Evaluation of the User-Centric Explanation Strategies for Interactive Recommenders

Berk Buzcu[1,2(✉)], Emre Kuru[1], Davide Calvaresi[2], and Reyhan Aydoğan[1,3]

[1] Computer Science, Özyeğin University, Istanbul, Turkey
berk.buzcu@hevs.ch
[2] University of Applied Sciences and Arts Western Switzerland HES-SO Valais/Wallis, Sierre, Switzerland
[3] Interactive Intelligence, Delft University of Technology, Delft, Netherlands

Abstract. As recommendation systems become increasingly prevalent in numerous fields, the need for clear and persuasive interactions with users is rising. Integrating explainability into these systems is emerging as an effective approach to enhance user trust and sociability. This research focuses on recommendation systems that utilize a range of explainability techniques to foster trust by providing understandable personalized explanations for the recommendations made. In line with this, we study three distinct explanation methods that correspond with three basic recommendation strategies and assess their efficacy through user experiments. The findings from the experiments indicate that the majority of participants value the suggested explanation styles and favor straightforward, concise explanations over comparative ones.

Keywords: Explainable Recommendations · Explanation Strategies · User Studies

1 Introduction

In the rapidly evolving technological environment, our dependence on algorithm-powered recommendation systems for a variety of decision-making processes is growing. These systems are used in a wide range of applications, from suggesting content on movie streaming services to recommending products on e-commerce platforms. While these systems prioritize the selection and presentation of recommendations, they often overlook the user's curiosity about the rationale behind the recommendations. To address this, it is essential to engage users in an interactive communication setting. This allows users to delve deeper into the reasoning behind the recommendations, fostering a stronger understanding of the domain and satisfying their curiosity about the "why" behind the recommendations. This interactive setting necessitates methods for the system and users to express themselves, akin to a conversation between a sales assistant and a customer. Enhancing the recommender system's ability to express itself can make it more user-friendly, potentially leading to more effective outcomes.

D. Calvaresi et al. (Eds.): EXTRAAMAS 2024, LNAI 14847, pp. 21–38, 2024.
https://doi.org/10.1007/978-3-031-70074-3_2

In this context, our study aims to illuminate the workings of food recommendation systems, with a particular emphasis on their use in providing health-conscious dietary recommendations. Our primary goal is to enhance the trustworthiness and credibility of these food recommendation systems by equipping them with the ability to offer informative explanations for their recipe recommendations. To achieve this, we investigate recommendation strategies and their corresponding explanation generation strategies in two main categories: (i) model-agnostic explanations and (ii) model-intrinsic explanations. The former involves generating explanations by examining the results of the recommendation strategy using a separate model, a process known as "post-hoc" explanation generation. The latter uses a single model to generate both recommendations and explanations, making them "intrinsically explainable". Numerous studies have experimented with model-agnostic explanations due to the increasing predictive power of black-box models [2,8,19,28,29]. However, other research argues that if the generated explanations are not connected to the model's decision-making process, the system cannot be considered transparent [18,22], implying that explanations and recommendations should not be separated.

In light of this, our study employs and evaluates basic recommendation strategies from existing literature along with their corresponding explanation methods. Those explanation strategies can be categorized as *tree-based model agnostic*, *cluster-based model-agnostic* and *popularity-based model-intrinsic* explanation generation approaches. We scrutinize current approaches for explanation generation, incorporate them into our food recommender system, and compare the methods through user experiments, assessing user satisfaction and the effectiveness of the explanations. In the subsequent sections, we first outline the baseline strategy from the literature, followed by a detailed presentation of our proposed strategies.

2 Related Work

This section briefly overviews the literature on explainable recommender systems and different explanation strategies to persuade and convince users about given recommendations. Recent studies have emphasized incorporating explanations into recommendations to enhance transparency, trust, and acceptability. For instance, Tintarev and Masthoff investigate various aspects of explanations' impact, such as transparency, scrutability, trustworthiness, persuasiveness, effectiveness, efficiency, and satisfaction. Experiment results showed that in various domains especially in low investment domains, providing explanations is likely to improve these aspects of a recommender system [26]. All of these attributes enhance the system's reliability as supported by Gedikli *et al.* in which they assessed varying explanation attributes, including user satisfaction, efficiency, effectiveness, and trust, by evaluating different explanation styles in recommender systems via user study responses [10] (e.g., empirically they use response times to measure explanation effectiveness and subjectively user ratings for user satisfaction). Additionally, Herlocker *et al.* investigate the effects of

varying techniques used to explain collaborative-filtering recommendation methods. They follow the principle of collaborative filtering in their recommendation strategy and show ratings of similar users to the underlying user in the form of explanations [13]. They could not reach any concrete outcome according to their hypotheses that adding collaborative filtering based explanations to recommendation systems would improve the acceptance of that system and the performance of filtering decisions within the groups of ordinary users; however, they find out their system makes it more convenient for an expert to sympathize with a recommendation (e.g., the group of experts is more fond of the system with higher success in predictions of user acceptance).

Sharma and Cosley devised a framework to investigate the influence of social explanations (e.g., explanations that are similar to collaborative filtering in nature, where they are generated according to a grouping of users, and relating to other users contextually) within music recommenders [20]. They found that varying explanations might have different effects depending on the person. Similarly, Milliecamp explores the visual explanations within the music domain. They show that users react to explanations depending on their need for cognition, confidence, sophistication, and visualization literacy. Explanations boost confidence for those with a low need for cognition and speed up song judgments for those with higher musical sophistication. Users with lower visualization literacy tend to judge songs more quickly and precisely [16]. Furthermore, Pu and Chen develop an explanation interface to investigate the user experience advantages of using explanations for building trust and to assess whether system features can contribute to trust-related benefits [17]. They show that users prefer to re-use systems that offer explanations more often than those that do not, and users prefer a comparative explanation style where they get a broader view of available items and respective differences.

