PERSONALIZED EXPLAINABLE ARTIFICIAL INTELLIGENCE FOR DECENTRALIZED AGENTS WITH HETEROGENEOUS KNOWLEDGE

D3.2b
Scientific papers focusing on conflict resolution
[M27]

Version: 1.0
Last Update: 11/04/2024
Distribution Level: CO

Distribution levels*
PU = Public;
RE = Restricted to a group within the given consortium;
PP = Restricted to other program participants (including commission services);

Project supported by

CO = Confidential, only for the members of the EXPECTATION Consortium (including commission services);

The EXPECTATION Project Consortium groups Organizations involved:

Partner Name	**Short name**	**Country**
University of Applied Sciences and Arts Western Switzerland HES-SO Institute of Information Systems	HES-SO	Switzerland
University of Bologna / Dep. Of Computer Science and Engineering	UNIBO	Italy
University of Luxembourg, Department of Computer Science	UNILU	Luxembourg
Özyeğin University	OZU	Turkey

Document Identity

Creation Date: Last Update:	11/04/2024 11/04/2024

Revision History

Version	**Edition**	**Author(s)**	**Date**
1	1	All PIs	11/04/2024
Comments:	Creating front page, papers are already published		

Project supported by

Autonomous Agents and Multi-Agent Systems (2024) 38:5
https://doi.org/10.1007/s10458-023-09634-5

Towards interactive explanation-based nutrition virtual coaching systems

Berk Buzcu[1,5] · Melissa Tessa[2] · Igor Tchappi[4] · Amro Najjar[3,4] · Joris Hulstijn[4] · Davide Calvaresi[5] · Reyhan Aydoğan[1,6,7]

Accepted: 15 December 2023

Abstract
The awareness about healthy lifestyles is increasing, opening to personalized intelligent health coaching applications. A demand for more than mere suggestions and mechanistic interactions has driven attention to nutrition virtual coaching systems (NVC) as a bridge between human–machine interaction and recommender, informative, persuasive, and argumentation systems. NVC can rely on data-driven opaque mechanisms. Therefore, it is crucial to enable NVC to explain their doing (i.e., engaging the user in discussions (via arguments) about dietary solutions/alternatives). By doing so, transparency, user acceptance, and engagement are expected to be boosted. This study focuses on NVC agents generating personalized food recommendations based on user-specific factors such as allergies, eating habits, lifestyles, and ingredient preferences. In particular, we propose a user-agent negotiation process entailing run-time feedback mechanisms to react to both recommendations and related explanations. Lastly, the study presents the findings obtained by the experiments conducted with multi-background participants to evaluate the acceptability and effectiveness of the proposed system. The results indicate that most participants value the opportunity to provide feedback and receive explanations for recommendations. Additionally, the users are fond of receiving information tailored to their needs. Furthermore, our interactive recommendation system performed better than the corresponding traditional recommendation system in terms of effectiveness regarding the number of agreements and rounds.

Keywords Explainable AI · Recommender systems · Interactive · Nutrition virtual coach

1 Introduction

Approximately 63% of all deaths worldwide are attributed to non-communicable diseases such as cardiovascular diseases, chronic respiratory diseases, and diabetes.[1] The World Health Organization emphasizes that these diseases can be prevented by addressing common risk factors, such as unhealthy nutrition habits and diets. However, personal preferences, cultural and religious constraints, and taste heavily affect individuals' habits. Tasty—yet unhealthy components—are increasingly hidden in a wide range of processed

[1] https://www.who.int/news-room/fact-sheets/detail/noncommunicable-diseases.

Extended author information available on the last page of the article

Published online: 20 January 2024

food items. Therefore, society needs guidance on making suitable and sustainable dietary choices [8, 12, 44]. To counter the unhealthy trend, food recommender systems—assisting individuals in recipe selection have gained popularity [8, 44]. The need for these systems can be attributed to increased globalization, leading to greater availability and variety of food, as well as the prevalence of ultra-processed food, contributing to metabolic and overweight issues [18]. Although numerous recipes are freely accessible (i.e., via many online collectors), determining the "best" recipe for a *specific* individual in a *given* situation can be remarkably complex. Indeed, it involves managing a wide range of possibilities while considering bounding variables such as allergens, nutritional values, personal requirements, calorie intake, historical data, and momentary preferences. Consequently, there is a need for a personalized support system. Nutrition virtual coaches (NVCs) are systems that aim to recommend recipes that align with users' specific needs and preferences while considering their health and long-term needs [43].

NVCs cater to various goals, including muscle gain, weight loss, and management of nutrition-related diseases such as obesity.[2] The underlying objective is to provide users with constructive "educational" support, gradually reducing their reliance on NVCs. Existing solutions, both from research and industry, have attempted to address these goals. However, they often lack transparency and clarity, leading to a lack of trust and effectiveness [8]. To enhance transparency and, henceforth, effectiveness, Explainable AI (XAI) techniques have been employed in various domains, such as transportation, fleet management, and neurosciences [14, 33]. Moreover, some studies have proposed semantic models [34] and incorporated negotiation techniques to guide users towards desired quality of life goals [28].. While these efforts have contributed to the field of recommender systems, to the best of our knowledge, there is currently no existing system that fully qualifies as an "explainable" Nutrition Virtual Coach (NVC) which is effectively an agent that provide recommendations, explain them to the user, and engage in interactive discussions to foster desired behavioral changes. Engaging the user in interactive (back-and-forth) communication is crucial as it allows the user to dive into the concept and build a more solid and backed-up knowledge/awareness that undoubtedly boosts information retention. Such mechanisms can assume a rather simplistic—yet effective—form of feedback [27]. Building on that, verifying/fixing misunderstandings and elaborating on follow-up questions becomes more feasible (from a designer/developer perspective) and easy to handle (from a user perspective).

This work builds upon the protocol described in [6], and it extends it by introducing a more sophisticated/dynamic explanation generation strategy consisting of decision trees in the form of Item and User based trees to generate explanations retroactively to recommendation selection. Moreover, we have improved the user interface, leveraging the feedback coming from the user study conducted in [6]. Finally, we have extended the comparative evaluation of the proposed system using a simple health score calculation, with a multi-criteria additive utility function for recipe selection and an Web Ontology Language (OWL) based ontology database to classify users and recipe ingredients.

Our main assumption is that people can have different preferences (i.e., taste over healthiness or vice-versa). However, recommender systems, in prioritizing recommendations aligned with predefined goals, may sometimes overlook specific user preferences, leading to "conflicts" between user desires and system objectives. For instance, a user

[2] https://www.cdc.gov/chronicdisease/resources/publications/factsheets/nutrition.html.

seeking tasty yet conversely unhealthy food may clash with a system focused on promoting a healthy lifestyle. The system developers in that case must delicately balance meeting the system goals while delivering a personalized experience. Therefore, to address these conflicts, we model the resolution as a negotiation in a dialogical setting where the system concedes by making recommendations more fitting to the user profile than its own goals (healthiness). We classified the participants according to their priorities (obtained via a pre-experiment survey). Moreover, we assessed the protocol with individuals characterized by various backgrounds in online experimental settings consisting of a pre-experiment survey, two sessions (static vs. interactive), and a concluding post-experiment survey to question the participants about their experience with the different settings.

The rest of this paper is organized as follows. Section 2 presents the related work. Section 3 presents the explainable argumentation negotiation module for NVC. Section 4 evaluates and discusses the obtained results. Finally, Sect. 5 concludes the paper and outlines future works.

2 Related work

This section briefly overviews the literature on food recommender systems, focusing on conventional systems and their evolution to embrace explainable and interactive recommendations.

