systems in the sense that it introduces an efficient sampling around the recommendation instance to explain, to finally learn a proper local surrogate model.

Also, Shmaryahu et al. [16] provided post-hoc explanations for why a recommended item may be appropriate for the user, by using a set of simple, easily explainable recommendation algorithms supported by collaborative filtering and content-based recommendation.

Summarizing, although in the last few years several research works have been focused on post-hoc recommendation explanations, it is also necessary to mention that some of them were initially focused on other scenarios such as learning to rank or classification. The next section further details several of these approaches that will be used and adapted for our proposal of recommendation explanation in the recipe recommendation domain. Taking into account that it is a complex recommendation scenario, post-hoc approaches are considered more appropriate than intrinsic explanation approaches for recipe recommendation, as it has been previously mentioned.

3. Explainable recipe recommender systems based on post-hoc explanations

This section is focused on exploring in detail four post-hoc explanation approaches, and how to adapt them to be used in the cooking recipe domain.

The selected approaches are:

1. The recommendation supported by simple-but-effective post-hoc explanation approaches, due to its simplicity and intuitiveness [16] (Section 3.1).
2. The Global Explanation Mining for post-hoc interpretability for Recommender Systems (Section 3.2), taking as base the Peake and Wang's framework [15] which is one of the former approaches in post-hoc explanation, and based on rule mining which is a traditional tool for generating explanations.
3. The Local Explanation Mining for post-hoc interpretability for Recommender Systems (Section 3.3), as a particular approach based on a local rule mining process which has outperformed previous proposals based on this direction [15].
4. An improved locally interpretable model-agnostic explanation model, proposed by Chanson et al. [18] (Section 3.4), which is an improved version of LIME [17] contextualized to the recommendation scenario, being LIME a recently featured state-of-art model in AI explainability [55].

Table 2
Basic notation.

Notation	Meaning
u	User
c	Cooking recipe
c_{kcal}	Amount of kilo-calories of recipe c
$fats$	Fats of recipe c (in grams)
$carb$	Carbohydrates of recipe c (in grams)
$prot$	Proteins of recipe c (in grams)

These approaches have been identified as generalizable and well-established methodologies for supporting post-hoc recommendation explanations [14] in traditional recommendation scenarios such as Movies, TV Shows, etc. [15,16]. Therefore, they are an appropriate choice to be used as starting-point for generating explanations in the cooking recipe domain. In this way, we remark that for this study we do not consider other explanation approaches, such as SHAP [19], which are focused on providing explanation for a more holistic viewpoint, and less focused on justifying the personalized recommendations centered on each specific user [14].

In order to incorporate nutritional information of the recipes for accomplishing the dual goal related to provide both enjoyable and healthy recommendations, it will be considered as available some nutritional values associated to recipes. The basic used notation is represented at Table 2.

3.1. Recommendation supported by simple-but-effective post-hoc explanation approaches

This subsection adopts a general framework introduced by Shmaryahu et al. [16], to be used in the recipe recommendation domain. It is focused on using simple and transparent methods for explaining the output of complex recommendation models. In the context of the cooking recipe recommendation domain, the former framework is extended through the incorporation of a stage that introduces a nutritional value-aware re-ranking procedure of the recipes initially recommended by the black-box recommendation model (see Fig. 2). Furthermore, two specific simple-to-explain recommendation approaches are brought to be used in the current scenario, based on the available information of recipes, which are mainly represented by their list of ingredients.

In further detail, this approach works across several stages depicted at Fig. 2 and contextualized to the cooking recipe domain:

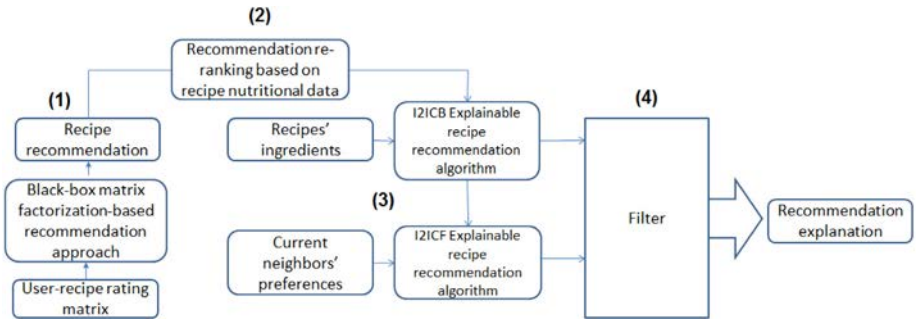


Fig. 2. Simpler methods-based recommendation explanations for recipe recommendation.

1. At first, a black-box main recommendation model receives as input the preferences data and retrieves appropriate recommendations (Step 1 in Fig. 2). The preference data is represented by a user-recipe rating matrix, that is processed by a black-box recommendation algorithm, to obtain a top n recipe recommendation list. In the current scenario it is used a matrix factorization-based recommendation algorithm previously detailed in Section 2.1.2 [29], which effectiveness has been proved across several domains (Eq. (2)).

$$min_{l^*,p^*} \sum_{(u,c)\in\kappa} (r_{uc} - l_c^T p_u)^2 + \lambda(\|l_c\|^2 + \|p_u\|^2) \quad (2)$$

   In Eq. (2), l_c represents the latent factors associated to the recipes, while p_u represents the interest of u toward such latent factors. Here the vectors l^*, p^* are learned through a gradient descendent method [29].
2. Furthermore, it is executed a re-ranking stage of the recipes at the top n recipe recommendation list (Step 2 in Fig. 2). This re-ranking is based on the nutritional score $nut(p)$ of the recipe (Eq. (6)), represented by how close it is to the ideal balance of macronutrients (e.g. proteins, carbohydrates and fats) in a food intake.
   To perform this analysis, two facts considered in previous works are taken into account [6]:
   (a) The daily energy intake should be composed of 50% of carbohydrates, 20% of proteins, and 30% of fats.
   (b) The energy values of carbohydrates, proteins, and fats, are represented through the following equivalences:

$$1g\ of\ proteins = 4kcal \quad (3)$$

$$1g\ of\ carbohydrates = 4kcal \quad (4)$$

$$1g\ of\ lipids = 9kcal \quad (5)$$

   Based on the amount of kilo-calories c_{kcal} of a recipe c, such facts allow to calculate the ideal *expected* amount of carbohydrates, proteins and fats that some food should have containing such kilo-calories. The deviation of the *actual* values of such macronutrients at the recipe, can be regarded as its nutritional score. A higher score, a less nutritional value of the recipe. Eq. (6) formalizes such value.

$$nut(p) = |4carb - 0.5c_{kcal}| + |4prot - 0.2c_{kcal}| + |9fats - 0.3c_{kcal}| \quad (6)$$

   After the re-ranking of the top n recipes according to their nutritional value, only the top p are retrieved and used as input for the further steps.
3. The delivered recommendations are then given as input for several simpler and explainable recommendation algorithms, which can also use additional information sources (e.g. item features, user demographic information, and so on) (Step 3 in Fig. 2). The explainable algorithm is used for generating a score for the initially recommended items, and if such score is sufficiently high, then the explainable algorithm considers its associate explanation as valid, and retrieves it as a possible output of the whole framework.
   Formerly, Shmaryahu et al. [16] specifically implemented six simple-to-explain algorithms. Each algorithm takes as input a user profile, and an item i that is recommended by a complex black-box model. Such six simply-to-explain algorithms are Popularity, Item–item content-based, User–item content-based, Item–item overview, Item–item collaborative filtering, and User–user collaborative filtering.
   Considering the expected sparsity of the cooking recipe recommendation domain, here it will be considered item–item recommendation algorithms which usually perform well in sparse scenarios [56]. Particularly, it will be considered the item–item content-based and item–item collaborative filtering explanation algorithms.
   (a) Item–item content-based (I2ICB): Considering the initial goal of explaining the recommendation of a recipe c, in this approach for each recipe d that the user has rated, it is computed a similarity value between d and c. The explanation is then generated through the features of such recipes more similar to the currently recommended recipe c, or through all the recipes which similarity is over a threshold δ:

$$sim(c,d) > \delta \rightarrow c\ is\ explainable\ through\ features\ of\ d \quad (7)$$

   Here Jaccard coefficient [57] is used for calculating similarities between recipes. In this particular scenario, the ingredients are considered as the features of the recipes. A sample of the provided explanation would be:
   Recipe **c** *is recommended, because you love* **d** *in the past, and both recipes have in common tomato and cheese as main ingredients.*
   (b) Item–item collaborative filtering (I2ICF): It is computed the item–item similarity score between each recipe d rated by the user, and the currently recommended recipe c. This score is represented as the ratio between the number of users who preferred both recipes, and the number that preferred at least one. The explanation is then focused on presenting items with a high Jaccard values in relation to the

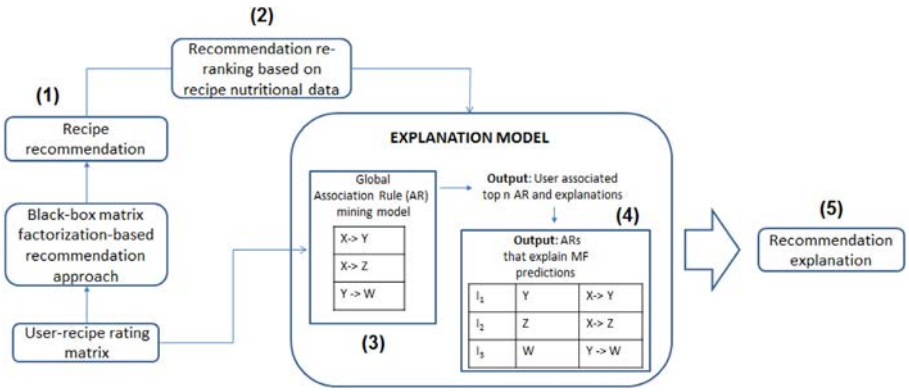


Fig. 3. Global Explanation Mining for post hoc interpretability in Recommender Systems.

current item, regarding a threshold δ:

$$sim(c, d) > \delta \rightarrow \textit{c is explainable through users preferring d} \quad (8)$$

A sample of the provided explanation would be:
Recipe **c** *is recommended, because in the past you preferred* **d**, *and many people that preferred* **d**, *also preferred* **c**.

4. Finally, the valid recommendations provided by all the explainable algorithms are ranked, finally shown the best scored explanation to the final user (Step 4 in Fig. 2). The authors point out that this final selection could be subjective, regarding each algorithm provides scores in a different way.

3.2. Global explanation mining: Post-hoc interpretability for recommender systems

This subsection contextualizes to the recipe recommendation domain one of the firstly presented post-hoc approaches for explaining black-box recommendation models. This method, proposed by Peake and Wang [15], is focused on the training of association rules on the output of a matrix factorization black-box model. Such association rules contribute to the explanation of the recipe recommendations generated by the main recommendation model, by extracting explanations that can be used to understand the model behavior.

Specifically, the association rules used in this method are obtained across the entire dataset, therefore the recommendations generated for any user are explained by taking as basis the same set of association rules. Based on this fact, the method is coined as global explanation mining.