Besides, Balog, Radlinski, and Arakelyan present a set-based recommendation framework that utilizes interrelated features for generating explanations that account for conditional preferences [3]. Such as liking "Science Fiction" movies only when it's about "Space Exploration". Symeonidis *et al.* introduces a prototype for a movie recommender system designed to gauge user satisfaction via various explanation styles [24]. They point out that providing an explanation along with movie recommendations will increase the likelihood of a user estimating its movie ranking while also increasing the number of correct estimations to predict a user's favorite movie by boosting the user's confidence in providing information to the system. Guesmi *et al.* showed that users have different goals and may react differently to given explanations [12]. They claimed through their work that explanations are not a one-size-fits-all solution and that the explanations should be customized according to the characteristics of the users. In the following section, we survey the existing explanation mechanisms for recommenders.

In recent years, ample research has focused on developing model-agnostic (i.e., post-hoc) explanation generation techniques in machine learning, where explanations are generated after the predictions are made without requiring

modifications to the underlying model's architecture or training process. The goal is to improve transparency and interpretability while not decreasing the accuracy of the predictions. Post-hoc explanation generation models often leverage feature importance analysis [19], rule-based reasoning [28], gradient-based attribution [2], or surrogate models [29] to generate meaningful explanations that can shed light on the factors influencing the model's predictions. These explanations help stakeholders gain insight into how the given model makes its decisions, build trust, and facilitate error analysis [8].

Unlike post-hoc explanations, model-intrinsic techniques focus on generating explanations directly extracted from the internal mechanisms of the recommendation models. Thus, the generated explanations offer a solid understanding of the model's decision-making process. Rago *et al.* present a novel graphical framework, which establishes connections between items and their aspects within a recommendation system [18]. This framework utilizes Tripolar Argumentation Frameworks (TFs), an extension of the classical argumentation frameworks, to represent relationships among the items' features and the recommendations. TFs incorporate three distinct types of relations: positive, negative, and neutral, signifying whether an aspect of an item supports, attacks, or remains neutral to a recommendation. Through these relations, the users have the flexibility to customize their explanations based on their queries. If a user seeks an explanation as to why an item was recommended, the system can focus on highlighting the positive aspects of the item (supporters) within the Tripolar framework. Conversely, if a user wishes to understand why an item was not recommended, the system can emphasize the negative aspects (attackers). Shimizu, Matsutani, and Goto improve the state-of-the-art knowledge graph attention network (KGAT) by significantly decreasing its computation time, thus allowing more side information for generating explanations [22]. Here, KGAT represents the relationships between users, items, and their side information. Using attention weights to signify the importance of a node's or an edge's influence on a recommendation. When the model makes a recommendation, it can explain why it made that particular recommendation by highlighting the nodes or edges in the knowledge graph that received the highest attention weights.

Recent studies have succeeded in the realm of *counterfactual* explanations. These mechanisms aim to provide users with insightful explanations for predictions by generating counterfactual instances through"what-if" scenarios. It inquires whether a particular interaction or an attribute of the recommended item may influence any changes in the recommendation. Tan *et al.* extract aspect-aware explanations by looking for the minimal change in the recommended items' features such that the item would have not been recommended anymore; thereby finding the most crucial features for explanations [25], whereas Tran,Ghazimatin and Roy generate explanations by observing how much the recommendation changes if certain interactions were missing from the training dataset [27]. Mainly, they focus on whether their appreciation of an item would change if they did not experience any particular product before. Table 1 summarizes the related explanation approaches in the literature.

Table 1. Comparison Matrix of Explanation Approaches

Study	Explanation Type	Visual Text	Approach	Model AgnosticIntrinsic	Domain	Dynamic Static
Guesmi [12]	Personalized	Both	NLP	Model-agnostic	Articles	static
Buzcu [6]	Contrastive	Text	Decision Trees	Model-agnostic	Food	static
Balog [3]	Knowledge-Based	Text	Knowledge-Based	Model-intrinsic	Movies	static
Shimizu [22]	Example-Based Knowledge-Based	Text	Knowledge-Graph	Model-intrinsic	Products	static
Tan [25]	Counterfactual	Text	Neural Network	Model-intrinsic	Movies	dynamic
Rago [18]	Content-based	Text	Knowledge-Graph	Model-intrinsic	Movies	static
Tran [27]	Counterfactual	Text	Neural Network	Model-intrinsic	Products	dynamic
Our Approach	Contrastive Personalized Content-Based	Text	Clustering Random Forest	Both	Food	dynamic

3 Recommendation and Explanation Strategies

In this study, we adopt three basic recommendation strategies with aligned explanation approaches.

3.1 Baseline Recommendation and Explanations

Baseline Recommendation Generation is adopted from [4], which relies on filtering and scoring recommendations by considering underlying conditions and users' preferences. First, the system filters items (e.g., food recipe, movie) with respect to the user's constraints. In turn, the utilities of the remaining candidates are calculated through a scoring function. The items are sorted according to the computed utilities. The item with the highest utility, which was not recommended before, is selected as a recommendation, and the system retroactively generates an explanation in line with the recommendation's properties/features. For this baseline recommendation, Buzcu *et al.* introduce two types of explanation generation methods, which will be explained briefly below.

- **Item and User Explanations:** A decision tree is constructed from historical data in which recommendations are labeled with all users' decisions (i.e., accept or reject) in the user-based explanation approach. In contrast, items are labeled according to **the current user's constraints and feedback** in the item-based explanation generation approach. We can extract the importance of the features while building the decision tree. This approaches pick the most important three features to generate an explanation for the given recommendation.
- **Contrastive Explanations:** This type of explanations can be generated by referring a *contrastive item*, which is an item similar to the chosen one but fails to satisfy user constraints/preferences. For this purpose, the most similar item is selected from the aforementioned candidate set of items with the current recommendation. The features of the selected item with those of the recommendations are compared one by one. The features influencing the user satisfaction positively or negatively are used to build explanations that highlight the positive side of the recommendation while sending away the contrastive item by emphasizing its opposing sides.