2.1 Conventional food recommendation

In 1986, Hammond et al. [21] developed one of the earliest food recommender systems. It is named CHEF and leverages case-based planning to replace or improve food items within recipes. It requires a substantial initial knowledge base, extensive pre-processing, and the creation of (backup) plans for each recipe. More recently, in 2010, Freyne and Berkovsky [16] implemented recommender algorithms, such as collaborative filtering (CF) and content-based (CB) approaches, to recommend recipes. The study concluded that incorporating ingredient weights within CF and CB improved prediction accuracy. In turn, Ge et al. [17] introduced the concept of personalization in food recommendations, prioritizing health over taste. Chi et al. [11] focused on recommending food for individuals with chronic conditions (i.e., kidney diseases) using an Ontology Web Language (OWL) ontology integrating health-relevant aspects. Chen et al. [10] proposed a generalized framework for healthy recommendations, explicitly targeting the modification of unhealthy recipes. The authors introduced a deep learning-based method called IP-embedding to match recipes with desired ingredients, creating a pseudo recipe that meets the requirements and then matching it with healthy ingredients and real recipes using the mean squared error (MSE) metric. Similarly, Teng et al. [39] developed a point-wise comparison metric to understand how to transform recipes into more healthier ones, using ingredient substitutions for healthier alternatives. Elsweiler et al. [1] addressed ingredient and food substitution, metricizing nutritional values to encourage users to prefer healthier options. Overall, food recommendation approaches often rely on factors such as recipe content (e.g., ingredients) [13, 15, 40], user behavior history (e.g., eating history) [32, 46], and dietary preferences [32, 45].

2.2 Towards explainable recommendation systems

Conventional food recommendation approaches are mostly "one-shot", offering the user minimal (if any) possibilities to interact. However, with the advent of explainable technologies, that aim for predictors and classifiers that show transparency, understandability, and inspectability in order to boost trust [4], recommender systems are expected to provide explanations for their recommendations [13, 15, 20, 46], allowing users to justify, control, and discover new aspects of the suggested outcomes [32, 45, 47]. Along this line, Padhiar et al. [34] proposed a food recommender system that generates explanations based on a knowledge-based ontology. However, the explanatory system only attempts to explain a given recommendation via different methods, with no dialogue option: no way for the user to reply or interact. Samih et al. [36] further explored this concept by developing a knowledge-based explainable recommender system that makes use of a probabilistic soft-logic framework to generate explanations. Lawo et al. [28] aimed to enhance the interaction between users and virtual assistants by incorporating a cluster of consumers with ethical and social priorities into the recommendation process and considering their feedback and preferences.

Finally, recommendation systems have been employed in the nutrition domain for some time, with objectives ranging from promoting health, sustainability, and finding combinations of ingredients that taste well. Recent studies have emphasized the importance of incorporating explanations into recommendations to enhance transparency, trust, and acceptability. Although explanations in food recommender systems are still not fully widespread, some approaches (or combinations of them) are gaining attention. In the following section, we survey existing explanation mechanisms, which could be adopted by food recommendation systems.

2.3 Post-hoc explanation generation mechanisms

In recent years, there has been ample research within the Machine Learning literature, focused on developing techniques for post-hoc explanation generation in various domains. These techniques are designed to explain the predictions made by complex black-box models. They operate "post-hoc," meaning they generate explanations after the main model has made its predictions, without requiring modifications to the underlying architecture or training process. The goal is to improve transparency and interpretability by providing human-understandable justifications for the model's decisions. Post-Hoc Explanation Generation models leverage techniques such as feature importance analysis [35], rule-based reasoning [50], gradient-based attribution [3], or surrogate models [51] 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 arrives at its decisions, builds trust, and facilitates error analysis, making them valuable tools for practical applications and model understanding [9].

We can distinguish various strategies to generate explanations. Note that these classes are not mutually exclusive but are often overlapping. So a user-centred explanation can also be content-based. The most suitable forms of explanations presented in the literature to be generated for food recommendations, can be classified as follows:

- *User-centered explanation:* The generated explanations are meant to assist users in achieving their goals. Sovrana and Vitali [38] emphasize that users are satisfied with the explanations if they are guided in answering the questions about the process of fulfilling their goals. An explanation such as "*we recommend you the following food recipe to lose weight since it has low fat and rich fiber*" could be considered an instance of this explanation type. It implicitly answers what is necessary to lose weight, which aligns with the hypothetical user's goals.
- *Knowledge-based explanations:* These explanations are generated by inferring some formal rules and facts in the knowledge base. For instance, a recommendation engine can offer a camera with less memory and resolution by referring to the rule that states "Less memory $\wedge$ lower resolution $\rightarrow$ cheap" [41]. Such rules need to be given to the system, and they can be derived from a decision tree modeling the system's or user's behavior. In other words, decision trees could be utilized to learn why the underlying decision is made from the data, and the rules extracted from the constructed decision tree can give insights on how the system works to the user as an explanation [19].
- *Example-based explanations:* Based on historical data or previous experiences, a system can generate some explanations by generalizing past behaviors/patterns for a given new situation [48]. For example, assume that a food recipe consisting of sugar-free ingredients was recommended to a diabetic person by a recommender system that recommends food to ill people, and the results were satisfactory. If a new diabetic person joins the system, it might generate the following explanation alongside its recommendation "Diabetic people are often satisfied with this food recipe with sugar-free ingredients.".
- *Content-based explanations:* Inspired by the content-based recommendation approach, the system can analyze the features of the items appreciated by a particular user and extract the preferred values for those features to explain the recommended item to that user [41]. For instance, the system can generate an explanation such as "This food recipe contains mozzarella, so you might like it." if the user previously liked the food recipes that contain mozzarella specifically.
- *Contextual explanations:* External factors affecting the decision could be used to generate such explanations. For instance, "Today fish is fresh. It has just arrived. Therefore, I recommend creamy salmon pasta." [34].
- *Contrastive explanations:* A recent review by [31] provides empirical evidence supporting the practical utility of everyday contrastive explanations, "comparing a certain phenomenon with a hypothetical one" [48]. While asking about a certain choice, someone may think of alternatives and wonder why those were not recommended with respect to the given one. Contrastive explanations focus on the difference between the current choice and alternative ones. For instance, "We were going to recommend you a healthier option, which is Turkish Salad instead of American Salad that contains a substantially higher amount of fats.".
- *Counterfactual explanations:* Like contrastive explanations, counterfactual explanations focus on the differences between alternative options. However, these explanations rely on hypothetical factors instead of factual factors [34]. For instance, "If you did not have an allergy to seafood, I would recommend you a salmon salad. However, now I have to recommend you a turkey salad.".

The first three types in the list above, namely user-centred, knowledge-based and example-based, differ in the type of argument to convince the user. The first relates to what the user previously stated as preference or goal, whereas the second refers to external knowledge,

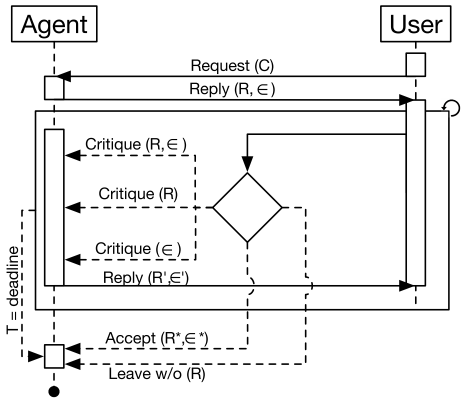


Fig. 1 FIPA description of the negotiation protocol where C corresponds to user constrains, R is a recipe recommended by the agent and ϵ is an explanation that comes with the recipe

in our case from a food expert. The third refers to an analogy with what other people in a peer group have chosen. By contrast, the fourth type, content-based explanations, is based on features derived from the recommendation itself. One can match those features with user preferences, external knowledge, or examples from peers, to make an argument, as mentioned before.

The fifth type picks contextual factors to focus the argument upon. In our case, the time of day determines the type of meal (breakfast, lunch, dinner). In that sense, most of our explanations are implicitly contextual. The final two types of explanations focus on the fact that explanation should help people make a choice among two or more alternatives. A contrastive explanation signals the differences between existing alternatives, whereas a counterfactual explanation signals the differences between the given selection criteria and other potential, but non-actual, selection criteria. In implementation, we have to make a combination of explanation generation strategies, and use those arguments that are most convincing in a given situation. For example, if a knowledge-based explanation fails to convince the user, an explanation based on examples from the same group of users, may work better. There are also interesting cultural differences. A user-based explanation may work better in an individualistic culture, for example. The proposed combination of strategies targeted to the food domain is novel, even if the component strategies (user-centered; content-based) have been used before.