This approach comprises the following steps, which are also represented in Fig. 3 over the same overall scheme linked to the simpler methods-based recommendation explanations (see Fig. 2). Such steps are also detailed at Algorithm 1:

1. Input data: The input data is a user–item matrix of observed interactions or preferences, as described by Koren et al. [29].
2. Train matrix factorization model and prediction: The traditional black-box matrix factorization training [29] is performed over the input data of the previous step. Subsequently, it is calculated the output prediction of the model, which is the completed user-recipe matrix with predicted ratings for all user-recipe pair (Step 1 in Fig. 3). To process this output for the final recommendation generation, ratings from the training data are filtered out. For the remaining items, the top n with the higher scores for each user, compose the recommendation list that is used in the subsequent steps (Line 1 at Algorithm 1).
3. Nutrition-aware re-ranking of the recommendation list: The recipes in the retrieved recommendation list are re-ranked according to the criteria presented at Eq. (6) (Step 2 in Fig. 3), in the previous Section 3.1. Afterwards, the top p recommendations of the re-ranked list are subsequently used in the following steps.
4. Train Association Rules (the interpretable model): Here the required set of transactions T [58] are generated by taking the predicted recipe ratings for each user from the unfiltered matrix factorization output (Line 5 at Algorithm 1). These transactions are used for building association rules, following the traditional approach formerly pointed out by Sarwar et al. [59] for recommendation scenarios (Line 6 at Algorithm 1) (see also Step 3 in Fig. 3).
5. Output association rules: The output of the previous stage is a list of all rules representing relationships between recipes in the matrix factorization predictions and their corresponding support, confidence, and lift measures. In order to contextualize such rules to the current recommendation scenario, for each user it is only filtered to a subset the rules where the antecedent X is in the user training data and the consequent Y is in the top p nutrition-aware recommendation list (Line 8 at Algorithm 1) (Step 4 in Fig. 3). The corresponding explanation can be then seen as the antecedents of the corresponding rules, and therefore they are both retrieved (Line 9) (Step 5 in Fig. 3).

Overall, this method can be considered as a particular item-to-item explanation style, having some common points with the I2ICB, and I2ICF explanation approaches, discussed in the previous Section 3.1.

3.3. Local explanation mining

Here, it is introduced a more sophisticated approach for explaining recipe recommendations, which is also based on the Explanation Mining framework [15].

This new approach, coined as Local Explanation Mining, assumes that the global association rules that are identified in the Global Explanation Mining approach (Section 3.2) are too general and might be insufficient for generating proper explanations for the active user [15]. Instead, Local Explanation Mining calculates first the neighborhood of the current user, and executes the rule mining method considering as transactional data only the associated to such users. Therefore, for each different user is necessary to execute a new association rule mining detection, for using them in the subsequent explanation generation process.

Algorithm 1. *Explaining matrix factorization approaches with global association rules*

Input: Initial user-recipe rating matrix, input data
Train the matrix factorization model to complete the initial user-recipe matrix
For each user i, obtain the top n recommendation list.
For each top n recommendation list, obtain the top p nutrition-aware re-ranked recipe list.
For each user i, obtain a transaction T_i containing the indexes of the D ratings in $\hat{R}$, associated to i
Generate the set Z_i of rules $(X \rightarrow Y)$ from the transaction list T, satisfying the selected interestingness measure
for all** user i=1,2, ..., N **do
    Compute the list of association rules $(X \rightarrow Y)$ in Z, being $X \in \{train\}$ and $Y \in \{top\ m\ nutrition-aware\ list\}$
    Return list of recommended recipes, and corresponding rules $(X \rightarrow Y)$ as explanations.)
end for
end

Algorithm 2. *Explaining matrix factorization approaches with local association rules*

Input: Initial user-recipe rating matrix, input data, neighborhood size
Train the matrix factorization model to complete the initial user-recipe matrix
For each user i, obtain the top n recommendation list.
For each top n recommendation list, obtain the top p nutrition-aware re-ranked recipe list.
for all** user i=1,2, ..., N **do
    Calculate the set G of k nearest neighbors considering their rating values
    For each user $g \in G$, obtain a transaction T_g containing the indexes of the D ratings in $\hat{R}$, associated to g
    Generate the set Z_i of rules $(X \rightarrow Y)$ for user i, satisfying the selected interestingness measure
    Compute the list of association rules $(X \rightarrow Y)$ in Z, being $X \in \{train\}$ and $Y \in \{top\ m\ nutrition-aware\ list\}$
    Return list of recommended recipes, and corresponding rules $(X \rightarrow Y)$ as explanations.)
end for
end

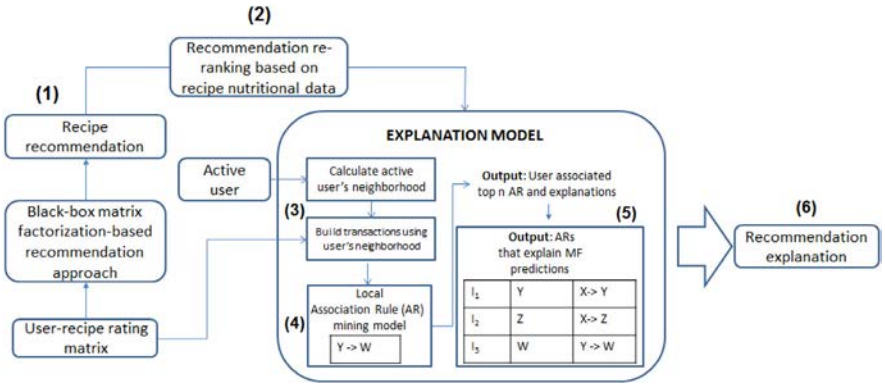


Fig. 4. Local Explanation Mining for post hoc interpretability in Recommender Systems.

This approach comprises the following steps, which are also represented in Fig. 4, build over the Global Explanation Mining approach (Fig. 3). Local Explanation Mining is also formalized in Algorithm 2, and detailed as follows.

1. Input data: The input data for Local Explanation Mining is very similar to the associated to the previous approach. In addition to the recipe information and the preference matrix, in this case it is also necessary the amount of nearest neighbors that will be used for building the local transactional dataset that is used as base for rule mining for the current user.
2. Train matrix factorization model and prediction: In a similar way to the Global Explanation Mining method, in this case the traditional black-box matrix factorization method [29] is used for recommendation generation (Step 1 in Fig. 4).
3. Nutrition-aware re-ranking of the recommendation list: Here the recipes in the retrieved recommendation list are

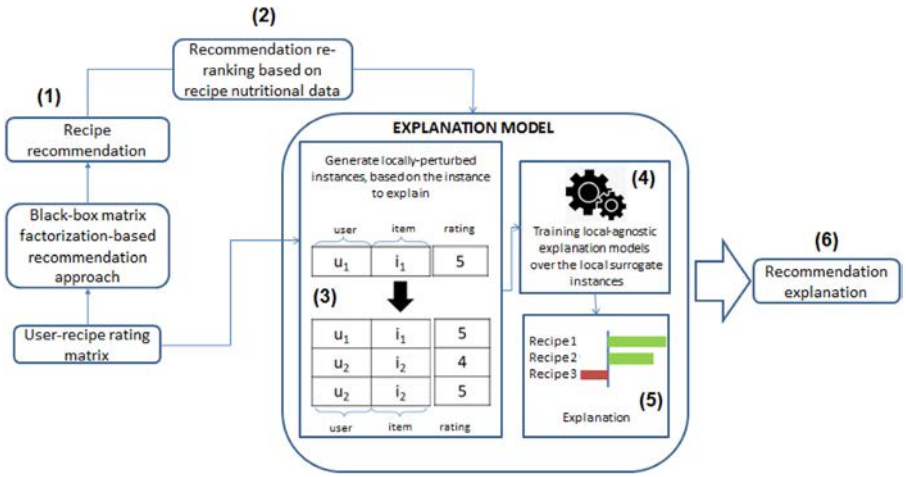


Fig. 5. Locally interpretable model-agnostic explanation model.

re-ranked according to the criteria presented at Eq. (6), in Section 3.1. Afterwards the top p recommendations of the re-ranked list are used (Step 2 in Fig. 4).

4. Train Association Rules (the interpretable model): This stage contains the most substantial difference of this approach in relation to Global Explanation Mining.
   In this case, for each user it is generated a different set of the required transactions T for rule mining [58], taking as base the preferences associated to the users in the neighborhood of the active one (Line 7 at Algorithm 2) (see also Step 3 in Fig. 4). These transactions are used for building local association rules, following the traditional approach formerly pointed out by Sarwar et al. [59] (Line 8 at Algorithm 2) (see also Step 4 in Fig. 4).
5. Output association rules: The output of the previous stage is a list of local rules representing relationships between recipes in the matrix factorization predictions, in the context of the neighborhood of the current user. In a similar way to Global Explanation Mining, for such user it is only filtered to a subset the rules where the antecedent X is in the user training data and the consequent Y is in the top p nutrition-aware recommendation list (Line 9 at Algorithm 2) (see also Step 5 in Fig. 4). The corresponding explanation can be then seen as the antecedents of the corresponding rules, and therefore they are both retrieved (Line 10) (Step 6 in Fig. 4).

3.4. An improved locally interpretable model-agnostic explanation model (LIRE)

This section adapts a locally interpretable model-agnostic explanation model to the recipe recommendation framework, recently proposed [18]. In this work, a explanation instance is a 3-tuple $(u, i, f(u, i))$ where $u \in U$, $i \in I$, and $f(u, i) \in R^+$ represents a rating prediction that is obtained by a black-box RS. Here the authors are focused on reaching the main interpretable features that are able to explain $f(u, i)$, instead of showing the working principle of the black-box model.