3.2 Enhanced Baseline Recommendations and Cluster-Based Explanations

Recommendations is decided in a similar way to the aforementioned approach while explanations are generated by relying on the clustering approach proposed in [5]. The score function of the recommendation strategy is more comprehensive, thus we name it the Enhanced Baseline Recommendation. According to this approach, items are clustered with respect to the user's estimated preferences and desires. As usual, it is expected to have similar behavior or pattern in the same cluster, and someone can inquire which features distinguish those items in the same cluster from other clusters. In the proposed approach, each item is represented as a vector of evaluation criteria (e.g., the preference score for food recommendation). As illustrated in Fig. 1, a clustering algorithm is applied to determine distinguishable items concerning users' preferences and needs. For each cluster a separate classifier (Random Forest Classifier) is trained to detect whether or not the item belongs to underlying cluster. The feature importances, particularly the most important feature, can be extracted from this classifier to generate the explanations.

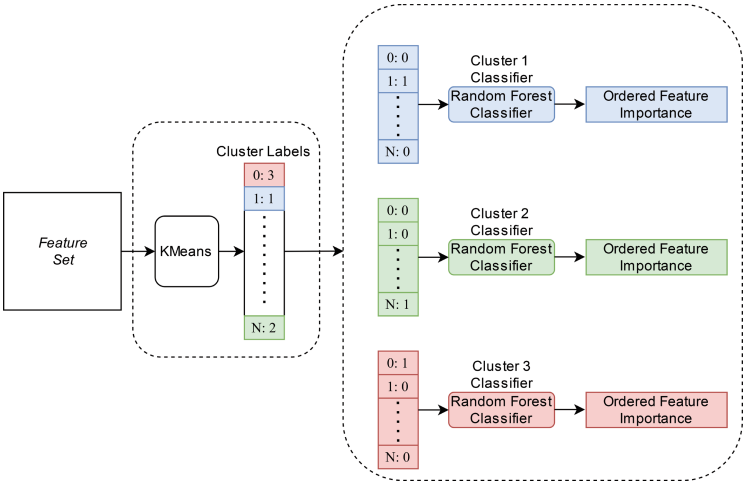


Fig. 1. Process of Cluster-Based Explanation [5]

While generating the contrastive explanations, the most similar item to the recommended item from another cluster is selected. By comparing the values of each feature of the contrastive item with those of the recommended item, positive and negative features are detected. While generating an explanation, the recommended item is promoted with the positively contrastive features, whereas negative features indicate why the system does not suggest the contrastive example.

3.3 Popularity-Based Explanations

The explanation generation techniques employed thus far have been model-agnostic, which could result in explanations that do not accurately reflect the actual decision-making process. This discrepancy arises because the system's overall outcomes may need to align better with the individual recommendations made by the system. In this approach, we utilize a historical dataset capturing the acceptance of past recommendations and a machine learning approach for their classification. In particular, a Random Forest Classifier is utilized to generate recommendations and explanations. Therefore, the proposed structure is inherently connected to the recommendation process, as both are generated from the same model. The Random Forest classifier is a valuable choice for popularity-based recommendation systems because it handles large-scale datasets and provides robust predictions while being explainable [23]. While making a recommendation, we calculate the probability estimates of each item and sort them based on the probability of acceptance (labeled as "1"). The recommendation algorithm combines this knowledge with personalization derived from previous sections and recommends the recipe with the highest level of probability of acceptance. On the other hand, we utilize the feature importance generated by the same Random Forest model in generating an explanation.

Algorithm 1 details the recommendation selection and explanation generation process. First, we train a Random Forest Classifier using our popularity-labeled data (Line 1). Then, we calculate the posterior probability for each class (Line 2). The class labels are acceptable (1) or unacceptable (0). We select the item with the maximum probability of acceptance (Line 2). This item's features are then compared according to the Random Forest model's feature importance vector. Finally, the most important feature is selected to construct the explanation (Line 3–4).

Additionally, we utilize the Popularity-based recommendation approach to generate contrastive explanations befitting the context. Algorithm 2 explains how contrastive explanations are generated accordingly. First, we extract the subset of recipes labeled as not recommended by the algorithm (Line 1). Then, we select the item with the minimum distance in the feature space to the current recommendation (Line 2). Finally, we choose the most important feature according to the feature importance of the Random Forest Classifier (Line 2).

4 Case Study: Food Recommendation

For baseline recommendation strategy, we use the original feature set in [4] to represent each food recipes as follows: *calorie count*, *fat amount*, *carbohydrates amount*, *fibers*, *preparation time*, *protein amount*, *preference score*. For our clustered-based recommendation strategy, we consider a vector of *preference score*, *health score*, *price score*, *time score*, and *taste score* with respect to the user profile since those scores captures users' preferences more comprehensively. For the popularity-based strategies, we have conducted an additional experimental study to gather a labeled dataset where if any user ever accepts them, the

recommended recipes are labeled as "1"; otherwise, they are labeled as "0". Note that the recipes that were never recommended are excluded from the dataset. We applied one-hot encoding to the features (e.g., *flavors of food*, *meal type*, *price*, and *cooking style*) to classify them accordingly. We utilized this data to train a Random forest classifier to predict whether a recommendation will be accepted or rejected by the user based on its popularity label in those experiments.

In the following part, we explain how aforementioned scores are calculated to suit the clustering technique used in our user experiments as features.