3 Proposed approach

Our earlier study presented in [6] proposes a design of an interaction protocol for explainable NVC. In particular, it provides recommendations for recipes seeking to balance the long-term user's diet while matching their immediate preferences. The approach presented in this study relies on the protocol presented in [6] to engage a dialogue between the user and the system. Recall that our previously developed explanation system was "static" with only nutritional factors determining the explanations. Following the feedback we acquired from previous experiments, we improved the explanation generation strategy in a more dynamic manner to enhance the dialogue between the user and agent. The protocol (see Fig. 1) is characterized by the user expressing their preferences and constraints to the NVC, which in turn replies by recommending an appropriate recipe, along with its explanation.

In the context of food recommendation, the user first reveals their constraints (C), which may consist of the ingredients the user may be allergic (e.g., milk, peanuts) to; the (dis) liked ingredients (e.g., specific meat/vegetables); and the desired type of cuisine (e.g., Middle Eastern, Italian, French). After receiving the user's constraints, the agent recommends a recipe (R) along with its explanation (ϵ). The user can *accept* R, *leave* without an agreement, *criticize* R, ϵ, or both. When the user makes a critique, the agent can revise its recommendation/explanation, generating (R'), (ϵ'), or both. This interaction continues in a turn-taking fashion until reaching a termination condition (i.e., Accept or Leave w/o Recommendation) or the time deadline is reached.

In our current implementation, a user can criticize the given recommendation by referring to pre-structured critiques as follows, where Y denotes one of the ingredients chosen by the user. (1) I ate Y recently, (2) I'm allergic to Y, (3) I don't like Y, and (4) I want to give custom feedback. Similarly, the user can criticize the explanations communicated alongside the recommendations with the pre-defined statements such as (1) The explanation is not convincing, (2) The explanation does not fit my case, (3) The explanation is incomplete, (4) The explanation is not clear enough, and (5) I disagree with the explanation.

In the following section, we look into to the ontology database that the recommendation engine takes advantage of while calculating the recommended recipes.

3.1 Ontology structure

The system incorporates an OWL-based Ontology database that includes ontological concepts to represent *users* and *food ingredients*. The *User* concept characterizes the individuals and their eating habits, including any allergy, religious, and lifestyle restrictions. The food concept is characterized by recipes and ingredients that are grouped in classes (e.g., cow-hearts, cherry tomatoes, etc. are grouped under the category of *Tomatoes*). A comprehensive view from *Food* concept in the Protege is shown in Fig. 2.

We establish the object property of *doesNotEat* to identify which food ingredients the user would/should avoid as seen in Fig. 3. The limitations, such as the prohibition of pork for Muslims, are represented by linking object properties (depicted as diamonds) to both the "User" and "Food" concepts. The system verifies whether a particular user class would/could consume a given ingredient class by the *doesNotEat* relation between users and food ingredients. We utilized a compact and localized recipe dataset [2] to build the ontology instances by fitting the ingredients into the respective concept structure manually. We annotated the recipe ingredients by the classes of ingredients within the ontology. A final filter on recipes with incomplete information leaves 1.3K recipes to recommend.

3.2 The baseline recommendation strategy

In this section, we explain the main recommendation strategy of the food recommender system under the following outline. Section 3.2.1 explains the initial filtering and scoring of the food recipes under various modules. Then, Sect. 3.2.2 elaborates the utility function used in determining which recipes to recommend from a healthiness perspective. Finally, 3.2.3 outlines the calculation of the user satisfaction score used in the utility estimation of the recipes.

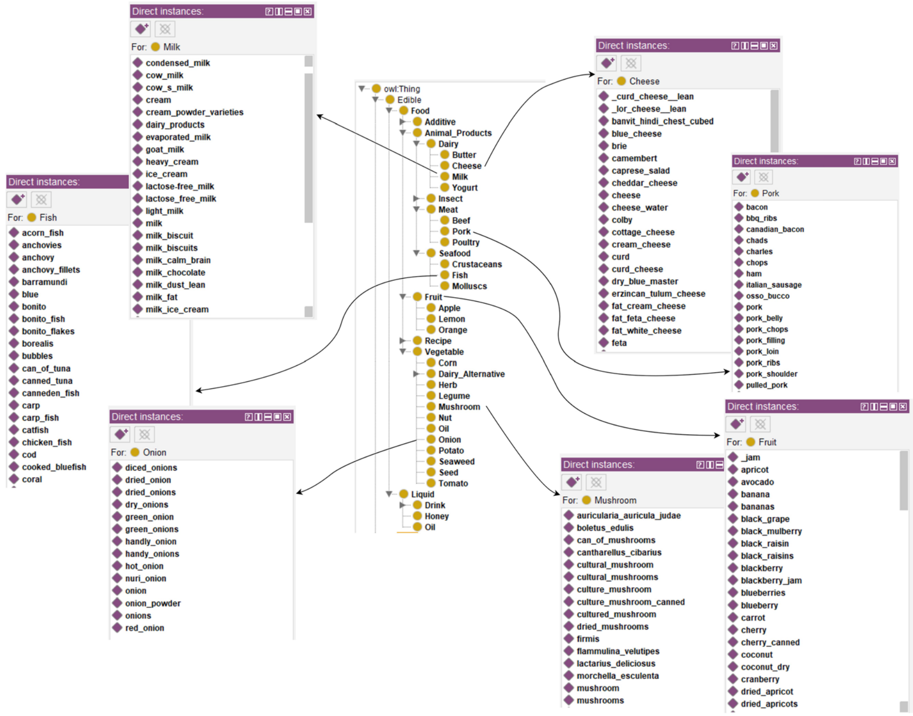


Fig. 2 Protege view of food class

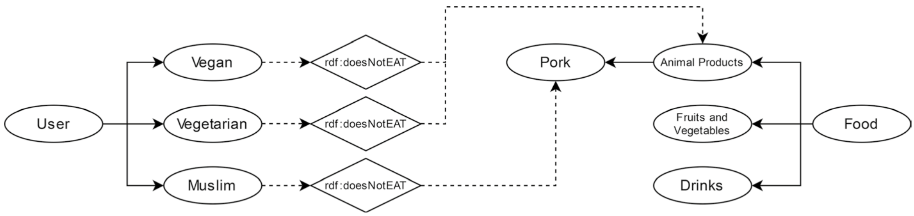


Fig. 3 Broad overview of the ontological structure for food concept

3.2.1 Filtering and scoring recipes

To analyze the applicability of the designed protocol, we have developed a basic recommendation strategy relying on filtering and scoring the recipes concerning the user's constraints and healthiness (see Algorithm 1). First, the NVC agent filters the recipes according to the user's eating habits/constraints via ontology reasoning on what (classes of) ingredients the user would not consume (Lines 1–3). Assuming that the user is vegan, the NVC agent first filters the recipes containing animal-related products. Then, if the same user specifies that they do not like "zucchini", the NVC agent removes the recipes containing zucchini from the remaining candidate list, R_u. In turn, the utilities of the remaining candidate are calculated by considering both healthiness and their alignment with the user preferences. Then, the recipes are sorted according to the calculated utilities (Lines 4–5).[3] The recipe with the highest utility is taken as a candidate recipe, and the system retroactively generates an explanation in line with the recipe's properties (Lines 6–7). This candidate recipe and its corresponding explanation are given to the user.

When the NVC agent receives feedback from the user regarding the recipe, F_r, it filters the candidate recipes according to the updated constraints given by the feedback and selects the highest-ranked recipe similarly (Lines 10–15). When the NVC agent receives feedback from the user regarding the explanation, F_ϵ, it simply generates a new explanation with the underlying recipe (Lines 16–18).

Algorithm 1 AgentDecisionFunction

Require:
R: Recipes;
U: User;
$R_u \subset R$: Recipe dataset tailored for the user;
H_u: Eating habits of the user;
P_u: User Constraints/Preferences;
r_c: Candidate recipe;
ϵ: Explanation for candidate recipe;
F_r: Feedback to the recipes;
F_ϵ: Feedback to the explanation;
Ensure: r_c, ϵ
1: **if** firstRecommendation **then**
2: $R_u \leftarrow$ filterRecipesByCondition(R, H_u)
3: $R_u \leftarrow$ filterRecipesByCondition(R_u, P_u)
4: $U_{R_u} \leftarrow$ calculateUtilities(R_u)
5: $R_u \leftarrow$ rankRecipes(R_u, U_{R_u})
6: $r_c \leftarrow$ getHighestRankRecipe(R_u)
7: $\epsilon \leftarrow$ generateExplanation(r_c)
8: **else**
9: **if** F_r exists **then**
10: $R_u \leftarrow$ filterRecipesByCondition(R_u, F_r)
11: $U_{R_u} \leftarrow$ calculateUtilities(R_u)
12: $R_u \leftarrow$ rankRecipes(R_u, U_{R_u})
13: $R_c \leftarrow$ getHighestRankRecipe(R_u)
14: **end if**
15: **if** F_ϵ exists **then**
16: $\epsilon \leftarrow$ generateExplanation(r_c)
17: **end if**
18: **end if**
19: return (r_c, ϵ)

[3] The details of the utility calculation are explained below.