Fig. 5 presents the steps of this model, which are explained in detail as follows:

1. Black-box model training, execution, and nutrition-aware re-ranking: In a similar way to the previous works, here a traditional matrix factorization approach [29] will be used as black-box model for recommendation generation. Such model leads to the top n recommendation list, which are re-ranked according to the nutrition-aware criteria already pointed out in Eq. (6), to obtain the final top p recommendation list (Steps 1 and 2 in Fig. 5).
2. Generation of the locally-perturbed instances, based on the instance to explain: The concept of locally-perturbed instance is closely related to the locally interpretable features, regarding such features are used to represent the instances.
   Such interpretable features are related to feature names that represent directly comprehensible pieces of domain knowledge. Formally, in this work a set of interpretable features is represented by the set I of n items names $I = \{I_1, I_2, \ldots, I_n\}$, each one associated with a domain of value $dom(I_1), \ldots, dom(I_n) = R_+^n$. A feature vector over I is represented then as a n-tuple $t = (c_1, c_2, \ldots, c_n)$, $t \in R_+$. Equivalently, the tuple is viewed as a function t as $I \rightarrow \cup_k dom(I_k)$, being $t(I_k)$ the value c_k and $t_{|I'}$ is the restriction of t to the subset $I' \subseteq I$. $t(I_k) = t_{|I_k}$.
   Summarizing, for a 3-tuple $(u, i, f(u, i))$, the list of interpretable features is formalized as the tuple $t^u_{|\cup_{j \neq i} I_j}$. This can be considered as the restrictions of t to the subset of items $j \in I$, being $j \neq i$. This representation is focused on associating a real value to each feature name that represents the importance of the feature for the current explanation instance.
   Taking as base this context and using as input the instance to explain, the perturbations are defined as a random modification of the values of the tuple $t^u_{|\cup_{j \neq i} I_j}$, based on some Gaussian distribution [18]. As alternative, perturbations will be considered as randomly picked users from the same cluster as the user u for which the explanation is to be computed (Step 3 in Fig. 5). In practice, a mixed approach will be used, where 50% of training instances originates from perturbations and the other 50% is generated from the cluster neighbors.
3. Training of the local-agnostic explanation models over the local surrogate instances: In Chanson et al. [18], the authors assume the definition previously pointed out by Nobrega and Marinho [17] and Ribeiro et al. [54], where the task of explaining a recommendation for certain user u over certain item i, is associated to finding the top n or minimal subset of interpretable features that maximize the fidelity of the surrogate model to the original model.
   This surrogate model represents the explainable model and needs to be approximated. As surrogate models, this proposal is focused on linear explanation models with the

form $g(z) = w \bullet t^u$, being t^u the vector of values constructed from $t^u_{|\cup_{j \neq i} I_j}$; and linked to interpretable features of user u in the explanation instance $(u, i, f(u, i))$. The explanation model $e_f(u, i)$ is formalized as:

$$e_f(u, i) = arg\ min_{w \in R^n} L(f(u, i), w \bullet t^u) + \omega(w) \quad (9)$$

Here L is a loss function that penalizes any difference between the original prediction $f(u, i)$ and the value predicted by the surrogate, explainable linear model $w \bullet t^u$. $\omega(w)$ is the complexity of the linear explanation model.

Therefore, the problem is reduced to find the most appropriate interpretable feature weight vector w. To this end, for explaining each independent prediction the current approach needs to identify a set of instances that are close to the instance to explain, for using such set as training for learning the local explanation model (Eq. (9)) (Step 4 in Fig. 5). This task is accomplished by two different approaches: (1) generating gradually perturbed instances around the instance to explain, finding the appropriate vector w for such set of instances, or (2) finding natural grouping of neighbors that share similar evaluation patterns, to estimate a local explanation.

The underlying problem is then formalized as a simple regression problem between this local training set T_{train} of instances expressed on interpretable features containing perturbations or cluster neighbors of user u, and the respective predictions Y_{train} either obtained by the main black-box recommendation model, or by the explainable lineal regression approach.

To solve the problem, the authors consider a LASSO regression model [60], that introduces a penalty term $\|w\|_1$ similar to $\omega(w)$ in Eq. (9):

$$diff = Y_{train} - w * T_{train} \quad (10)$$

$$e_f u, i = argmin_{w \in R^n} \{diff * diff^t + \lambda \|w\|_1\} \quad (11)$$

being λ the Lagrangian coefficient linked to the constraint that minimizes the sum of weights w.

4. Explanation generation: The obtained surrogate model contains a set of weights associated to each interpretable feature (see Step 5 in Fig. 5), focused on showing the local importance of the corresponding feature in the current instance. Therefore such features, in this case represented as other recipe names, can be used as a way to explain the current recommendation generation (Step 6 in Fig. 5). Such set will be referred as the "explanation set" in the remaining of the paper.

4. Experiment and analysis

This section evaluates the performance of the discussed explanation algorithms which have been explored, adapted, and applied to the cooking recipe recommendation domain [34]. With this goal in mind, such evaluation is developed over a cooking recipe recommendation dataset. At first, some details on the dataset, the evaluation metric, and the experimental protocol used, are presented. Specifically, we plan to analyze our results on issues such as:

- To study the explanation capability of the analyzed post-hoc models over a cooking recipe recommendation environment (Objective 1).
- To evaluate the effect of adding some nutritionally-aware criteria into the explainable recommendation frameworks (Objective 2).
- To compare the performance of each analyzed post-hoc explanation model, in the cooking recipe recommendation domain (Objective 3).

Table 3
General information on Food.com dataset.

Data	Description
Users	25075
Recipes	178264
Interactions	1125284
Rating range	[0,5]

4.1. Dataset

This evaluation uses the popular Food.com cooking recipes dataset, taking from Kaggle [61]. Such dataset, which main figures are presented at Table 3, covers 18 years of user interactions and uploads on Food.com (formerly GeniusKitchen). Each recipe contains associated ingredients, list of used techniques, and nutritional information. In addition, for each user interaction with certain recipe, it is a registered a rating value in the range [0, 5] and a textual review. Considering the sparsity of the dataset, this work will consider the interactions of users with more than 150 interactions, over the items with more that 500 associated interactions.

Some of the methods presented across this work depend on item features for recommendation/explanation generation. In this context, we will consider recipe's ingredients as features. They are represented as binary features, that consider the presence or absence of the ingredient in the current cooking recipe.

4.2. Evaluation metrics

The main evaluation metric that is included in this study is Model Fidelity [15]. Model fidelity is defined as the proportion of the recommendations provided by the black-box model, that can be explained by the white box explainable recommendation model (Eq. (12)):

$$Modelfidelity = \frac{|explainable\ items \cap recommended\ items|}{|recommended\ items|} \quad (12)$$

In this direction, it is worthy to mention that in this work we are not focused on evaluating the performance of the main black-box recommendation method, and for this reason we do not use accuracy-oriented metrics such as Precision, NDCG, etc [62]. Instead, we are focused on measuring the explanation capabilities of the discussed explanation models, over the recommendations generated by the black-box approach. Therefore, we use explanation-oriented evaluation measures, such as the model fidelity [14,15].

4.3. Evaluation protocol

The post-hoc explanation models discussed across this work, will be tested using the dataset built from Food.com data (Section 4.1). The popular Surprise Python scikit will be used as base for implementing the experiments [63].

The training of the black-box recommendation model will be developed by considering the basic matrix factorization approach, also known as Funk SVD by some authors [29]. As parameters, it will be used the default values of this method in the Surprise framework (see Table 4). The tuning of these values is out of the scope of this work; however they were obtained from the related literature [29,63], and are usual choices for reaching effective recommendations.

Furthermore, it will be considered the reaching of the performance criteria (i.e. model fidelity), with and without incorporating the top n recommendations re-ranking stage based on the nutritional value of the recipe. As was previously stated,

Table 4
Parameters for the matrix factorization approach used as black-box in the proposal.

Parameter	Value
$n - factors$	20
$n - epochs$	20
$init - mean$	0
$init - std - dev$	0.1
$learning - rate$	0.007
$regularization - term$	0.02

Table 5
Model fidelity of the I2ICB explanation scheme in top n recommendations. δ represents the minimum exclusive threshold for a Jaccard similarity between a previously preferred item and the recommended item, to be considered the former as a valid explanation.

δ	0	0.1	0.2	0.3	0.4
I2ICB (top 3)	0.763	0.56	0.182	0.128	0.078
I2ICB (top 5)	0.764	0.54	0.176	0.133	0.076
I2ICB (top 10)	0.746	0.506	0.158	0.123	0.067

the objective is to evaluate whether this nutritional knowledge domain criterion impacts on the explanation capabilities of the proposal.

We want to evaluate the explanation capabilities of the discussed approaches, over *any* top scored unknown item for each user. To reach this goal we follow the subsequent steps:

1. Train the black-box recommendation model using the available dataset.
2. For each unknown item of each user in the dataset, predict a rating value using the trained model.
3. For each user, generate the top n recommendation list based on the predicted values.
4. For each recommended item, generate the corresponding explanation using methods presented at Section 3.
5. Calculate the model fidelity (Eq. (12)) of each explanation model.

4.4. Results

This section presents the experimental results associated to the presented protocol, focused on the cooking recipe recommendation domain. According to the initial objectives presented at the beginning of this section, it is focused on exploring the explanation capability of the analyzed post-hoc models (Objective 1, Section 4.4.1), exploring the effect of adding the nutritional information into the explanation performance (Objective 2, Section 4.4.2), and performing a comparison across the different models (Objective 3, Section 4.4.3). Eventually, some case studies are also presented (Section 4.4.4), and finally the main experimental findings are briefly synthesized (Section 4.4.5).

4.4.1. Exploring the explanation capability of the analyzed post-hoc models

This subsection is focused on evaluating the explanation capability of the analyzed post-hoc approaches, based on the model fidelity criteria, and without considering the nutritional recommendations re-ranking. Such approaches are: the simply-to-explain post-hoc explanation schemes of Shmaryahu et al. [16], the global explanation mining approach [15], the local explanation mining approach [15], and the LIRE approach [18].

Simple post-hoc explanations:. Here it is explored in the cooking recipe recommendation domain, two simple post-hoc explanation schemes pointed out at Section 3.1. These methods are the item–item content-based explanation scheme (I2ICB), and the item–item collaborative filtering explanation scheme (I2ICF). Here we recall that cooking recipe features are represented by their associated ingredients.

Following the overall experimental procedure presented at Section 4.3, we evaluate this approach by calculating the model fidelity of the proposal, on the discussed dataset (see Section 4.1). To reach this goal, we generate the top n recommendation list (exploring $n = \{3, 5, 10\}$) for each user using the matrix factorization method.

Subsequently we try to explain each item recommendation, by using the two mentioned explanation schemes, for identifying relationships between current recommendations and the set of previously preferred items. In practice, the model fidelity (Eq. (12)) is then characterized as the proportion of recommended items that could be connected with previous preferences of the current user.

At first, Tables 5 and 6 present the fidelity of the I2ICB scheme, for explaining the top n recommendations for each user, considering different values of n. For building Table 5 it is taken into account for each top n recommended item, the similarity between them and the previously preferred items by the current user (i.e. those with a rating value equal or greater than 4). If such similarity is greater than a threshold δ (see Eq. (7)), then the item is considered as explainable, and contributes to the calculation of the model fidelity value (Eq. (12)). In this way, Table 5 shows the fidelity values for $\delta \in [0; 0.4]$. At first, for $\delta = 0$ (assuming as explainable any item with at least some common feature with previously preferred items), the fidelity values yield around 0.75 for all the considered sizes of the recommendation lists. In practice, it means that for each 3 out of 4 recommended cooking recipes, they can be connected and justified with a previously preferred recipe by the current user.

However, for larger values of δ, Table 5 also shows that too few recipes can be justified with strongly connected and previously preferred items. In fact, when a previously preferred recipe is considered as a proper explanation if the similarity threshold is higher than $\delta = 0.4$, only the 6%–7% of recommended recipes are able to find proper explanations. Furthermore, for larger values of δ, the fidelity values decrease substantially, tending to 0. For this reason, δ was evaluated in the range [0; 0.4].

Finally, it is also relevant that a best fidelity tends to be obtained for small recommendation lists. It suggests that it is more possible to find a proper explanation for those items at the top of the recommendation list, as could be expected.

Beyond these results associate to the parameter δ, it is also necessary to characterize the I2ICB post-hoc explanation scheme using other criteria more representative of the current cooking recipe domain. In this direction the parameter m is introduced, for representing in this case the minimum number of common features (i.e. ingredients, see Section 4.1) that should have a previously preferred recipe in common with someone in the current recommendation list, to be considered a valid explanation. Table 6 presents these results. Here it is worthy to note than even considering at least two common ingredients with a preferred item ($m = 2$), the discussed framework is able to explain around the half of the recommended items. Furthermore in a more strict scenario that considers at least 5 common items ($m = 5$), the framework is able to achieve a model fidelity over 0.1.