Algorithm 1. Popularity-based Recommendation & Explanation

Require:
F: Features of items;
I: All items,
I_p: Subset of popular recipes that were recommended in previous experiments; labeled "1" if they were ever accepted or "0" if they were recommended but not accepted;
Ensure: ϵ: Selected feature; r: Selected recommendation;
1: $randomForest \leftarrow RandomForestClassifier(I_p)$
2: $R \leftarrow randomForest.predictproba(I)$
3: $r \leftarrow \text{argmax}\, R[1]$
4: $\epsilon \leftarrow max(randomForest.featureImportance(F)$
5: return ϵ, r

Algorithm 2. Popularity-based Contrastive Explanation Selection

Require:
F: Feature set of Popularity;
P_u: Users feature preference weights;
I: Set of scored items;
i: Recommended item;
Ensure: ϵ: Explanation feature; r': Contrastive item;
1: $C \leftarrow R[0]$
2: $r' \leftarrow \text{argmin}_{c \in C}\, distance(c, r)$
3: $\epsilon \leftarrow \text{argmax}_{f \in F}$ featureImportance(r)
4: return ϵ, r'

Preference Score: To calculate the preference score of a user for the recipe dataset, we utilize a novel Active Learning framework [7] that is proven effective within our research project. The system first generates a diverse sample of recipes from the dataset. It asks the user to specify whether they like or dislike a given recipe. Afterward, the system shows a set of recipes to the user. The participants are asked to indicate the correct labels for the predictions made by the system by adjusting whether they like its respective features or not. This user feedback is utilized to generate synthetic data to enrich the user's preference data and increase the system's accuracy with a small dataset. Ultimately, the labeling

generated from this process is used to train a supervised learning model, Logistic Regression. We use the positive class probability by the model as the indicator for the user preference score for a given recipe, which corresponds to the likelihood of user's acceptance of a recipe.

Health Score: In order to calculate the health score, we follow intuition, where we take the user's personal information to calculate their nutritional needs in a day. The final *healthScore* is the mean of *calorieScore* and respective nutrient scores for each nutrition value of the recipe. To calculate the *calorieScore* we use the daily active metabolic rate (AMR) described in the literature [6] and the final score is derived as described in Eq. 1 where R corresponds to a recipe within the dataset. Essentially, we constrain the calorie score within the range of [0, 1], and we assume that a higher amount of calories is better as long as it is less than the active metabolic rate. For the nutrient scores, we use the nutrient density score [9] as described in Eq. 2. To simplify the healthiness decision, each nutrition is scored higher if they have a higher density per calorie except for the amount of fat, which is reversed $(1 - fatScore)$. The $w_{nutrient}$ corresponds to the predefined weight for each nutrient. Currently, all the weights are equal given our use-case does not define a distinction for nutrient importance. Finally, all scores are clamped to the $[0, 1]$ range, then averaged to derive the final *healthScore*.

$$calorieScore_R = \begin{cases} \frac{R_{calories}}{max_{calories}}, & \text{if } calories_R \leq AMR \\ 0, & \text{else} \end{cases} \tag{1}$$

$$nutrientScore_R = \frac{amount_{nutrient}(gr)}{calories_R(kcal)} \tag{2}$$

$$healthScore_R = \sum_{nutrient} w_{nutrient} * score_{nutrient} \tag{3}$$

Price Score : We first label the recipes within three classes; cheap, standard and expensive (labelled as $, $$, $$$ in order). We assume that the cheaper is better for a given food recipe, therefore, we assign these classes the scores of 1, 0.67, 0.33 respectively.

Time Score : The time scores are a summation of the recipes preparation and cooking time. For the calculation of this score, we assume that the quicker is better. Therefore, we apply max-normalization on the time of preperation in terms of minutes and reverse the order of scores for all the recipes. Thus, the quickest recipe is scored as "1".

Taste Score : The taste score corresponds to how well the flavor preferences of the user matches the flavor profile of a recipe. The flavor profile is comprised of the following tastes: *Savory*, *Bitter*, *Sour*, *Salty*, *Sweet* and *Spicy*, which are recognized as the main tastes the humans can distinguish [11]. Each food recipe holds boolean fields for the dimensions of a flavor profile. Table 2 shows an example recipe where each taste is labelled "1" if it is a part of the profile, or "0" if it is not.

Table 2. Flavors of a Tomato Soup

Recipe	Savory	Spicy	Sour	Salty	Sweet	Bitter
User Profile	1	1	1	1	0	0
Tomato Soup	1	0	1	1	0	0

Finally, we ask user to elicitate their desired tastes in the same form (e.g., they mark whether or not they want it in a boolean fashion). For each recipe, we apply the Jaccard Similarity [14] as described in the Eq. 4 to calculate the final score within the range of [0, 1].

$$tasteScore_R = \frac{flavors_R \cap flavors_{user}}{flavors_R \cup flavors_{user}}, \tag{4}$$

5 Evaluation

In order to thoroughly evaluate the proposed explanation generation strategies we conducted user experiments via a Web-based interface for food recommendations[1]. The experimental setup is presented in Sect. 5.1, consecutively, Sect. 5.2 reports and discusses the experimental results elaborately.

5.1 Experimental Setup

Prior to commencing the experiments, each participant was required to fill out a pre-survey and registration form, wherein they provided details about their gender, age, height, weight, level of physical activity, dietary preferences, and any allergies they might have. Additionally, they were asked to rank food related factors, and their taste preferences. The system utilizes this information to score the recipes respective to each participant's healthiness and preferences (Sect. 4). To evaluate the acceptability and effectiveness of the explanation-generation techniques proposed, we conducted a study involving participants experiencing three iterations of food recipes, each accompanied by three explanations. The system presents a recipe each time in the following order of recommendation strategies: (i) Baseline Recommendation (Sect. 3.1), (ii) Enhanced Baseline Recommendation and respective explanations compatible with the recommendation strategy (Sect. 3.2), and (iii) Popularity-Based Recommendation (Sect. 3.3):

- Baseline Recommendation: We use the following explanation methods: Item-based, User-based and Contrastive explanations (Sect. 3.1).
- Enhanced Baseline Recommendation (Sect. 3.2): We employ the following explanation methods: Cluster-based, Contrastive Cluster explanations, and Enhanced Item-based Sect. 3.1.