Table 1 Daily recommended kilocalories (kcal) intake to maintain weight [42]

Activity level	Daily calories
Too little exercise	$calories = BMR * 1.2$
Light exercise	$calories = BMR * 1.375$
Moderate exercise	$calories = BMR * 1.55$
Strong exercise	$calories = BMR * 1.725$
Very strong exercise	$calories = BMR * 1.9$

3.2.2 Utility estimation

To select the suitable recipe, this study relies on multi-criteria decision-making [25]. Multi-criteria decision analysis allows decisions among multiple alternatives evaluated by several conflicting criteria [49]. The adopted multi-criteria decision analysis is done by ranking recipes through a multi-criteria function. The multi-criteria function gives each recipe a score in the dataset. One of the main advantages of using a mathematical function is the transparency of the function and its outcomes. This feature is well suited for our proposed NVC due to the explainability of the generated behavior.

The overall utility of the recipes, based on the multi-criteria, is computed by considering three criteria: Active Metabolic Rate (AMR) score, nutrition value score, and users' Satisfaction score. The final score of the recipes is the weighted sum of the score provided by each module as presented by Eq. 1 where w_a, w_n, w_u denote the weights of each AMR score, nutrition value score, and users' satisfaction score, respectively. Note that each score is normalized to ensure that the overall score is ranged within [0,1].

$$recipeScore = w_n * nutrientsScore + w_a * amrScore + w_u * UsersScore \tag{1}$$

The nutrient-based score is calculated according to the nutritional information of the recipes, such as proteins, lipids, carbohydrates, cholesterol, sodium, and saturated fats. These nutrients have respective recommended amounts for a healthy life [42]. In this work, we take into account the nutrition intake limits specified by the WHO organization.[4] Accordingly, the nutrition-based score is calculated as seen in Eq. 2, where each nutrition score is calculated according to Eq. 3. We assume that consuming less than each nutrient's minimum amount (min_n) is better than its maximum amount (max_n). By following this heuristic, the individual score of each nutrient is calculated.

$$\begin{aligned} nutrientScore(recipe) = score(pro) + score(lip) + score(cb) + \\ score(ch) + score(sod) + score(sat) \end{aligned} \tag{2}$$

$$score(n) = \begin{cases} 5 & \text{if } n \in [min_n, max_n] \\ 3 & \text{if } n < min_n \\ 1 & \text{else} \end{cases} \tag{3}$$

[4] https://www.who.int/news-room/fact-sheets/detail/healthy-diet, http://www.mydailyintake.net/daily-intake-levels/

AMR is the number of calories a person must consume daily depending on height, sex, age, weight, and activity level. Such preliminary information is taken during the registration of the users. The value of AMR is based on the value of Basal Metabolic Rate (BMR), the number of calories required to keep a body functioning at rest, the person's activity level, and the person's desire to maintain or reduce his current weight. Table 1 presents the values to keep the current weight. To compute the AMR score based on the minimum and maximum amount of calories required for a given user available in literature [42], we rely on the same assumption of Eq. 3 that is consuming fewer calories than required ($score = 3$) is better than consuming more calories than required ($score = 1$). In addition, when the amount of calories computed is between the minimum and maximum amount of calories, the score is set to 5.

Conventionally, the most used formula to compute BMR is the Harris equation [22] with Eq. 4 and 5, for men and women, respectively. The authors estimated the constants of Eq. 4 and 5 by several statistical experiments [22].

$$BMR = 10 * weight + 6.25 * height - 5 * age + 5 \tag{4}$$

$$BMR = 10 * weight + 6.25 * height - 5 * age + 161 \tag{5}$$

3.2.3 User satisfaction score

The user satisfaction score is calculated by considering the recipe's popularity among all users and the current user's preferences equally. For the recipe's popularity, we use the ratings the other users gave between [1, 5]. These values are normalized to [0, 1]. Meanwhile, regarding the user's preferences, we check how many ingredient classes are considered to be liked by the user. Here, to determine whether an ingredient is liked or not, we can use explicit feedback from the user as well as rely on user profiling to predict whether the given ingredient is likely to be preferred to be consumed. Here, we use Jaccard Similarity [5] to estimate individual user satisfaction (the rate of the preferred ingredients over the number of all the ingredients of a given recipe).

Let us assume the user-submitted his preference for some ingredients (e.g., ingredients; i_1, i_2, i_3) and we have a recipe such that $R = i_1, i_2, i_5, i_6$ (where i_5 and i_6 are ingredients the user has no preference for). Each ingredient that exists with the liked constraint is considered to be 1 and 0 otherwise. The mean of this operation is 0.5, which is effectively the score of R for this user. For all the recipes, the scores are then max-normalized to place the values between [0, 1], resulting in a relative level of importance for the given recipe. For instance, let us assume that the system knows that the user likes the ingredients i_1, i_2, and i_3 and calculate the score of a recipe consisting of the following ingredients:i_1, i_2, i_5, i_6. The individual user satisfaction would be 2/4, according to Jaccard similarity. If the overall user rating of that recipe is equal to 4 out of 5, then the overall score would be equal to 0.65 ((0.5+0.8)/2).

3.3 Post-hoc explanation generation strategies

This study proposes a Post-Hoc explanation generation technique to improve the transparency and the sociability of the food recommender system to nudge the users to consume healthier food. Section 3.3.1 elaborates on our use of decision trees to explain given food recommendations and Sect. 3.3.2 explains the contrastive food recommendations, where we offer an alternative and explain the differences between. Finally, Sect. 3.3.3 explains how all these approaches are combined.

3.3.1 Item and user based explanations

Decision trees are often used for decision support systems because they are simple and intuitive models that can be easily understood and visualized. They can explain the reasoning behind AI predictions or decisions in a more straightforward form than an otherwise black-box model [4]. In order to discover the important features significantly influencing users' decisions (e.g. carbohydrates, protein, etc.), a decision tree is constructed from a labelled dataset (see Line 1 in Algorithm 2). When we employ the user-based explanation generation method, the decision tree is constructed from historical data in which recipes are labelled with *all users' decisions* (i.e., accept or reject). Conversely, the item-based explanation generation approach utilizes the decision tree constructed from a set of recipes labelled according to *the current user's constraints and feedback*. For that tree, filtered and low-scoring recipes are negatively labelled (-1), recipes that aligned with the user's constraints are positively labelled (+1) and the rest is labeled neutrally (0). After sorting features with respect to their importance (Line 2), we choose three of them to generate an explanation for the given recipe (Lines 3–4 in Algorithm 2).

Algorithm 2 Item-Based/User-Based Explanation Generation

Require:
ϵ_T: Selected features for explanation templates
R_l: Set of labelled items;
m: The amount of explanations to show;
Ensure: ϵ: Explanations of a given item
1: tree $\leftarrow$ DecisionTreeClassifier(R_l);
2: sortedFeatures $\leftarrow$ tree.getExplanationTags()
3: **for** i = 0; i < m^5; i + 1 **do**
4: $\epsilon_T \leftarrow \epsilon_T \cup sortedFeatures.nextImportantFeature()$
5: **end for**
6: return ϵ_T

Figure 4 illustrates a sample item-based tree from one of the live experiment participants' data. For this participant, one could observe that the protein is the most important decision factor for the constructed tree, as it is also visible on Fig. 5 as well.