In addition to these numerical results it is necessary the development of a qualitative analysis, to verify according to the cooking recipe domain, whether the generated explanations follow the common sense and whether the common ingredients used for generating it are actually relevant features to connect past preferences and currently top n recommended recipes.

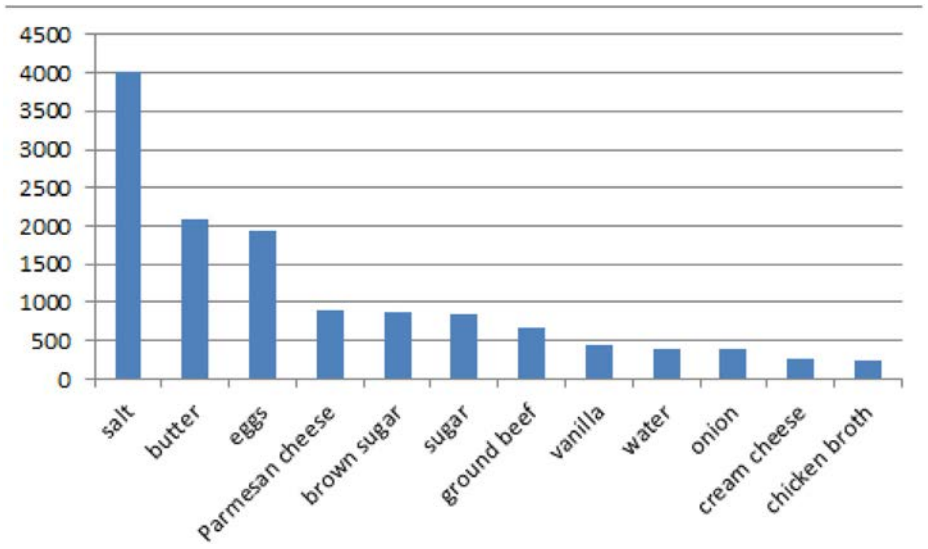


Fig. 6. Most common ingredients used as base in the I2ICB recommendation explanation method ($m = 1$).

Table 6
Model fidelity of the I2ICB explanation scheme in top n recommendations. m represents the minimum exclusive threshold for a number of common features between a previously preferred item and the recommended item, to be considered the former as a valid explanation.

m	1	2	3	4	5
I2ICB(top 3)	0.763	0.534	0.211	0.128	0.104
I2ICB (top 5)	0.764	0.505	0.202	0.133	0.107
I2ICB (top 10)	0.746	0.448	0.182	0.123	0.1

Table 7
Model fidelity of the I2ICF explanation scheme in top n recommendations. δ represents the minimum exclusive threshold for a Jaccard similarity between a previously preferred item and the recommended item, to be considered the former one as a valid explanation.

δ	0.06	0.09	0.12	0.15	0.18
I2ICF (top 3)	0.977	0.972	0.879	0.717	0.375
I2ICF (top 5)	0.977	0.972	0.869	0.678	0.34
I2ICF (top 10)	0.977	0.972	0.857	0.628	0.297

Fig. 6 illustrates the most relevant features (i.e. ingredients) that were used as base for linking the recommendation list with past preferences through I2ICB approach with $m = 1$, as well as the amount of generated explanations based on such features. As it could be expected, the figure reflects some ingredients that are pretty irrelevant for linking two recipes, such as salt, sugar, onions, or water. However, we think that it is relevant that several ingredients with a higher specificity degree (e.g. eggs, Parmesan cheese, ground beef, chicken broth, vanilla), have also driven to explanation generation. The preference over foods with such ingredients can be associated to more specific user characteristic, evidencing this result that the explanation scheme can be coupled to such characteristic, and therefore contributing to the desired wow effect of the generated explanations [64].

In a different direction, Table 7 presents the results of the I2ICF explanation scheme, also proposed by Shmaryahu et al. [16]. This table is similar to Table 5 in the sense that it used a threshold δ to compare a currently recommended item and a previously preferred one, to determine whether the last one can be used as explanation for the current recommendation. However, here we found out that the similarity values among items, tend to be close overall. Particularly, it is showed that for $\delta = 0.06$, it can be reached a fidelity greater than 0.97; however, for $\delta = 0.18$, the fidelity is already around 0.3. These close similarity values make this approach unable to discriminate between items that can be valuable explanations for a current recommendation. Therefore, at least for the current data, I2ICF cannot be identified as a proper explanation scheme for supporting recipe recommendations.

Here it is also worthy to note that for $\delta < 0.06$ the obtained fidelity is very close to 1, while it substantially decreases for $\delta > 0.18$. Therefore, we report only the values in the range [0.06, 0.18].

Global explanation mining. In this scenario we are focused on exploring the behavior of the Global Explanation Mining approach in the cooking recipe recommendation domain.

In a similar way to the previous approach based on simple-to-explain methods, exposed at Section 3.1, here Global Explanation Mining is used to explain the top n recommendation list ($n = \{3, 5, 10\}$) generated through a matrix factorization method.

Specifically, the rule mining procedure considers as optimal values $min - support = 0.015$ and $min - confidence = \{0.2, 0.3, 0.4, 0.5, 0.6, 0.7\}$, being such values in correspondence with the typical values of minimum support and confidence in traditional RS datasets, initially reported in this explanation method [15]. Furthermore, in a similar way to Peake and Wang [15], we focus on rules with consequents of size 1.

Table 8 presents the fidelity of this approach in the food recommendation dataset, as well as the average number of explanation rules mined for each user. The table shows that this explanation approach is able to reach a high fidelity for lower minimum confidence values (e.g. a fidelity over 0.91 for $min - conf = 0.2$). However, it has the cost of managing a higher number of rules for explanation, as well as the possibility that some of them were actually not an appropriate explanation. In contrast, when the number of rules decreases, the model fidelity also decreases. In other direction, it is also relevant that, in a similar way to the tendency at the explanation approach presented at Section 3.1, here the model fidelity decreased for a larger size of the recommendation list. The results is expected, taking into account that top items should be the more preferred ones, and therefore easier to explain.

In this scenario, we use $min - conf$ values in the range [0.2, 0.7] because for lower values the number of rules increases considerably. On the other hand, for larger values too few rules are detected, the method takes too much time to discover it, and the fidelity values tend to 0.

Table 8
Model fidelity and number of rules of the Global Explanation Mining scheme in top n recommendations, considering several minimum confidence levels of the identified rules.

$min-conf$	0.2	0.3	0.4	0.5	0.6	0.7
Expl-Min (top 3)	0.917	0.689	0.452	0.297	0.14	0.053
Expl-Min (top 5)	0.902	0.658	0.426	0.268	0.12	0.041
Expl-Min (top 10)	0.879	0.614	0.379	0.219	0.09	0.03
Amount of rules	1871	1577	1032	626	298	115

Table 9
Model fidelity and number of rules of the Local Explanation Mining scheme in top n recommendations, considering several minimum confidence levels of the identified rules.

$min-conf$	0.2	0.3	0.4	0.5	0.6	0.7
Expl-Min (top 3)	0.892	0.671	0.420	0.317	0.186	0.107
Expl-Min (top 5)	0.878	0.647	0.401	0.293	0.167	0.094
Expl-Min (top 10)	0.858	0.610	0.372	0.271	0.151	0.084
Amount of rules	4671	4529	4155	3824	3094	2441

Local explanation mining. Here it is explored the behavior of the Local Explanation Mining approach, discussed at Section 3.3.

The evaluation of this method uses the same parameters of the Global Explanation Mining method, presented at the previous subsection. In this case, the $min-support$ parameter was adjusted, using in this case $min-support = 0.04$. In addition, for each user, the profiles of the top 100 nearest users are used as input data for the corresponding local rule mining.

Table 9 presents the fidelity of this approach, as well as the average number of explanation rules, mined for each user.

Overall, the obtained results are similar to the associated to the Global Explanation Mining approach. The best fidelity values were obtained for small recommendation lists and small $min-conf$ values, being $min-conf \in [0.2, 0.7]$ based on the similar criteria also used in Global Explanation Mining approach. Here it is also relevant that for the same $min-conf$ values but with a small $min-support$, as could be expected, a larger number of rules are identified.

LIRE. Furthermore, we evaluate the fidelity of the explanations generated by the LIRE approach [18] in the cooking recipe recommendation domain.

In a similar way to the previous scenarios, we use this approach to explain the top n recommendation list ($n = \{3, 5, 10\}$) generated through the matrix factorization method described in the experimental protocol. As it was explained in the method description, for each item recommendation LIRE is focused on determined an "explanation set" of other items, which could be a valid explanation of the current item recommendation. Such recommended item is then considered as explainable, if all the items at such set represent actual preferences of the user ($r_ui >$ 4).

Table 10 presents the fidelity of this approach for the top n recommendation list, being $n \in \{3, 5, 10\}$ in a similar way to the previous approaches. Furthermore, it considers different sizes of the explanation set, in the range [1, 10]. Here it is relevant that the approach reaches a similar fidelity for the three mentioned top n recommendation scenarios. For all cases, LIRE reaches a fidelity close to 0.94 when $exp-set-size = 1$, which are across the best fidelity values analyzed in this research paper. However, this very small size of the explanation set can be a shortcoming considering the goal of providing trustworthy and convincing explanations.

In this direction, for $exp-set-size = 2$ the fidelity of the proposal lies around 0.880, which is also a relevant result regarding the values associated to the other proposals. As could be expected, for larger sizes of such sets, the fidelity of the generated explanations quickly decreases, taking into account that the explanation sets can be then composed by some items that are not actual preferences of the current user. As example, for $exp-set-size = 5$ the fidelity lies around 0.225, and for $exp-set-size = 7$, around 0.07.

4.4.2. Exploring the effect of adding the nutritional information into the explanation performance

This subsection is focused on exploring the effect of the introduction of the nutritional re-ranking criteria in the fidelity of each recommendation explanation approaches, discussed at Section 3. With this goal in mind, the protocol previously presented at Section 4.3 is used for evaluating these explanation approaches using the nutritional recommendations re-ranking. Furthermore, for each case it is developed a comparison against the same approach using the same size of the top n recommendation list, but without considering the nutritional criteria, for measuring its effect in the recommendation performance.

Simple post-hoc explanations. At first, this study is developed for the simple post-hoc explanations [16]. Here, it is considered the incorporation of the nutrition-aware re-ranking approach (I2ICB+nut), considering (n = 10, p = 5) and (n = 5, p = 3) (i.e. top n recommendations generation, which are reduced to p recommendations after nutrition-aware re-ranking). The achieved results are respectively compared with their counterparts in the cases that do not use nutritional information (i.e. I2ICB (top 5) and I2ICB (top 3)).