[1] The user experiments in this study was reviewed and approved by the Ethics Committee of Özyeğin University, and informed consent was obtained from all the experiment participants.

– Popularity Recommendation (Sect. 3.3): We apply the following explanation methods: Popularity-based, Contrastive Popularity, and Popularity User-based (Sect. 3.1) explanations.

Here, we adapted the User and Item-based explanation generation strategies described in Sect. 3.1 to generate explanations with the proposed recommendation strategies and their respective features. Afterward, the participants were asked to provide feedback on the perceived performance (effectiveness and convincability) of these explanations using a 5-point Likert scale. After completing the rating of the explanations, the system asks the user to choose their favorite recipe among the shown recipes with respective explanations. This design choice was based on research suggesting that users make better-informed decisions without experiencing excessive cognitive load when selecting from three items simultaneously [21]. As shown in the Fig. 2, the user can easily view nutritional information, recipe ingredients, and various explanations. However, the food's picture and detailed recipe information are not immediately visible, but the user can still access them by clicking the respective buttons, similar to earlier studies. After concluding the experiment, the participants are requested to complete a questionnaire consisting mainly of 5-point Likert scale questions. The questionnaire aims to assess their experiences with the explanations provided by the system. Participants are shown an explanation generated by the system and asked to respond to seven questions designed to gauge the effectiveness and success of the explanations they received.

In total, there are 80 participants (25 female, 55 male) with diverse backgrounds and age groups took part in the test (mean 24.70, min: 18 and max:59). The participants primarily consist of 51 bachelor's students, followed by 23 master's students, 3 doctoral students and 3 high school graduates. Meanwhile, the participants also reported 26 of them are sedentary (engaging in sports 0–1 days a week), 17 are lightly active (1–2 days), 24 are moderately active (2–3 days), 8 are active (4–5 days), and 5 are very active (5–7 days). They were first requested to order the importance of five criteria, relative to a given food recommendation: "Nutritional factors", "Past experience with taste", "How it looks", "Price of the ingredients", and "Cooking style". Participants were asked to rate various factors on a scale from 1 to 5, with 1 indicating the highest importance. The results show that a significant portion of the participants (specifically, 46%) considered their experience with the taste of such food to be the most critical factor influencing their cooking recipes. Additionally, 35% of the participants prioritized the healthiness of the food as their top concern. Conversely, 41% of the participants considered the time required to prepare the recipe the least important factor. In contrast, 28% of the participants rated the food price as the least significant factor in their decision-making process. The dataset used in the experiment is acquired from Diyetkolik [1] and it is comprised of 1382 recipes, where 210 of them were recommended to the users. In total, 125 of those recipes were accepted by the users cumulatively from previous studies [4]. Additionally, we have examined the average ratings between groups of genders and interest-

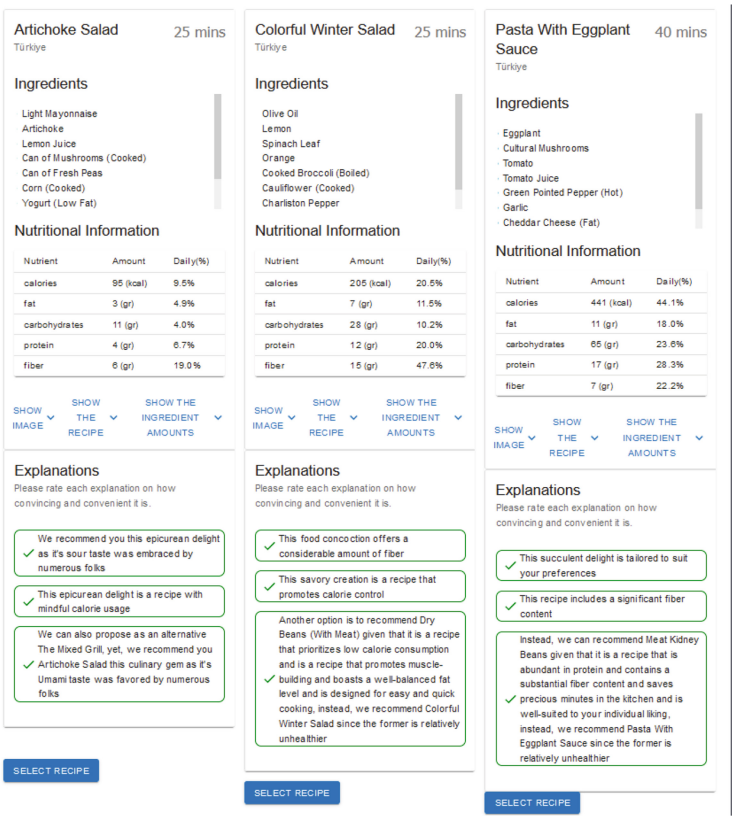


Fig. 2. User Interface for Recipe Selection Step

ingly, we found that female participants generally rated explanations higher than male participants on average (3.67 ± 0.488 versus 3.22 ± 0.564).

5.2 Experimental Results

The experimental setup is mainly comprised of experiment participants providing subjective input on explanations offered to recommendations.To recall, the process is outlined in Fig. 3.