3.3.2 Contrastive explanations

Additionally, we generate contrastive explanations as outlined in Algorithm 3. First, we select a recipe that is similar to the recommended recipe but it's *recipeScore* is less than the recommended one. To do so, we utilize a pool of filtered (i.e., eliminated from the recommendation pool due to the user constraints/preferences) and/or low-scoring (i.e., not healthy or not tasty for the given user) recipes. We employ the Jaccard Similarity metric [26] to determine the recipe similarity based on their ingredients. From this candidate set of recipes, we choose the one whose similarity with the current recommendation is maximum (Line 1). Then, we compare features of the chosen counter recipe with those of the recommended

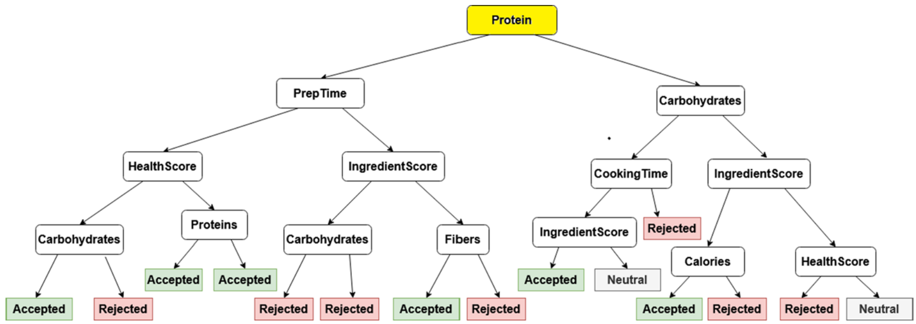


Fig. 4 Sample tree for item-based explanations where "protein" is the most informative feature regarding the information gain

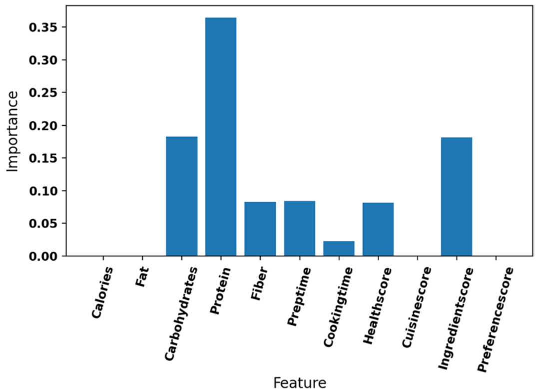


Fig. 5 Corresponding feature importances for the Fig. 4

recipe one by one. If the feature of the chosen recipe has a lower score for healthiness or user satisfaction, we added them into negative feature set, ϵ^{-}, (Lines 2–4); otherwise, inserted into positive feature set, ϵ^{+}, (Lines 5–7). Those features will be used to build a contrastive explanation sentence highlighting the positive side of the recommended recipe while sending away the contrastive recipe by emphasizing its negative sides.

Algorithm 3 Contrastive explanation generation

Require:
R: List of recommended recipes;
R': List of filtered and/or low-recipeScore recipes;
$HScore()$: Scoring function to measure healthiness;
$PScore()$: Scoring function to measure user satisfaction;
r: The recommended recipe;
F: set of recipe features
Ensure:
ϵ^+: Positive Contrastive Explanations of a given recipe
ϵ^-: Negative Contrastive Explanations of a given recipe

```
r' ← argmax_{maxP_u(r)} ||JaccardSimilarity(r, R')|| ;
for each f in F do
    if (HScore(r', f) > HScore(r, f)) || (PScore(r', f) > PScore(r, f)) then
        ε⁻ ← f ;
    else
        ε⁺ ← f ;
    end if
end for
return ε⁺, ε⁻
```

3.3.3 Grammar structure and visual components

From the features acquired by the methods explained in the previous sections, we generate a sentence using the pre-defined grammar-based structure. The structures are composed of two variants: one for the user / item-based explanations is shown in Fig. 6 and the other one for contrastive explanation in Fig. 7. The phrase repository of the

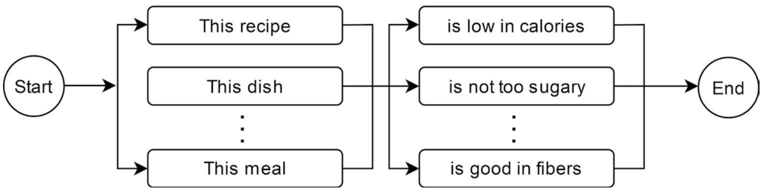


Fig. 6 Grammar structure of the item/user based explanations

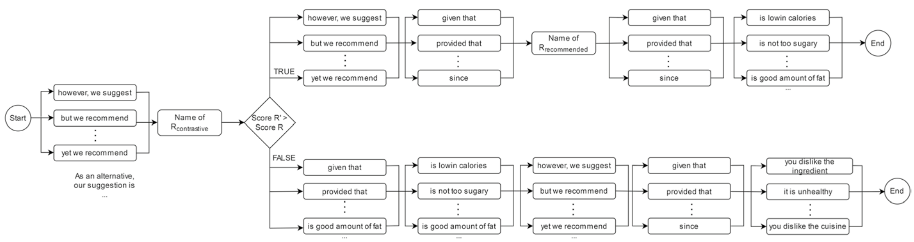


Fig. 7 Grammar structure of the contrastive explanations

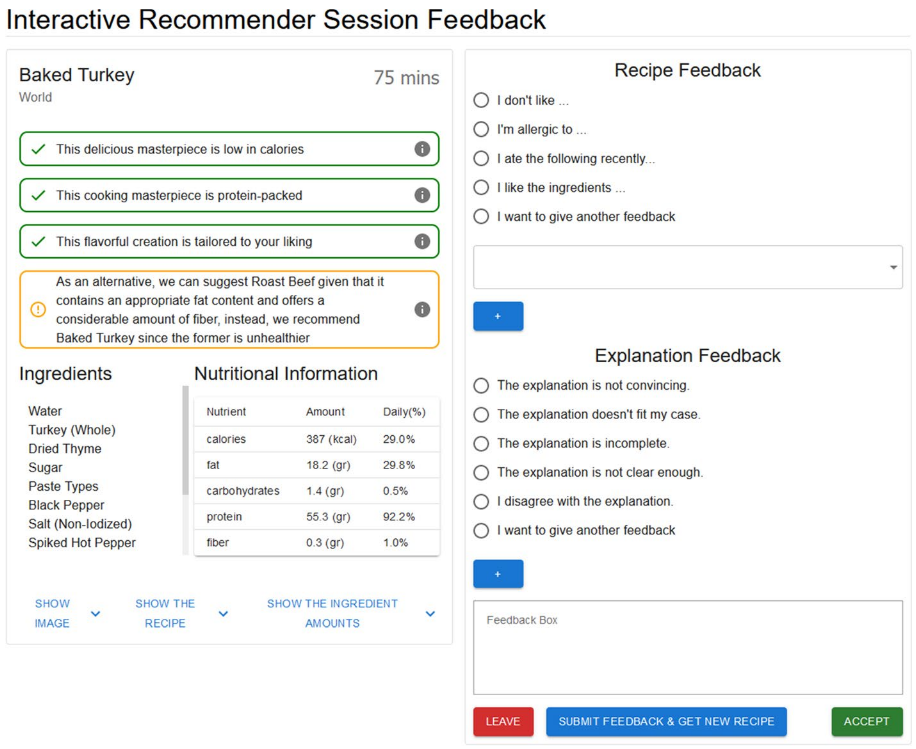


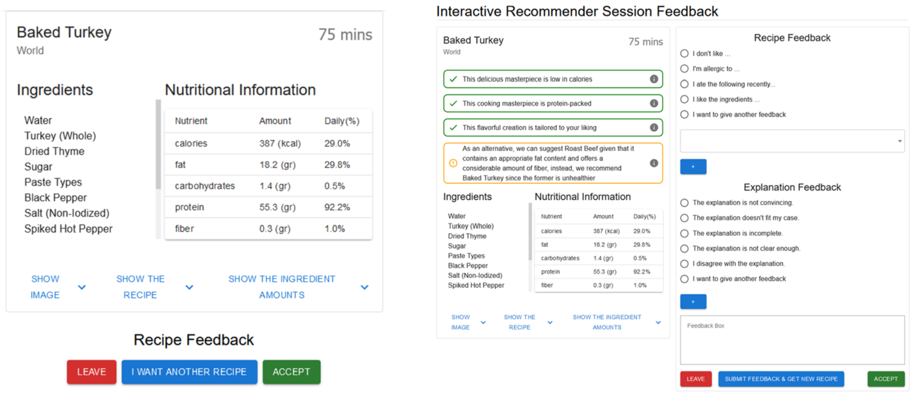
Fig. 8 A sample of the food recommendation interface with explanations

system consists a set of phrases for each decision factor (e.g., for protein: "...provides sufficient protein..."), and other types of phrases such as subject and noun (e.g., "...this recipe..."). The user/item-based explanations are alluring sentences about each positive feature. They are intended to be brief and pithy, whereas contrastive explanations aim to create a comparative explanation with a worse alternative (which can be longer).