Table 11 presents these results. In this case for larger values of δ, specifically for $\delta \geq 0.2$, both approaches outperform their respective counterparts that do not consider the nutritional knowledge, being in this case I2ICB (top 3) and I2ICB (top 5). On the other hand, for $\delta < 0.2$, the incorporation of nutritional knowledge leads to a reaching of lower fidelity values. Such results suggest that when recommendation explanations are only generated through items very similar to the currently recommended, the incorporation of the nutrition-aware criteria can lead to an improvement in the recommendation generation. However, where explanations are also driven through less similar items, such criteria do not introduce an improvement in the explanation capability.

Table 12 also presents the evaluation of incorporating the nutrition-aware re-ranking in the top n recommendation process, in the I2ICB explanation scheme that consider the minimum exclusive threshold value m. As could be expected, these results closely match with those associated to the analysis of the parameter δ. In this case, for $m \geq 3$, the approaches that incorporate nutritional knowledge obtain a better fidelity. On the other hand, for lower m values the best fidelity was obtained for the methods that do not incorporate this kind of knowledge.

Furthermore, Table 13 presents the results for the I2ICF explanation approach. In this experimental scenario it is not reflected a well-defined behavior of the methods that incorporate nutritional knowledge (I2ICF (top 5, p = 3) and I2ICF (top 10, p = 5)),in relation to their counterparts that do consider such knowledge. In this case, both approaches reach a similar performance for $\delta = \{0.06, 0.09, 0.12\}$, while for larger values of δ, the approach that does not consider nutritional knowledge reaches the best performance.

Summarizing, the results suggest that the incorporation of nutritional information into the post-hoc explanation approaches presented by Shmaryahu et al. [16], leads to an improvement in the model fidelity for larger values of the parameters δ and m in I2ICB schemes, and keeps a similar performance for lower values of the parameter δ in the I2ICF scheme.

Table 10
Model fidelity of the LIRE approach scheme in top n recommendations, considering several sizes of the explanation set.

$exp-set-size$	1	2	3	4	5	6	7	8	9	10
LIRE (top 3)	0.936	0.880	0.571	0.373	0.225	0.137	0.068	0.040	0.017	0.012
LIRE (top 5)	0.939	0.878	0.570	0.374	0.222	0.133	0.067	0.039	0.020	0.012
LIRE (top 10)	0.938	0.879	0.572	0.376	0.225	0.132	0.071	0.039	0.019	0.013

Table 11
Model fidelity of the I2ICB explanation scheme in top n recommendations, with and without nutritional information. δ represents the minimum exclusive threshold for a Jaccard similarity between a previously preferred item and the recommended item, to be considered the former as a valid explanation.

δ	0	0.1	0.2	0.3	0.4
I2ICB (top 3)	0.763	0.56	0.182	0.128	0.078
I2ICB+nut (top 5,p = 3)	0.728	0.473	0.2	0.177	0.126
I2ICB (top 5)	0.764	0.54	0.176	0.133	0.076
I2ICB+nut (top 10,p = 5)	0.716	0.478	0.237	0.198	0.133

Table 12
Model fidelity of the I2ICB explanation scheme in top n recommendations, with and without nutritional information. m represents the minimum exclusive threshold for a number of common features between a previously preferred item and the recommended item, to be considered the former as a valid explanation.

m	1	2	3	4	5
I2ICB(top 3)	0.763	0.534	0.211	0.128	0.104
I2ICB+nut (top 5,p = 3)	0.728	0.424	0.229	0.177	0.177
I2ICB (top 5)	0.764	0.505	0.202	0.133	0.107
I2ICB+nut (top 10,p = 5)	0.716	0.403	0.255	0.198	0.197

Table 13
Model fidelity of the I2ICF explanation scheme in top n recommendations, with and without nutritional information. δ represents the minimum exclusive threshold for a Jaccard similarity between a previously preferred item and the recommended item, to be considered the former as a valid explanation.

δ	0.06	0.09	0.12	0.15	0.18
I2ICF (top 3)	0.977	0.972	0.879	0.717	0.375
I2ICF+nut (top 5,p = 3)	0.977	0.973	0.878	0.651	0.269
I2ICF (top 5)	0.977	0.972	0.869	0.678	0.34
I2ICF+nut (top 10,p = 5)	0.977	0.973	0.881	0.642	0.255

Global explanation mining. This subsection is focused on exploring the effect of the addition of nutritional information in the Global Explanation Mining approach [15].

In a similar way to the simpler post-hoc approaches [16], here it is considered the incorporation of the nutrition-aware re-ranking approach (see Section 3.1), considering (n = 10, p = 5) and (n = 5, p = 3) (i.e. top n recommendations generation, which are reduced to p recommendations after nutrition-aware re-ranking).

Table 14 presents the model fidelity of Global Explanation Mining after the incorporation of nutritional information (Exp-Min+Nut), as well as compares it with the same approach but without such information (Expl-Min). As can be expected, for several scenarios the application of the re-ranking approach for prioritizing the recommendation of nutritionally-appropriated recipes, introduces the suggestion of some recipes that are not possible to explain. Such facts imply the reduction of the fidelity at the corresponding model, in relation to a similar model that does not consider such re-ranking (e.g. Expl-Min+Nut (top 5, p = 3) vs. Expl-Min (top 3); or Expl-Min+Nut (top 10, p = 5) vs. Expl-Min (top 5), see Table 14). Such decreasing becomes relevant when only rules with very high confidence values are mined.

In contrast, for low confidence values, specifically, for $min-conf = 0.2$, it is worthy to note that Table 14 shows that the fidelities of the nutrition-aware methods reach a value of 0.902 and 0.896, which are very close to their counterparts that do not consider this criterion (getting fidelities of 0.917 and 0.902). This fact suggests that when a relevant set of rules are available to build the explanations, the nutritionally-appropriated items could be easier to explain, in relation to formerly top preferred items. Further studies are necessary to validate this hypothesis.

Local explanation mining. This subsection is focused on exploring the effect of the addition of nutritional information in the Local Explanation Mining approach [15].

Table 15 presents the results associated to this approach. In a similar way to the behavior of Global Explanation Mining (Table 14), here for $min-conf = 0.2$ and $(top10, p = 5)$, the nutrition-aware methods reach a similar fidelity value in relation to its counterpart $(Expl-Min(top5))$, while for larger $min-conf$ values, the methods that do not consider nutritional issues, outperform those that consider it.

Such similar fidelity value also reinforces the previously mentioned hypothesis that more rules make nutritionally-appropriated items easier to explain, taking into account that Local Explanation Mining generates a larger number of rules in relation to Global Explanation Mining, as was exposed in Section 4.4.1.

LIRE. Finally, Table 16 presents the effect of adding the nutritional information in the top n recommendation list generation, in the LIRE approach.

In this case, Table 16 evidences that the nutritional-aware re-ranking leads to an increasing in the model fidelity values that can be reached by the LIRE approach.

For the top 3 recommendation task, $LIRE + Nut(top5, p = 3)$ outperforms $LIRE(top3)$ for 8 out of 10 experimental scenarios, achieving very similar results in the two remaining scenarios (i.e. size 2 and 6, of the explanation set). In a very similar way, for the top 5 recommendation task, $LIRE + Nut(top10, p = 5)$ outperforms $LIRE(top5)$ for 8 out of 10 experimental scenarios, with similar results in the remaining two, in the case for explanation sets with sizes 6 and 9.

These results evidence a different behavior in relation to the Explanation Mining approaches. While the incorporation of nutritional information decreases the fidelity of the model in Explanation Mining (Sections 3.2 and 3.3), here at the LIRE approach (Section 3.4) it leads to a slight improvement in such values. We think that this improvement is with the cost of providing less convincing explanations considering they are limited by the established size of the explanation set (Section 3.4). A detailed comparison across the discussed approaches, will be presented in the next subsection.

4.4.3. Comparison across the proposals

Previous subsections have been focused on studying the performance of previously presented post-hoc explanation approaches, with and without nutritional information. This subsection is focused on performing a comparison between them, being centered in two directions: (1) performing a direct comparison according to the model fidelity criterion, and (2) studying the overlapping degree between the set of explanations generated by each different explanation approach.

Table 14
Model fidelity and number of rules of the Global Explanation Mining scheme in top n recommendations, considering several minimum confidence levels of the identified rules, and with and without nutritional information.

$min - conf$	0.2	0.3	0.4	0.5	0.6	0.7
Expl-Min (top 3)	0.917	0.689	0.452	0.297	0.14	0.053
Expl-Min+Nut (top 5, p = 3)	0.902	0.633	0.390	0.226	0.089	0.020
Expl-Min (top 5)	0.902	0.658	0.426	0.268	0.12	0.041
Expl-Min+Nut (top 10, p = 5)	0.896	0.621	0.365	0.198	0.075	0.020

Table 15
Model fidelity and number of rules of the Local Explanation Mining scheme in top n recommendations, considering several minimum confidence levels of the identified rules, and with and without nutritional information.

$min - conf$	0.2	0.3	0.4	0.5	0.6	0.7
Expl-Min (top 3)	0.892	0.671	0.420	0.317	0.186	0.107
Expl-Min+Nut (top 5, p=3)	0.885	0.621	0.375	0.260	0.130	0.077
Expl-Min (top 5)	0.878	0.647	0.401	0.293	0.167	0.094
Expl-Min+Nut (top 10, p=5)	0.877	0.607	0.361	0.260	0.132	0.075

Table 16
Model fidelity of the LIRE approach scheme in top n recommendations, considering several size of the explanation set, and using nutritional information.

$exp - set - size$	1	2	3	4	5	6	7	8	9	10
LIRE (top 3)	0.936	0.880	0.571	0.373	0.225	0.137	0.068	0.040	0.017	0.012
LIRE+Nut (top 5, p = 3)	0.941	0.879	0.572	0.380	0.227	0.136	0.071	0.043	0.021	0.014
LIRE (top 5)	0.939	0.878	0.570	0.374	0.222	0.133	0.067	0.039	0.020	0.012
LIRE+Nut (top 10,p = 5)	0.941	0.881	0.571	0.375	0.223	0.130	0.069	0.042	0.020	0.014

Direct comparison between the proposals. Fig. 7 shows the performance of the six considered explanation approaches which are I2ICB with Jaccard similarity (a), I2ICB with minimum number of common features (b), I2ICF (c), Global Explanation Mining (d), Local Explanation Mining (e), and LIRE (f). Each approach depends on different parameters (X-axis), but the results are presented using the same scale for the fidelity values (Y-axis). The top 3 recommendation task was considered for all cases, even though the obtained results were similar for the top 5 and top 10 task, also considered previously in this experimental analysis.

From a general viewpoint, all the analyzed approaches perform similarly in the sense that they are able to reach fidelity values in a wide range located from around 0.7, to a value very close to zero depending on the corresponding parameter.

However, several difference across the methods can be also observed. At first it is relevant that the approaches that depend on item features (Fig. 7a and b) obtain a lower fidelity values in relation to other approaches such as Explanation Mining and LIME. It suggests that there are several relationships between items, which represent valid explanations for the performed recommendations, and that go beyond simple relationships between features.

In a different direction, it is necessary to mention the high fidelity values achieved by the I2ICF approach (c at Fig. 7). However, in contrast to the other approaches and as it was pointed out previously, the I2ICF fidelity decreases quickly for small changes of the parameter δ. It suggests then that this method is not able to discriminate between appropriate and not appropriate explanations for a current item recommendation.