We applied the Repeated Measures ANOVA statistical test rejected the null hypotheses, which revealed a significant difference among the types of explanations ($F=3.71$, $p=0.0003$). For furter analysis, the data, as determined by the Kolmogorov normality test, does not conform to a normal distribution, a crucial assumption for conducting pairwise T-tests. Consequently, we opted for the appropriate non-parametric alternative, the Wilcoxon signed-rank test [15], for our statistical tests. In all our analyses, we set the Confidence Interval (CI) to 0.95, corresponding to a significance level of $\alpha = 0.05$. Our test results com-

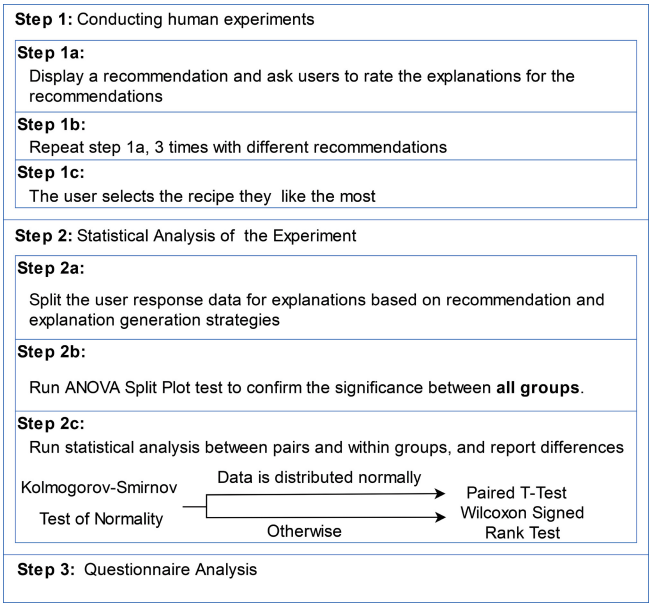


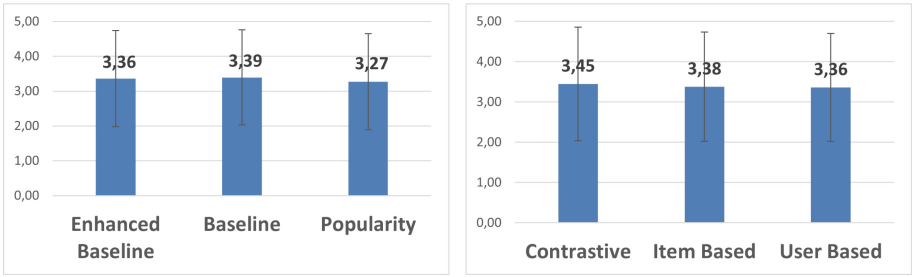
Fig. 3. Methodological Road Map for Results Analysis

prised of user responses to explanations are first analyzed by the recommendation strategy.

We conducted pairwise tests between these groups, yielding the following results: Enhanced Baseline vs. Popularity Recommendation ($p = 0.13$), Enhanced Baseline vs. Baseline Recommendation ($p = 0.44$), and Popularity vs. Baseline Recommendation ($\mathbf{p = 0.04}$). One could notice that the explanations generated along the Popularity-based recommendation have underperformed compared to the ones generated with the baseline recommendation strategy, as seen in Fig. 4a. Before drawing such conclusions, we must look into further analysis. Additionally, we categorized explanation generation techniques based on their underlying differences and the form of labelling strategy they utilized.

- Item Based: Methods that utilize an item's attributes on-line (Basic Cluster, Enhanced Item-based, and Item-based).
- User Based: Techniques that use historical data (user acceptance) as labeling (Popularity, Popularity User-based, and User-based).
- Contrastive: Explanations that are in the contrastive form (Contrastive Cluster, Contrastive Popularity, and Contrastive).

We conducted pairwise tests between these groups, leading to the following outcomes: Contrastive vs. Item-based ($\mathbf{p = 0.02}$), User-based vs. Item-based ($p = 0.84$), and Contrastive vs. User-based ($\mathbf{p = 0.02}$). Observing Fig. 4b, we note that the contrastive explanations under-performed slightly compared to the other methods statistically. We notice a trend where the participants prefer



(a) Recommendation (b) Explanation Generation

Fig. 4. Avg. Scores of Explanations Grouped per Technique

explanations based on the attributes of the recommended item more. The participants did not appreciate both contrastive explanations and popularity-based metrics, potentially pointing to the fact that users care more about facts on their recommendation than comparative explanations.

Subsequently, we conducted pairwise tests to compare different types of explanation generation techniques within the same recommendation strategy individually as follows:

Popularity-Based Recommendation:

- Popularity vs. Popularity User-based ($p = 0.70$)
- Popularity vs. Contrastive Popularity ($p \leq \mathbf{0.0001}$)
- Contrastive Popularity vs. Popularity User-based ($p \leq \mathbf{0.0001}$)

Enhanced Baseline Recommendation:

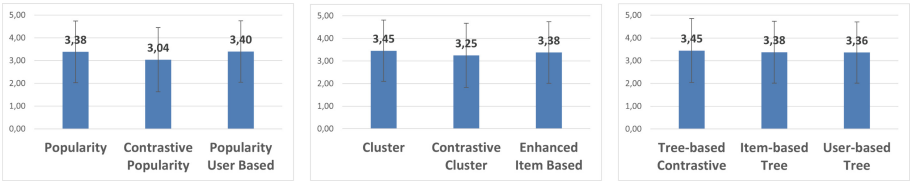
- Cluster vs. Enhanced Item-based ($p = 0.78$)
- Cluster vs. Contrastive Cluster ($p = 0.13$)
- Contrastive Cluster vs. Enhanced Item-based ($p = 0.16$)

Baseline Recommendation:

- Item-Based vs. User-Based ($p = 0.92$)
- Contrastive vs. Item-Based ($p = 0.44$)
- Contrastive vs. User-Based ($p = 0.31$)

These results provide insights into the comparative performance of explanation techniques within each recommendation strategy. The significant p-values (highlighted) indicate noteworthy differences deserving further investigation. We note that only in popularity group that the contrastive explanations significantly under-perform as shown in Fig. 5. In other recommendation groups, there is no significant results observed.