Figure 8 shows the novel interface developed to display these explanations. We added visual aspects of explainable recommendations given the success of "graphics" in explaining recommendations [30]. The health-oriented explanations are shown in a green box. Contrastive explanations are outlined in yellow. Additionally, we present nutritional factors related to food to the user.

4 Evaluation

To evaluate the performance of the proposed negotiation framework equipped with enhanced explanations, we conducted tests via a web-based interface for food recommendations. The experimental setup and participants are presented in Sects. 4.1 and 4.2, respectively. Consecutively, Sect. 4.3 reports and discusses the experimental results elaborately.



(a) Traditional Recommender Session (b) Interactive Recommender Session

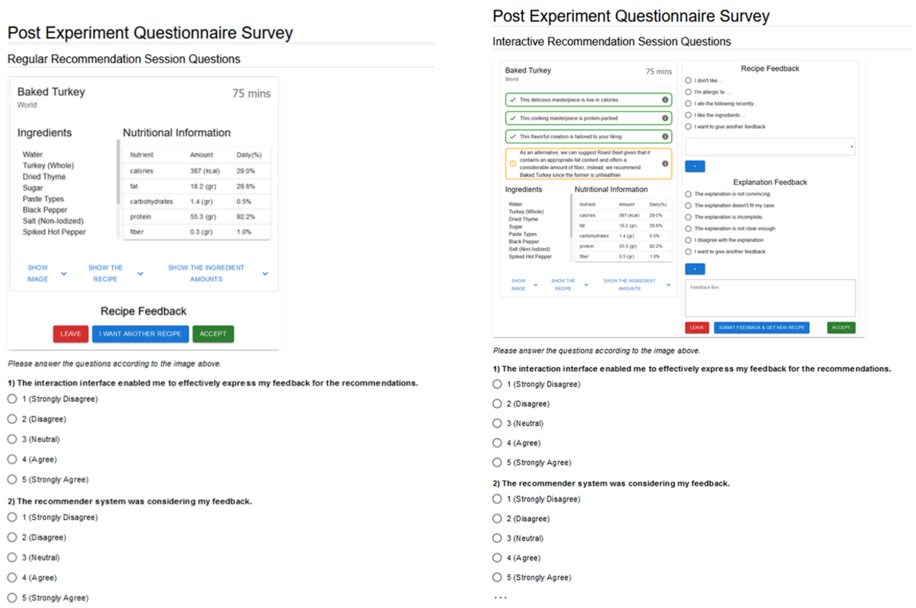
Fig. 9 traditional and interactive recommendation sessions

4.1 Experimental setup

To assess the acceptability and effectiveness of the proposed negotiation-based recommendation framework, we asked participants to experience two variants of food recommender systems: (i) *traditional recommender* where the the system provides solely a recommendation (picture and recipe) without any explanation, leaving the user to accept it or ask for a new recommendation, and (ii) *interactive recommender*, where the original explanation-based negotiation approach is adjusted to an interactive setting, providing explanations for the recommendations and allowing users to give feedback (i.e., approvals and critiques of the recipe and/or explanations). It is worth noticing that we revised the Web participant interfaces in both conditions based on the feedback received in the earlier study presented in [6]. We improved how the food recipes and their supportive explanations are displayed to communicate the explanations more effectively and diminish the effect of factors irrelevant to the quality of explanations, such as pictures. Nutritional information and main ingredients are shown directly alongside several types of explanations. Conversely, as visible in Fig. 9, a picture of the food as well as the details of the recipes are not directly displayed, but available only via an additional click.

We follow the following steps in our experiments[5]. Before conducting the experiments, every participant completed a pre-survey and registration form to provide information about their gender, age, height, weight, level of physical activity, dietary preferences, and any allergies they might have. This information concurs to estimate the healthiness score of recipes recommended to the participant (see Sect. 3.2). To reduce the learning effect among the sessions, the participants were split into two "groups", inverting the starting settings order. A three-minute break was given between the two sessions. Initially, we scheduled a longer break. However, in our pilot experiment, we received negative feedback about the too-long waiting interval.

[5] We selected three explanations for this study. Since it's commonly considered the maximum number of items to show to a user without overwhelming them with too much information [37]



(a) Traditional Recommender Session (b) Interactive Recommender Session

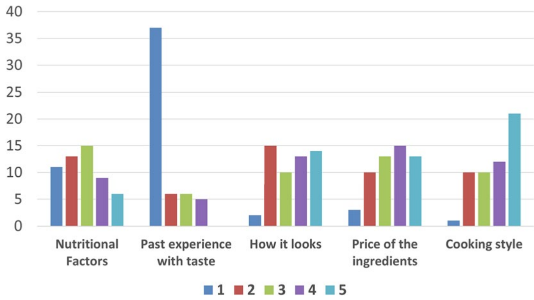
Fig. 10 Traditional and interactive recommendation sessions questions

Following the completion of the experiment, the participants are asked to fill in a questionnaire that primarily comprises 5-point Likert scale questions to assess their experiences in both sessions (one questionnaire per session). The questionnaire follows a within-subject design [29] to gather participants' insights regarding the system's success. To facilitate recalling their experiences and differentiate the sessions, a picture (screen capture) of the given system's setting is displayed at the beginning of the questionnaire page (see Fig. 10). Finally, additional 5-point Likert scale questions were asked to the participants about their perceptions of the received explanations during the Interactive system.

4.2 Participants

In total, there were 54 participants (19 female, 35 male) with diverse backgrounds and age groups took part in the test. The mean age of the attendees is 26.31 years old (with a minimum of 19 and a max of 58 years old). The participants were 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". Figure 11 shows the histogram analysis of the questionnaire. The participants ranked these factors on a scale of 1 to 5, with 1 being the most important factor. One could observe that the majority of the participants (i.e., 69 % of the participants) ranked past experience with the taste of such food to be the most crucial factor in deciding their food recipes to cook, whereas 21% of the participants marked nutritional factors to be the most important.

Fig. 11 Histogram analysis of the pre-survey questionnaire



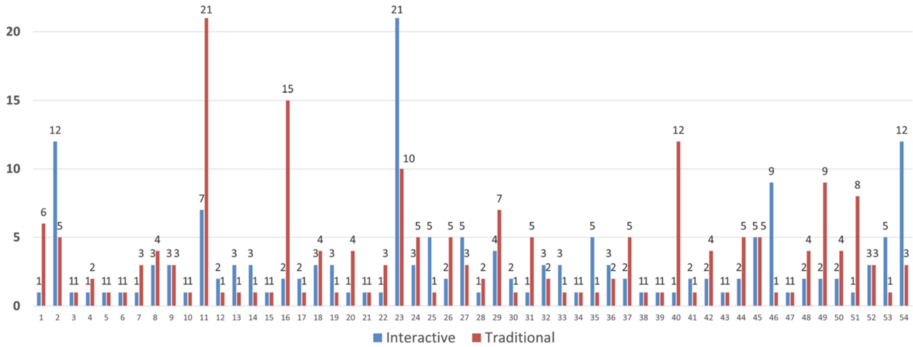


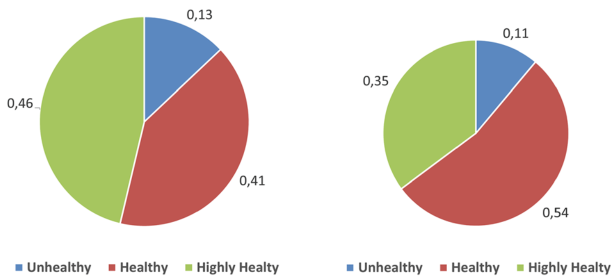
Fig. 12 Total number of rounds per participant, for each interaction type

Conversely, 39% of the participants marked cooking style as the least important factor, whereas the food's appearance was rated the least important by 26% of the participants.