Finally, Global Explanation Mining (Fig. 7d), Local Explanation Mining (Fig. 7e), and LIRE (Fig. 7f) are able to obtain fidelity values around 0.9 for several scenarios. In this context Global Explanation Mining obtains better fidelity values than Local Explanation Mining for lower $min{-}conf$ values, even though it needs a smaller number of rules as was pointed out in Section 4.4.1. It is also interesting that for high $min - conf$ values (i.e. only rules with a high trust level), the local approach outperforms the global one.

Furthermore, the LIRE approach (f) obtains fidelity values over 0.87 for $exp{-}set{-}size = 1$ and $exp{-}set{-}size = 2$, which are values comparable with the best performance of Explanation Mining. In addition, this approach does not depend on the generation of intermediate knowledge such as the rules in the explanation mining scenarios, allowing LIRE to generate the explanations in a more succinct way. However, it also has as shortcoming the fact that its fidelity decreases substantially for $exp - set - size \geq 3$, and therefore a high fidelity is only guaranteed with explanation sets with limited sizes.

In short, this analysis suggests that the selection of an appropriate explanation approach depends on several facts such as the desired expressiveness of the generated explanations, or the available computational capability to generate the rules that are necessary in some approaches. Furthermore, the number of available features for characterizing items is also necessary to consider, even though in this case the methods that directly depend on features (Fig. 7a and b), have a lower fidelity in relation to the other approaches.

In a different direction, it is worthy to mention that it was performed a direct comparison between the fidelity values obtained by the approaches incorporating nutritional information. The obtained results were very analogous with these previously discussed in this subsection. Therefore, we do not included them in a explicit way here.

Analysis of the overlapping degree. This subsection is focused on studying the overlapping degree between the explanations generated by each different explanation approach, in order to characterize the difference between them beyond the fidelity values.

To perform such analysis, we analyze the provided recommendations as well as its associated explanation, generated for 20 users, randomly selected. For each case, such explanations are generated through: (1) the I2ICB considering the minimum exclusive threshold with $m = 0$ (Section 3.1), (2) the Global Explanation Mining approach with $min - support = 0.2$ (Section 3.2), and (3) the LIRE approach with $exp{-}set{-}size = 2$ (Section 3.4); in the three cases without using nutritional information. These three methods have performed with a similar fidelity for the selected parameter values. For all scenarios, the top 3 recommendation task was considered. Afterwards, we analyze the items used for providing explanations for the recommendation generated in the three scenarios. Here we do not analyze Local Explanation Mining

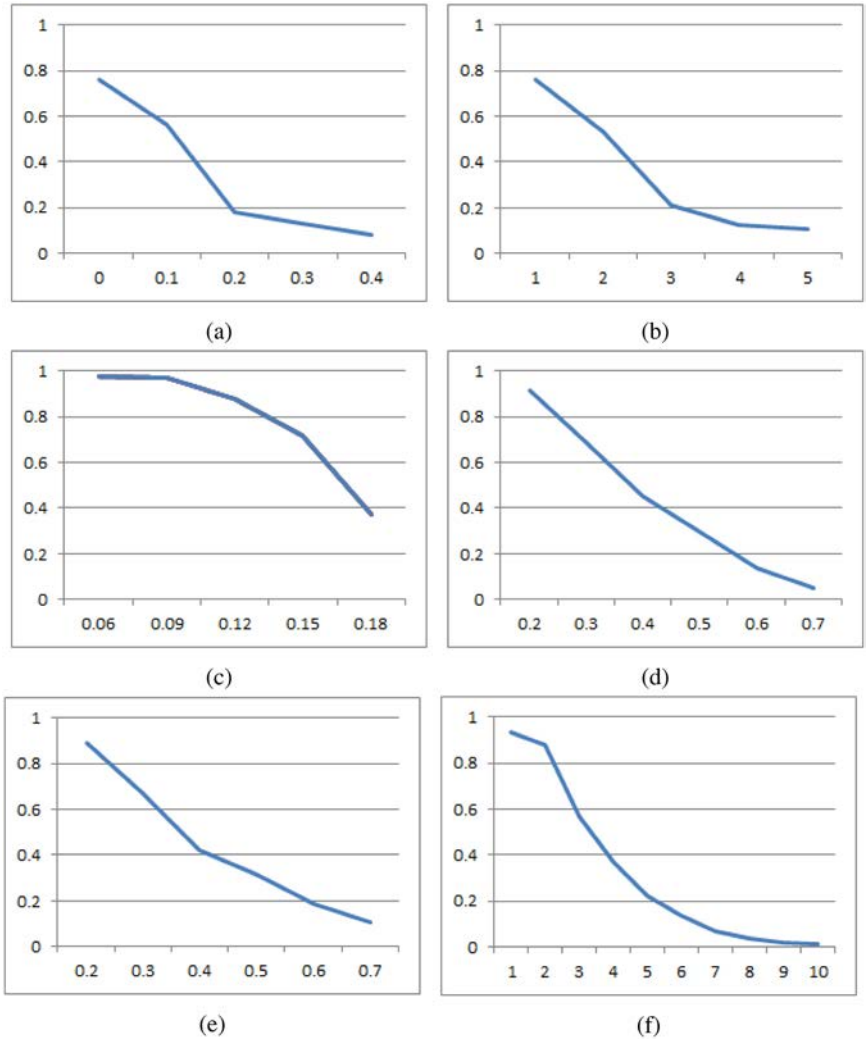


Fig. 7. Comparison across the discussed explanation approaches. (a) I2ICB with Jaccard similarity. (b) I2ICB with minimum number of common features. (c) I2ICF. (d) Global Explanation Mining. (e) Local Explanation Mining. (f) LIRE.

considering it performs similar to Global Explanation Mining, that was already included in this analysis.

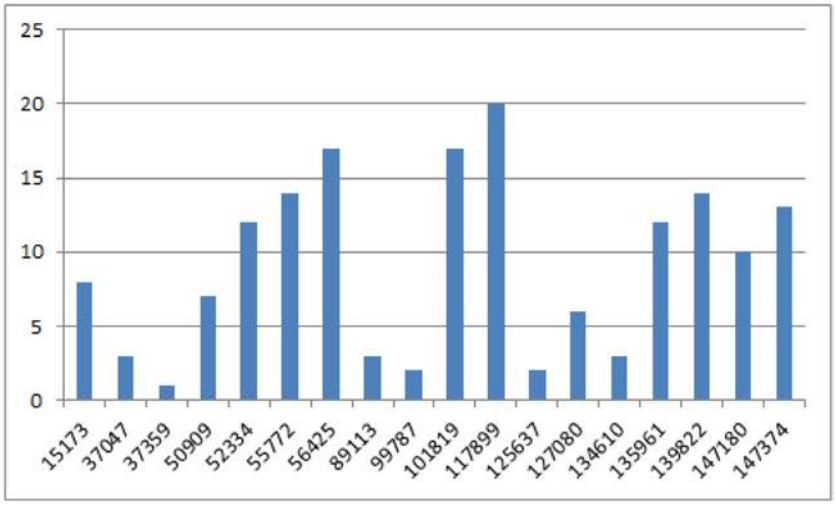
Fig. 8 shows three histograms presenting the frequency of the most common items used as explanation in the three corresponding explanation approaches, in the top 3 recommendation task for the 20 selected users. At first, it is relevant that disregarding the used explanation approach, the explanations tend to be supported by the same set of items.

However, a detailed analysis of Fig. 8 can identify some differences between the explanation methods. Specifically, it is relevant some differences between the I2ICB approach with $m = 0$ [16] (Fig. 8(a)) and the other two approaches (i.e. Global Explanation Mining at Fig. 8(b), and LIRE at Fig. 8(c)). In the case of I2ICB, items such as $id = 56425$ and $id = 101819$ are frequently used as explanations, while in the other two approaches, they are used with a lesser extent. Furthermore, other items such as $id = 99787$ with a lower frequency in I2ICB, are usually used for supporting explanations in the other two approaches. This behavior can be expected, considering that the explanations generated by I2ICB are supported by the features associated to each item (in this case ingredients of the recipes), while Global Explanation Mining and LIRE are based on the discovery of latent relationship between the items based on the user preferences. Based on such fact, it can be expected that they would provide different explanations for the same set of recommended items.

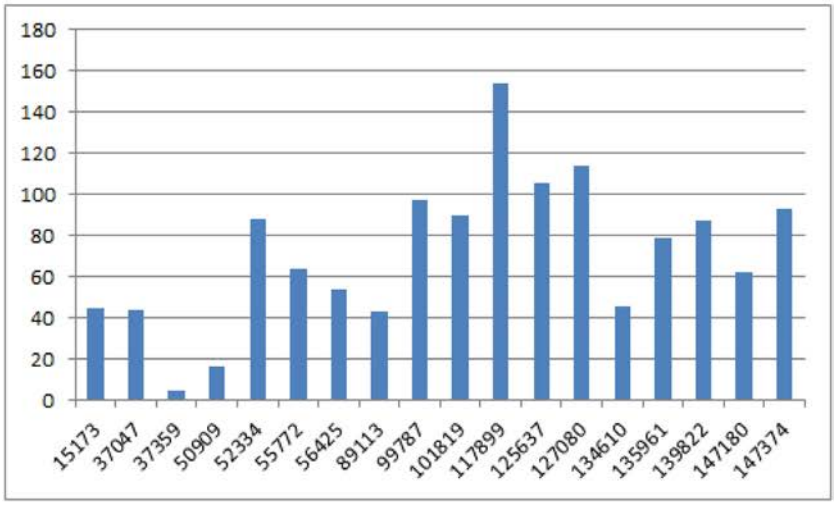
Furthermore, even though the analysis of Figs. 8(b) and 8(c) suggests a clearer correlation between the frequency of the explanation items linked to the Global Explanation Mining and the LIRE approaches, some difference can be also appreciated. In the case of Global Explanation Mining, the item $id = 117899$ was the most commonly used in the explanations, while for LIRE the item $id = 99787$ was the most used, being $id = 117899$ also used with larger extent. Other items such as $id = 125637$ also have relevant differences in their frequencies across these two approach, considering their relative frequencies in relation to the associated to the other items.

Summarizing, the analysis developed at this section suggests that even though each analyzed method can perform similarly according to their fidelity values, the generated explanation can be different and closely related to the working foundations of each approach. Therefore, it is also necessary to consider the nature of the data such as the amount of common ingredients, the amount of co-rated recipes, or the possibility of finding/building an appropriate neighborhood for the current user, before selecting the explanation approach to use.