Additionally, the distribution of accepted recipes are (i) Baseline recommendation 38%, (ii) Enhanced baseline 32% and (iii) Popularity-based 30%. Our take-aways may be further supported by this outcome given the simplest method

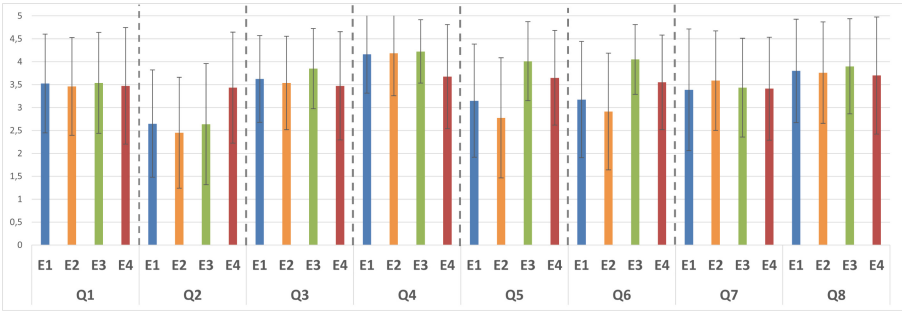


(a) Popularity Based (b) Cluster Based (c) Baseline

Fig. 5. Avg Scores of Explanations by Recommendation

of recommendation seems to be favored more than the others. However, this may be just a result of combination of explanations and the food recipe being more fitting to the users.

Moreover, we conducted an analysis of user responses to the post-experiment survey to assess their perceptions of the given explanations. Since each participant was given each form of explanation during the experiment, and the survey questions as well as the provided examples of the types were identical for all participants, we employed a within-subjects statistical comparison test. Figure 6 shows the average responses to the questionnaire questions, as well as the questions and respective explanations.



Id	Label
Q1	This type of explanation for recommendations has helped me choose the most convenient recipe.
Q2	This type of explanation for recommendations were too detailed.
Q3	This type of explanation displayed during the interaction were satisfactory.
Q4	This type of explanation for recommendations were clear and easy to understand.
Q5	This type of explanation were sufficient to make an informed decision for healthiness.
Q6	This type of explanation were realistic in terms of healthiness of given recipes.
Q7	This type of explanation let me know how convenient the recipe is.
Q8	Rate your appreciation of the idea of receiving this type of explanations in addition to recommendations.
E1	Popularity-based explanation sample
E2	Cluster-based explanation sample
E3	Baseline explanation sample
E4	Contrastive explanation sample

Fig. 6. Questionnaire Responses About Explanations

Table 3 shows the Wilcoxon paired test results for each type of explanation for each question. We observe significant differences between pairs of explanations on Q2, Q3, Q4, Q5 and Q6 whereas there is no significance within explanation pairs for Q1, Q7 and Q8. The Q3 tells us that baseline explanations were the most satisfactory explanations, it is also the simplest explanation generation method. One could draw the conclusion that the participants favor simplistic methods over complicated ones. This finding is further supported by Q4, where contrastive explanations were found too complicated and the fact that they were rated lowest among the explanation types. The simpler explanations were found to be more effective in coming to healthy decisions, as it is seen from Q5. Finally, Q8 shows us that the users would still use this system despite it's short-comings with no significant difference among pairs of explanations.

Table 3. Pairwise Wilcoxon Test Results

P-Value	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
E1 vs E2	0.565	0.045	0.549	0.679	**0.007**	0.063	0.294	0.909
E1 vs E3	0.986	0.683	**0.010**	0.573	$\leq$ **0.000**	$\leq$ **0.001**	0.902	0.579
E1 vs E4	0.878	$\leq$ **0.001**	0.426	**0.001**	**0.003**	**0.012**	0.859	0.635
E2 vs E3	0.524	0.053	**0.034**	0.621	$\leq$ **0.001**	$\leq$ **0.001**	0.498	0.475
E2 vs E4	0.855	$\leq$ **0.001**	0.638	$\leq$ **0.001**	$\leq$ **0.001**	$\leq$ **0.001**	0.284	0.751
E3 vs E4	0.656	$\leq$ **0.001**	**0.017**	$\leq$ **0.001**	**0.023**	**0.001**	0.641	0.228

6 Conclusion

In conclusion, this research contributes to the ongoing dialogue about incorporating explanation generation strategies into recommendation systems, especially those focused on health-aware recommendations. As we search to enhance the transparency and effectiveness of recommendation systems, we find that user-centrism, simplicity, and clarity are crucial for effective explanations. Despite these findings, it is important to acknowledge that the effectiveness of explanation strategies may vary depending on the specific user, rather than the collective user opinion on recommendation items guiding explanations. This study does not particularly focus on a group of individuals and it involves participants from diverse backgrounds and dietary preferences. Such diversity could affect their perspectives on the styles of explanations. Future studies could delve deeper into fine-tuning the explanation strategies toward user profiles and preferences, where we offer different styles of explanations at varying degrees to diverse profiles of users.