4.3 Experimental results

The success of the self-explanatory systems is usually measured under two categories of metrics; subjective and objective metrics [23, 24, 48]. Objective metrics are metrics derived from the participant actions within the experimental setup, such as success rate (i.e., percentage of sessions ending with an agreement), number of rounds per session, healthiness level of the accepted food recipe, and annotator analysis of possible misunderstandings and feedback given during the Interactive session. Subjective metrics denote the participant scores for the post-experiment questionnaire (see Fig. 15 below). The subjective evaluation questions are about perceived effectiveness, level of detail, user satisfaction, understandability, informativeness, and convenience, meaning that the explanations are appropriate relative to the stated user preferences and constraints. In addition, we asked about the general idea of receiving explanations in addition to recommendations.

We first analyzed the number of sessions that ended successfully. Out of 54 sessions, only two traditional and one Interactive session ended without any agreements. It is worth noting that the participant who failed to find agreement with the Interactive system also couldn't find one with the traditional system.



(a) Traditional Session (b) Interactive

Fig. 13 Percentage of healthiness level of the agreement

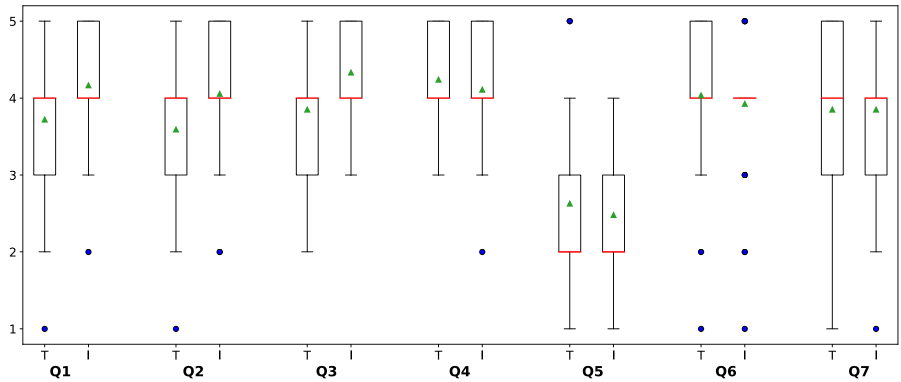
Moreover, participants reached an agreement in the third round on average when they engaged in the Interactive session (i.e., average=3.1 standard deviation= 3.5). In contrast, they accepted the given offer in the forth round on average for the traditional session (i.e., average=3.6 standard deviation= 3.9). Total number of rounds per each participant in each session can be seen in Fig. 12 where the red and blue bars denote the total number of rounds for the traditional and Interactive sessions, respectively. Compared to traditional recommender session, 18 participants had more interaction in the Interactive session, whereas 22 participants required more rounds to find an agreement in the traditional sessions. Another 14 participants finished the interaction in the same number of rounds. The results are not normally distributed according to Kolmogorov–Smirnov test of Normality ($p =< 0.001$), therefore we applied the corresponding non-parametric Wilcoxon Signed-Rank Test ($p = 0.347$). Ultimately, we can infer that the interactive recommender systems do not necessarily take more rounds to reach an agreement, as might be expected.

For the Interactive session, 19 participants accepted recipes of—what we classified according to Eq. 3, the recipes that are labeled “5” as—highly healthy foods; 29 participants preferred healthy foods, and six accepted unhealthy food recipes. For the traditional session, on the other hand, the participants accepted 25 highly healthy options and 22 healthy options; in contrast, seven participants went for unhealthy options. These results are illustrated in Fig. 13. That shows that the Interactive and traditional sessions are similarly effective in meeting the objective of recommending healthy foods. When the Chi-square statistical test was applied, we observed that there was no statistically significant difference between the distributions ($p = 0.40$). Recall that the recommendation strategy itself is the same in both sessions.

The aforementioned results concerning the total number of rounds per session indicate that 18 participants ended the session in the traditional session earlier than the Interactive one. It is possible that they enjoyed exploring the system more in the Interactive system. Since the recommendation strategy employs a time-based concession strategy, the longer it endures, it may offer less healthy food relative to its previous offers. As a result, traditional sessions may end up with healthier food recipes compared to the Interactive system in some cases. On the other hand, there are less unhealthy food recipes agreed by the participants in the Interactive session.

Out of 54 participants, the system received the following evaluative feedback for Interactive sessions:

- “The explanation doesn’t fit my case”, from 4 participants,
- “The explanation is not convincing”, from 4 participants,



Id	Questions label	Metric	W	p
Q1	The interaction interface enabled me to effectively express my feedback for the recommendations.	Sociability	102.0	**.002**
Q2	The recommender system was receptive to my feedback.	Sociability	33.0	**.001**
Q3	The recommender system gave me enough information to decide.	Completeness	58.5	**.0009**
Q4	The recommender system was easy to use.	Usability	121.5	.874
Q5	It was time consuming to find the best recipe in the system.	Usability	155.5	.838
Q6	The interaction with the system was smooth and effortless.	Usability	102.0	.910
Q7	I would like to use such a system in my daily life.	Acceptability	33.0	.5

Fig. 14 Box plot and p-values of comparative analysis of subjective questions between traditional and interactive sessions. Significant results are shown in bold

- "The explanation is not clear enough", from 2 participants,
- "The explanation is incomplete", from 1 participants,
- "I disagree with the explanation", from 1 participants.

Additionally to our given feedback options, which were all negative, participants utilized the custom feedback option to compliment the explanations: "The explanation is acceptable" or "The explanations are enough for me".

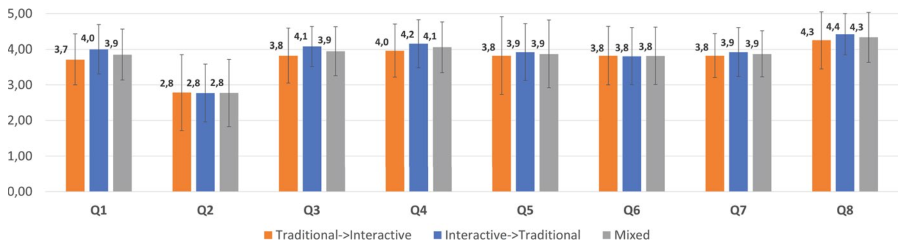
Furthermore, we analyzed the users' responses to the post-test survey to examine how they perceived the traditional and Interactive recommendation system. Since each participant experienced both sessions and the questions are the same for both, we performed a within-analysis statistical comparison test. The data is not normally distributed which is one of the main assumptions made by the pairwise T-test. Thus, we apply the corresponding non-parametric test called the Wilcoxon sign rank test [29]. For all tests, the Confidence Interval (CI) is set to 0.95, $\alpha = 1 - CI = 0.05$.

Figure 14 shows the box plot of the comparative questionnaire between the traditional (R) and the Interactive (I) session, respectively. The orange lines represent the median, the triangles in green the means, and the small blue circles the outliers.

The analysis in the box plot shows that there is a significant improvement for the Interactive sessions, especially for questions $Q1(p = 0.002)$ and $Q2(p = 0.001)$. These two questions measure the system's sociability where the feedback corresponds to the binary of choice of accept and reject for the Traditional system, and the additional live-feedback options for the Interactive system. That is, the results show that the Interactive session is statistically significantly better than the traditional session in terms of sociability. This improvement is reasonable given the additional dialogue options, such as feedback mechanisms, in the Interactive session. Q3 measures the amount of information the participants perceived to be fruitful. The added explanations were recognized by the participants to be effective, hence, here too a significant improvement has been reported ($p = 0.0009$). In other words, the participants perceived that the Interactive session provided better information than the traditional session to make an informed decision.

Moreover, questions Q4 ($p = 0.874$), Q5 ($p = 0.838$), and Q6 ($p = 0.910$) qualify the usability of the system. These values show that there is no significant difference. That means that adding an interactive dimension to the system, can still be effective and efficient. This is in line with what we found earlier about the similar number of turns. Lastly, we measured the acceptability of the two versions of the system. According to the statistical test, there is no significant difference between traditional and Interactive systems for Q7 ($p = 0.5$). The average acceptability score for the Interactive session is approximately 3.85, where 3 is neutral and 4 denotes "agree". Furthermore, we asked all participants which systems they prefer. Only a minor part of the participants (3 out of 54 participants or 6% of them) prefer the traditional one over the Interactive system. In other words, the majority favors the Interactive system (45 participants). The rest is indifferent.