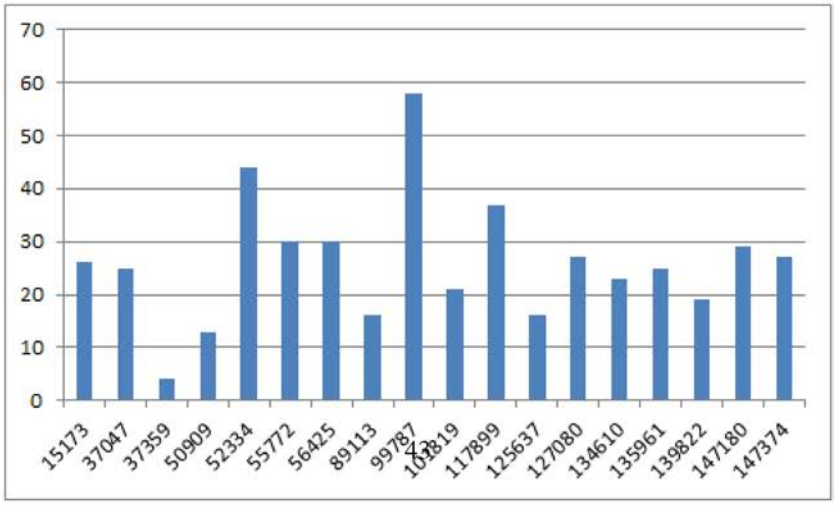
In a similar way to the previous subsection, the obtained results with the use of nutritional information were very analogous to the presented here without such information. Therefore, we do not included them in a explicit way.



(a)



(b)



(c)

Fig. 8. Most common items used as explanations. (a) I2ICB with minimum number of common features. (b) Global Explanation Mining. (c) LIRE.

Table 17
Examples of real explanations generated through the I2ICB explanation scheme, considering $m = 3$.

Recommendation	**banana cake with cream cheese frosting**	**creamy cajun chicken pasta**
Explanation	lighten up whatever floats your boat brownies banana banana bread 5 min cinnamon flop brunch cake	moist cheddar garlic oven fried chicken breast
Recommendation	**jo mama's world famous spaghetti**	**italian melt in your mouth meatballs**
Explanation	sloppy joes pizza joes	5 min cinnamon flop brunch cake

Table 18
Example of real explanation generated both through Local Explanation Mining and the I2ICB explanation scheme with $m = 1$.

Recommendation	**my family s favorite sloppy joes pizza joes**
Explanation	creamy cajun chicken pasta oven fried chicken chimichangas

4.4.4. Case studies

This subsection presents several case studies of recommendation explanations generated by the previously discussed approaches, as real examples that illustrate the particularities of each method.

Simple post-hoc explanations. At first, we present some explanations generated by the I2ICB simple post-hoc explanation approach. In this case we assume $m = 3$, for guaranteeing a strong relation between associated foods. Table 17 illustrates some recommended recipes and their corresponding justifications, showing that the I2ICB method is able to provide plausible explanations. Here it is observed that a recommendation of a dessert (banana cake), is explained by the previous consumption of other desserts (e.g. brownies and cinnamon cake), and other recipe having banana (i.e. banana bread). Similarly, a recommendation of a chicken with pasta was directly supported by the previous preference of a dish containing chicken breast; and a recommendation of spaghetti was explained by the previous preference of a pizza. Furthermore, the table also refers to possible explanations which nature are not completely clear, such as the recommendation of meatballs tailored by the previous preference of a brunch cake.

Explanation mining. In order to illustrate some explanations provided by the explanation mining approach and to perform a further analysis on the relationship between Explanation Mining and the approach based on simply-to-explain methods (Section 3.1), Tables 18 and 19 show some recommended items, that can be either explained or not explained, also by the Shmaryahu et al. scheme with $m = 1$. Such explanations are generated in this case by the Local Explanation Mining approach, which was able to generate a larger set of rules in relation to Global Explanation Mining.

Regarding Table 18, it presents an explanation example similar to the discussed in Table 17, where similar foods are linked as explanation mean. On the other hand, Table 19 illustrates some recipe explanations only generated through Explanation Mining, that even though they relate foods that do not have common ingredients, can be considered plausible explanations taking into account the common sense. In the first case, the preference over a bread recipe is explained by the preference for two recipe that could be associated to breakfasts (i.e. meatballs and a burrito). In the other case, a meatloaf recipe is explained by a preference over pork chops and a common dessert such as banana cake with cream cheese. These examples prove that it is possible to generate appropriate explanations by linking recipes without obviously common features, raising in this way the role of approaches such as Explanation Mining.

LIRE. Finally, this subsection presents some explanations generated through the LIRE approach, illustrated at Table 20.

Here it is presented the explanation generated through LIRE, of an item recommendation (banana cake with cream cheese frosting), already explained previously with the I2ICB approach but for a different user. Here it is interesting the explanation generated for this recommendation, that is support by two items that apparently do not have a direct connection with a cake, which are a roast (to die for crock pot roast) and a chicken pasta recipe (creamy cajun chicken pasta). However, on the other hand, the other recommendation-explanation pair presents more expected results, being here a recommendation of a chicken recipe (oven fried chicken chimichangas) justified with a previous preference of other chicken recipe (kittencal's moist cheddar garlic oven fried chicken breast); and pizza recipe that was also previous related with the former chicken recipe recommendation, by both the Explanation Mining and the I2ICB explanation scheme (see Table 18).

Summarizing, in the case of LIRE the illustrated examples show that it is able to generate explanations that have been also generated by the simple post-hoc explanation approaches, and by Explanation Mining. However, in some cases it can generate explanations which validity is not clear. Additional studies are then necessary for finding the role of such explanations and the causes that lead to their generation.

4.4.5. Summary

The developed experiments lead to the following findings:

- Globally, the discussed explanation methods are able to explain up to around 94% of the cooking recipe recommendations generated by a black-box recommendation method, in this case the matrix factorization-based collaborative filtering approach.
- The use of item-to-item content-based filtering as a complement for supporting the explanation of the black-box recommendation method output, is able to explain more than 75% of the generated recommendations (Tables 5–6). Herein, the use of parameters such as the number of common ingredients or the similarity degree, allows to manage the reaching of more or less trustable explanations. On the other hand, even though the use of item-to-item collaborative filtering explanations (Table 7) leads to a high model fidelity in some stages, we have not found an appropriate parameter to make this approach useful to discriminate between recipes that can be valuable explanations for a current recommendation.
- The use of the Global Explanation Mining approach helps to explain up to 92% of the generated recommendations (Table 8). Here the explanation capability is closely related to the number of generated rules for supporting explanations: a higher number of mined rules, a higher fidelity of the associated methods. On the other hand, the use of the Local Explanation Mining approach helps to explain up to around 90% of the generated recommendations (Table 9). Even thought, it is focused on a more sophisticated working principle, it usually performs worse than Global Explanation Mining.

Table 19
Examples of real explanation generated both through Local Explanation Mining, but not by the I2ICB explanation scheme with $m = 1$.

Recommendation	**banana banana bread**	**yes virginia there is a great meatloaf**
Explanation	creamy burrito casserole kittencal s italian melt in your mouth meatballs	pork chops yum yum best ever banana cake with cream cheese frosting

Table 20
Examples of real explanation generated through the LIRE approach with $exp - set - size = 2$.

Recommendation	**banana cake with cream cheese frosting**	**oven fried chicken chimichangas**
Explanation	to die for crock pot roast creamy cajun chicken pasta	kittencal's moist cheddar garlic oven fried chicken breast my family's favorite sloppy joes pizza joes

- The use of the LIRE approach in the context of the cooking recipe recommendation explanation, helps to effectively explain up to around 94% of the generated recommendations (Table 10). However, such fidelity values are only obtained for very small sizes of the explanation set associated to this method (see Section 3.4). In this direction, for $exp - set - size > 2$, the associated fidelity decreases quickly.
- The incorporation of the nutrition-aware criteria in the recommendation generation process leads to a decreasing in the associated fidelity in the simple content-based post-hoc explanation models (Tables 11 and 12). However, it is relevant that in the case of the Explanation Mining approaches, it keeps a similar fidelity value in relation to their counterparts without nutritional information (Tables 14 and 15). This result is important, and suggests that in this scenario the nutritional information can be added without affecting the explanation fidelity. Furthermore, the incorporation of nutritional information in the LIRE approach, is able to introduce an improvement on its associated fidelity.
- A direct comparison between the presented approaches (Fig. 7), suggests that the explanation methods that depend on recipe features (i.e. common ingredients), performed worse than the other approaches. However, at last the most appropriated method depends on the nature of the data.
- Finally, it was proved that even though there is a important overlapping between the explanations generated by the considered approaches (Fig. 8), each method tends to prioritized some kind of recipes in their generated explanations, based on their working principle (e.g. rules, similar features, learned surrogate model, etc.). Furthermore, the analysis of real explanations generated by each approach (Tables 17–20) corroborates this issue.

5. Conclusions and future works

The current paper has been focused on explaining recipe recommendations, as a particular RS domain that requires that the recommendation provides both enjoyable and nutritionally-appropriated items.

Taking as starting point this singularity, the research has discussed and adapted currently state-of-art recommendation explanation models, to the recipe recommendation domain. Such models were (1) a family of simple-to-explain recommendation algorithms for supporting explanations of black-box recommendation models, (2) the global explanation mining algorithm based on the discovering of global association rules, (3) the local explanation mining algorithm based on the discovering of local association rules, and (4) the LIRE approach, focused on model-agnostic explanations through learned surrogate models. For all scenarios, the goal was to explain why the recommended recipes are enjoyable, as well as controlling how the incorporation of the nutrition-aware criteria affects such explanation capability.

The developed experiments suggest that the explanation mining approaches obtain a higher fidelity in relation to the approach based on simple-to-explain methods. However, it was also shown that there is an important overlapping between the results associated to each method, and in other cases one method can act as a complement of the other. Finally, the incorporation of the nutrition-aware criteria does not notably affect the explanation capability of the proposals for minimum support values of the Explanation Mining approaches, and in the LIRE approach. Furthermore, in some scenarios it outperforms it.

The future work to the current research, will be the development of post-hoc explanation approaches for the group recommendation scenario [26], also focused on the cooking recipe domain. Furthermore, we will also explore how the management of the natural noise in user preferences in the food recommendation dataset [10], can affect the fidelity associated to the explanation methods considered in this work.

CRediT authorship contribution statement

Raciel Yera: Software, Validation, Formal analysis, Writing – original draft, Visualization. **Ahmad A. Alzahrani:** Conceptualization, Methodology, Validation, Project administration, Funding acquisition. **Luis Martínez:** Conceptualization, Methodology, Validation, Formal analysis, Writing – original draft, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The Deanship of Scientific Research (DSR) at King Abdulaziz University, Jeddah, Saudi Arabia has fund this project, under grant no. (Kep-15-611-42)

References

[1] G. Adomavicius, A.T. Tuzhilin, Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions, IEEE Trans. Knowl. Data Eng. 17 (6) (2005) 734–749.
[2] R. Yera, L. Martinez, Fuzzy tools in recommender systems: A survey, Int. J. Comput. Intell. Syst. 10 (1) (2017) 776–803.
[3] J. Ben Schafer, J.A. Konstan, J. Riedl, E-commerce recommendation applications, Data Min. Knowl. Discov. 5 (1-2) (2001) 115–153.
[4] J. Lu, D. Wu, M. Mao, W. Wang, G. Zhang, Recommender system application developments: a survey, Decis. Support Syst. 74 (2015) 12–32.
[5] R. Yera, L. Martínez, A recommendation approach for programming online judges supported by data preprocessing techniques, Appl. Intell. 47 (2) (2017) 277–290.
[6] R. Yera, A.A. Alzahrani, L. Martinez, A food recommender system considering nutritional information and user preferences, IEEE Access 7 (2019) 96695–96711.
[7] E. Carballo-Cruz, R. Yera, E. Carballo-Ramos, M. Betancourt, An intelligent system for sequencing product innovation activities in hotels, IEEE Latin Am. Trans. 17 (2) (2019) 305–315.