References

1. Yemek tarifleri (2022). https://www.diyetkolik.com/yemek-tarifleri/. Accessed 1 Jan 2022
2. Ancona, M., Ceolini, E., Öztireli, A.C., Gross, M.H.: A unified view of gradient-based attribution methods for deep neural networks (2017)
3. Balog, K., Radlinski, F., Arakelyan, S.: Transparent, scrutable and explainable user models for personalized recommendation. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 265–274 (2019)
4. Buzcu, B., et al.: Towards interactive explanation-based nutrition virtual coaching systems. Auton. Agent. Multi-Agent Syst. **38**(1), 5 (2024). https://doi.org/10.1007/s10458-023-09634-5
5. Buzcu, B., et al.: User-centric explanation strategies for interactive recommenders. In: The 23rd International Conference on Autonomous Agents and Multi-Agent Systems (2024)
6. Buzcu, B., Varadhajaran, V., Tchappi, I., Najjar, A., Calvaresi, D., Aydoğan, R.: Explanation-based negotiation protocol for nutrition virtual coaching. In: Aydoğan, R., Criado, N., Lang, J., Sanchez-Anguix, V., Serramia, M. (eds.) International Conference on Principles and Practice of Multi-Agent Systems, vol. 13753, pp. 20–36. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-21203-1_2
7. Cantürk, F., Aydoğan, R.: Explainable active learning for preference elicitation, p. 25 (2023). https://doi.org/10.21203/rs.3.rs-3295326/v1
8. Cemiloglu, D., Catania, M., Ali, R.: Explainable persuasion in interactive design. In: 2021 IEEE 29th International Requirements Engineering Conference Workshops (REW), pp. 377–382 (2021)
9. Drewnowski, A., Fulgoni, V.L.: Nutrient density: principles and evaluation tools123. Am. J. Clin. Nutr. **99**(5), 1223S–1228S (2014). https://doi.org/10.3945/ajcn.113.073395, https://www.sciencedirect.com/science/article/pii/S0002916523050748
10. Gedikli, F., Ge, M., Jannach, D.: Understanding recommendations by reading the clouds. In: Huemer, C., Setzer, T. (eds.) EC-Web 2011. LNBIP, vol. 85, pp. 196–208. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23014-1_17
11. Gravina, S.A., Yep, G.L., Khan, M.: Human biology of taste. Ann. Saudi Med. **33**(3), 217–222 (2013). https://doi.org/10.5144/0256-4947.2013.217, https://www.annsaudimed.net/doi/abs/10.5144/0256-4947.2013.217
12. Guesmi, M., et al.: Explaining user models with different levels of detail for transparent recommendation: a user study. In: Adjunct Proceedings of the 30th ACM Conference on User Modeling, Adaptation and Personalization, pp. 175–183 (2022)
13. Herlocker, J., Konstan, J., Riedl, J.: Explaining collaborative filtering recommendations. In: Proceedings of the ACM Conference on Computer Supported Cooperative Work, pp. 241–250 (2001). https://doi.org/10.1145/358916.358995
14. Jaccard, P.: The distribution of the flora in the alpine zone.1. New Phytol. **11**(2), 37–50 (1912). https://doi.org/10.1111/j.1469-8137.1912.tb05611.x, https://nph.onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-8137.1912.tb05611.x
15. Lazar, J., Feng, J., Hochheiser, H.: Research Methods in Human-Computer Interaction (2017)
16. Millecamp, M., Htun, N.N., Conati, C., Verbert, K.: To explain or not to explain: the effects of personal characteristics when explaining music recommendations. In: Proceedings of the 24th International Conference on Intelligent User Interfaces, p. 397–407 (2019)

17. Pu, P., Chen, L.: Trust building with explanation interfaces. In: International Conference on Intelligent User Interfaces, Proceedings IUI, vol. 2006, pp. 93–100 (2006). https://doi.org/10.1145/1111449.1111475
18. Rago, A., Cocarascu, O., Bechlivanidis, C., Lagnado, D., Toni, F.: Argumentative explanations for interactive recommendations. Artif. Intell. **296**, 103506 (2021)
19. Saarela, M., Jauhiainen, S.: Comparison of feature importance measures as explanations for classification models. SN Appl. Sci. **3** (2021)
20. Sharma, A., Cosley, D.: Do social explanations work? Studying and modeling the effects of social explanations in recommender systems. WWW 2013 - Proceedings of the 22nd International Conference on World Wide Web, pp. 1133—1144 (2013)
21. Shimazu, H.: ExpertClerk: Navigating shoppers' buying process with the combination of asking and proposing. In: Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 2, p. 1443–1448. IJCAI 2001, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2001)
22. Shimizu, R., Matsutani, M., Goto, M.: An explainable recommendation framework based on an improved knowledge graph attention network with massive volumes of side information. Knowl. Based Syst. **239**, 107970 (2022)
23. Speiser, J.L., Miller, M.E., Tooze, J., Ip, E.: A comparison of random forest variable selection methods for classification prediction modeling. Expert Syst. Appl. **134**, 93–101 (2019). https://doi.org/10.1016/j.eswa.2019.05.028, https://www.sciencedirect.com/science/article/pii/S0957417419303574
24. Symeonidis, P., Nanopoulos, A., Manolopoulos, Y.: MoviExplain: a recommender system with explanations. In: Proceedings of the Third ACM Conference on Recommender Systems, p. 317–320. RecSys 2009, Association for Computing Machinery, New York, NY, USA (2009)
25. Tan, J., Xu, S., Ge, Y., Li, Y., Chen, X., Zhang, Y.: Counterfactual explainable recommendation. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management, pp. 1784–1793 (2021)
26. Tintarev, N., Masthoff, J.: Explaining recommendations: design and evaluation. In: Ricci, F., Rokach, L., Shapira, B. (eds.) Recommender Systems Handbook, pp. 353–382. Springer, Boston, MA (2015). https://doi.org/10.1007/978-1-4899-7637-6_10
27. Tran, K.H., Ghazimatin, A., Saha Roy, R.: Counterfactual explanations for neural recommenders. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1627–1631 (2021)
28. Xu, Y., Collenette, J., Dennis, L., Dixon, C.: Dialogue-based explanations of reasoning in rule-based systems. In: 3rd Workshop on Explainable Logic-Based Knowledge Representation (2022)
29. Zhu, X., Wang, D., Pedrycz, W., Li, Z.: Fuzzy rule-based local surrogate models for black-box model explanation. IEEE Trans. Fuzzy Syst. **31**(6), 2056–2064 (2023)