[8] M. Pazzani, D. Billsus, Content-based recommendation systems, in: The Adaptive Web, Vol. 4321, Springer-Verlag, 2007, pp. 325–341.
[9] M.D. Ekstrand, J.T. Riedl, J.A. Konstan, Collaborative filtering recommender systems, Found. Trends Hum. Comput. Interact. 4 (2) (2011) 81–173.
[10] R. Yera, M. Barranco, A. Alzahrani, L. Martinez, Exploring fuzzy rating regularities for managing natural noise in collaborative recommendation, Int. J. Comput. Intell. Syst. 12 (2) (2019) 1382–1392.
[11] J. Bobadilla, F. Ortega, A. Hernando, A. Gutiérrez, Recommender systems survey, Knowl.-Based Syst. 46 (2013) 109–132.
[12] F. Ricci, L. Rokach, B. Shapira, Recommender Systems Handbook, Springer, 2015.
[13] N. Tintarev, J. Masthoff, Explaining recommendations: Design and evaluation, in: Recommender Systems Handbook, Springer, 2015, pp. 353–382.
[14] Y. Zhang, X. Chen, Explainable recommendation: A survey and new perspectives, Found. Trends Inf. Retr. 14 (1) (2020) 1–101.
[15] G. Peake, J. Wang, Explanation mining: Post hoc interpretability of latent factor models for recommendation systems, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018, pp. 2060–2069.
[16] D. Shmaryahu, G. Shani, B. Shapira, Post-hoc explanations for complex model recommendations using simple methods, in: IntRS@ RecSys, 2020, pp. 26–36.
[17] C. Nóbrega, L. Marinho, Towards explaining recommendations through local surrogate models, in: Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, 2019, pp. 1671–1678.
[18] A. Chanson, N. Labroche, W. Verdeaux, Towards local post-hoc recommender systems explanations, in: Proceedings of the 23rd International Workshop on Design, Optimization, Languages and Analytical Processing of Big Data, 2021, pp. 41–50.
[19] M. Guo, N. Yan, X. Cui, S. Hughes, K. Al Jadda, Online product feature recommendations with interpretable machine learning, in: Proceedings of 2021 SIGIR E-Com, 2020, pp. 1–7.
[20] W. Cheng, Y. Shen, L. Huang, Y. Zhu, Incorporating interpretability into latent factor models via fast influence analysis, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, pp. 885–893.
[21] T. Ueta, M. Iwakami, T. Ito, A recipe recommendation system based on automatic nutrition information extraction, in: International Conference on Knowledge Science, Engineering and Management, Springer, 2011, pp. 79–90.
[22] C.-J. Lin, T.-T. Kuo, S.-D. Lin, A content-based matrix factorization model for recipe recommendation, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer, 2014, pp. 560–571.
[23] C. Trattner, D. Elsweiler, Investigating the healthiness of internet-sourced recipes: implications for meal planning and recommender systems, in: Proceedings of the 26th International Conference on World Wide Web, 2017, pp. 489–498.
[24] S. Barko-Sherif, D. Elsweiler, M. Harvey, Conversational agents for recipe recommendation, in: Proceedings of the 2020 Conference on Human Information Interaction and Retrieval, 2020, pp. 73–82.
[25] R. Burke, Hybrid recommender systems: Survey and experiments, User Model. User Adapt. Interact. 12 (4) (2002) 331–370.
[26] Y. Pérez-Almaguer, R. Yera, A.A. Alzahrani, L. Martínez, Content-based group recommender systems: a general taxonomy and further improvements, Expert Syst. Appl. 184 (2021) 115444.
[27] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, J. Riedl, GroupLens: an open architecture for collaborative filtering of netnews, in: Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work, ACM, New York, USA, 1994, pp. 175–186.
[28] C.C. Aggarwal, Model-based collaborative filtering, in: Recommender Systems, Springer, 2016, pp. 71–138.
[29] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, Computer 42 (8) (2009) 30–37.
[30] Y. Koren, Collaborative filtering with temporal dynamics, in: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2009, pp. 447–456.
[31] X. Luo, M. Zhou, Y. Xia, Q. Zhu, An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems, IEEE Trans. Ind. Inf. 10 (2) (2014) 1273–1284.
[32] Y. Shi, M. Larson, A. Hanjalic, List-wise learning to rank with matrix factorization for collaborative filtering, in: Proceedings of the Fourth ACM Conference on Recommender Systems, 2010, pp. 269–272.
[33] S. Rendle, W. Krichene, L. Zhang, J. Anderson, Neural collaborative filtering vs. matrix factorization revisited, in: Fourteenth ACM Conference on Recommender Systems, 2020, pp. 240–248.
[34] C. Trattner, D. Elsweiler, Food recommender systems: important contributions, challenges and future research directions, 2017, arXiv preprint arXiv:1711.02760.
[35] C. Trattner, D. Elsweiler, Food recommendations, in: Collaborative Recommendations: Algorithms, Practical Challenges and Applications, World Scientific, 2019, pp. 653–685.
[36] J. Freyne, S. Berkovsky, Intelligent food planning: personalized recipe recommendation, in: Proceedings of the 15th International Conference on Intelligent User Interfaces, 2010, pp. 321–324.
[37] M. Ge, M. Elahi, I. Fernaández-Tobías, F. Ricci, D. Massimo, Using tags and latent factors in a food recommender system, in: Proceedings of the 5th International Conference on Digital Health 2015, 2015, pp. 105–112.
[38] L. Yang, C.-K. Hsieh, H. Yang, J.P. Pollak, N. Dell, S. Belongie, C. Cole, D. Estrin, Yum-me: a personalized nutrient-based meal recommender system, ACM Trans. Inf. Syst. (TOIS) 36 (1) (2017) 1–31.
[39] M. Trevisiol, L. Chiarandini, R. Baeza-Yates, Buon appetito: recommending personalized menus, in: Proceedings of the 25th ACM Conference on Hypertext and Social Media, 2014, pp. 327–329.
[40] B. Ludwig, S. Meyer, J. Dietz, G. Donabauer, A. Pfaffelhuber, D. Elsweiler, Recommending better food choices for clinically obese users, in: Adjunct Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization, 2021, pp. 79–82.
[41] M. Chen, X. Jia, E. Gorbonos, C.T. Hoang, X. Yu, Y. Liu, Eating healthier: Exploring nutrition information for healthier recipe recommendation, Inf. Process. Manage. 57 (6) (2020) 102051.
[42] F. Pecune, L. Callebert, S. Marsella, A recommender system for healthy and personalized recipes recommendations, in: HealthRecSys@ RecSys, 2020, pp. 15–20.
[43] M. Khan, E. Rushe, B. Smyth, D. Coyle, Personalized, health-aware recipe recommendation: An ensemble topic modeling based approach, in: The 4th International Workshop on Health Recommender Systems (HealthRecSys 2019), Copenhagen, Denmark, 20 2019, 2019.
[44] D. Elsweiler, C. Trattner, M. Harvey, Exploiting food choice biases for healthier recipe recommendation, in: Proceedings of the 40th International Acm Sigir Conference on Research and Development in Information Retrieval, 2017, pp. 575–584.
[45] H. Cheng, M. Rokicki, E. Herder, The influence of city size on dietary choices, in: Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization, 2017, pp. 231–236.
[46] D. Bianchini, V. De Antonellis, N. De Franceschi, M. Melchiori, PREFer: A prescription-based food recommender system, Comput. Stand. Interfaces 54 (2017) 64–75.
[47] M. Harvey, B. Ludwig, D. Elsweiler, You are what you eat: Learning user tastes for rating prediction, in: International Symposium on String Processing and Information Retrieval, Springer, 2013, pp. 153–164.
[48] P. Forbes, M. Zhu, Content-boosted matrix factorization for recommender systems: experiments with recipe recommendation, in: Proceedings of the Fifth ACM Conference on Recommender Systems, 2011, pp. 261–264.
[49] Y. Zhang, G. Lai, M. Zhang, Y. Zhang, Y. Liu, S. Ma, Explicit factor models for explainable recommendation based on phrase-level sentiment analysis, in: Proceedings of the 37th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2014, pp. 83–92.
[50] X. Wang, X. He, F. Feng, L. Nie, T.-S. Chua, TEM: Tree-enhanced embedding model for explainable recommendation, in: Proceedings of the 27th International Conference on World Wide Web., 2018, pp. 1543–1552.
[51] C. Li, L. Quan, L. Peng, Y. Qi, Y. Deng, L. Wu, A capsule network for recommendation and explaining what you like and dislike, in: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019, pp. 275–284.
[52] J. Singh, A. Anand, Posthoc interpretability of learning to rank models using secondary training data, in: Proceedings of the SIGIR 2018 International Workshop on ExplainAble Recommendation and Search, 2018.
[53] J. McInerney, B. Lacker, S. Hansen, K. Higley, H. Bouchard, A. Gruson, R. Mehrotra, Explore, exploit, and explain: Personalizing explainable recommendations with bandits, in: Proceedings of the 12th ACM Conference on Recommender Systems., 2018, pp. 31–39.
[54] M.T. Ribeiro, S. Singh, C. Guestrin, Why should i trust you? explaining the predictions of any classifier, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 1135–1144.
[55] V. Arya, R.K. Bellamy, P.-Y. Chen, A. Dhurandhar, M. Hind, S.C. Hoffman, S. Houde, Q.V. Liao, R. Luss, A. Mojsilovic, et al., AI explainability 360: An extensible toolkit for understanding data and machine learning models, J. Mach. Learn. Res. 21 (130) (2020) 1–6.
[56] B. Sarwar, G. Karypis, J. Konstan, J. Reidl, Item-based collaborative filtering recommendation algorithms, in: Proceedings of the 10th International Conference on World Wide Web, 2001, pp. 285–295.

[57] C.D. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge University Press Cambridge, 2008.
[58] R. Agrawal, R. Srikant, et al., Fast algorithms for mining association rules, in: Proc. 20th Int. Conf. Very Large Data Bases, VLDB, Citeseer, 1994, pp. 487–499.
[59] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, Analysis of recommendation algorithms for e-commerce, in: Proceedings of the 2nd ACM Conference on Electronic Commerce, 2000, pp. 158–167.
[60] R. Tibshirani, Regression shrinkage and selection via the lasso, J. R. Stat. Soc. Ser. B Stat. Methodol. 58 (1) (1996) 267–288.
[61] B.P. Majumder, S. Li, J. Ni, J. McAuley, Generating personalized recipes from historical user preferences, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP, 2019, pp. 5976–5982.
[62] G. Shani, A. Gunawardana, Evaluating recommendation systems, in: Recommender Systems Handbook, Springer US, 2011, pp. 257–297.
[63] N. Hug, Surprise: A python library for recommender systems, J. Open Source Softw. 5 (52) (2020) 2174.
[64] A. Stasiak, Wow effect, in: Encyclopedia of Tourism Management and Marketing, Edward Elgar Publishing, 2022, pp. 1–3.