Apart from the comparative analysis, we also ask questions to assess the perceived quality of the explanations in our system. Hoffman et al. provide a list of so called goodness criteria for explanations [23]. Inspired by those statements, we created corresponding statements for the food recommendation system and asked each participant to what extent they agreed. Figure 15 shows the questions and the respective average scores. To examine whether a learning effect may have influenced the results, we report the average scores with respect to (1) participants who started with the traditional sessions (i.e., traditional → Interactive), (2) participants who started with the Interactive session (i.e., Interactive → traditional), and (3) all participants irrespective of the order of sessions (i.e., Mixed). It is



Id	Questions label
Q1	The explanations for recommendations has helped me choose the most convenient recipe.
Q2	The explanations for recommendations were too detailed.
Q3	The explanations displayed during the interaction were satisfactory.
Q4	The explanations for recommendations were clear and easy to understand.
Q5	The explanations were sufficient to make an informed decision for healthiness.
Q6	The explanations were realistic in terms of healthiness of given recipes.
Q7	The explanation let me know how convenient the recipe is.
Q8	Rate your appreciation of the idea of receiving explanation in addition to recommendations.

Fig. 15 Evaluation questionnaire results, shown per order of the sessions

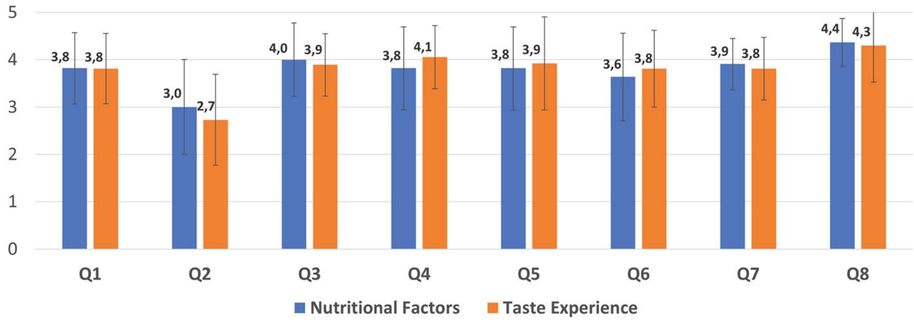


Fig. 16 Questionnaire results per pre-survey answers

clearly seen that the counter-balancing technique works. The results for both orderings are similar. In general, participants are satisfied with the given explanations and appreciated the idea of receiving explanations in addition to the given recommendations. They do not agree that the explanations were too detailed. In addition, they found the explanations help them choose the most convenient recipe.

Lastly, we categorized participants based on their responses to the pre-survey question—the importance of the factors on their decision making (See Fig. 11. Since there are a few participants who found the most important factor as how the food looks, price of the ingredients, and cooking style, we only categorized the participants who voted the most important factor in choosing a recipe to be the past experience with taste and the nutritional factors of a given food. This categorization is also in line with our objectives. Figure 16 shows the score of the aforementioned explanation related questions and responses of the participants in each category. Note that since order of session does not influence the results, we only show the average scores for all participants who fit in the given category. We could not find any significant differences in their responses.

5 Conclusions

The recent widespread use of opaque AI-based systems is raising questions about trustworthiness and transparency. Skepticism skyrockets when the decisions to be taken are safety-critical (i.e., AI outcomes can significantly influence people's life and health—like nutrition). This study presents an interactive explainable recommendation framework where the system seamlessly negotiates with its users by making offers and explaining why this offer is good for them. The user can criticize the given recommendation and/or associated explanation. The proposed framework aims to improve the system's transparency via interactive explanations. User experiments have been conducted to evaluate the proposed interactive recommender system. Participants have been asked to experience the interactive recommender and the regular one (a version of the system without explanation and feedback mechanism), as well as to fill pre- and post-experiment surveys. Although both the recommender might have recommended the same food item (in the same conditions), experimental results showed that the participants were more satisfied (in general) with the idea of explanations and appreciated generated explanations. Moreover, they perceived that the information and process for choosing their food recipe were more informative and complete in the proposed interactive recommender and felt more sociable and reactive to their feedback. Furthermore, interactive sessions performed slightly better in terms of effectiveness regarding the number of agreements and rounds.

We have tried to set-up the user studies in such a way that they give reliable results. However, our results may still suffer from limitations in the research set-up.

First, although the food recipes are derived from a real food recipe repository prepared by some nutritionists, it is worth noticing that participants were involved in a system test rather than receiving accurate food advice. We mainly compare interactive explainable recommenders with regular recommenders by keeping their recommendation strategy the same.

Second, in this research the main difference between a regular recommender and an interactive recommender system is the presence or absence of both explanation and feedback. Therefore, it is not possible to distinguish which effect, added explanation or added feedback, is responsible for the results. This signals a clear limitation in the set-up of the user experiments. In defense, consider the alternative. To separate these effects would require building a recommender system that allows negative feedback, without providing a response to that feedback in the form of a better explanation. Although theoretically interesting, that would not be a practically useful system.

Third, there is a lot of room to improve the recommendation algorithm itself. For example, we envision learning user preferences over time, and adapting the system behavior accordingly. Yet, our results already show that the proposed approach is promising.

In future work, we plan to study the effect of the precise moment in which the explanations are displayed, during the interaction and decision-making process. Recall that the current system generates explanations whenever it provides a recommendation. An interesting alternative would be to investigate so-called on-demand explanations, which are only provided when the need occurs. The need for an explanation may be signalled by a question like 'why' or 'how'?.

Furthermore, we plan to measure the effectiveness of each type of explanation strategy (user-centred, contrastive, counterfactual, etc) individually, rather than as the combined whole, we have now.

The ultimate goal of our research is to refine the current recommender engine, and integrate it into an existing chatbot framework for persuading and helping a user to change eating habits over a longer period of time. The existing chatbot system is called EREBOTS [7]. The combination of long-term persuasion and coaching from EREBOTS and explainable recommendation sessions from this system, will realize a fully agentified NVC system.

Acknowledgements This work has been supported by the CHIST-ERA grant CHIST-ERA-19-XAI-005, and by the Swiss National Science Foundation (G.A. 20CH21_195530), the Italian Ministry for Universities and Research, the Luxembourg National Research Fund (G.A. INTER/CHIST/19/14589586), the Scientific and Research Council of Turkey (TÜBİTAK, G.A. 120N680).

Author contributions All authors have contributed to the proposed approach, BB, MT, and RA additionally realized the system. BB, AN, IT, JH, and RA designed the post-survey questionnaire. BB, RA, DC, designed the experiment process. BB, IT, JH, DC, and RA analysis and discussion of experiment results. BB, IT, and RA prepared the figures. All authors wrote some parts of the sections and made a final review of the manuscripts.

Funding Open access funding provided by University of Applied Sciences and Arts Western Switzerland (HES-SO). This work has been supported by theCHIST-ERAGrant CHIST-ERA-19-XAI-005, and by the Swiss National Science Foundation (G.A. 20CH21_195530), the Italian Ministry for Universities and Research, the Luxembourg National Research Fund (G.A. INTER/CHIST/19/14589586), the Scientific and Research Council of Turkey (TÜBİTAK, G.A. 120N680).

Data availability The data that supports the findings of this study are available upon request. However, due to privacy and confidentiality concerns, certain restrictions may apply to the availability of specific datasets. Requests for access to the data can be directed to the corresponding author. They will be subject to a data-sharing agreement to ensure compliance with relevant regulations and ethical considerations.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval We would like to state that the experiment protocol adopted in this study was approved by the Ethics Committee of Özyeğin University, and informed consent was obtained from all participants.

References

1. (2017). *Sigir '17: Proceedings of the 40th international ACM sigir conference on research and development in information retrieval*. Association for Computing Machinery.
2. (2022). Yemek tarifleri. https://www.diyetkolik.com/yemek-tarifleri/. Accessed 01 January 2022.
3. Ancona, M., Ceolini, E., & Öztireli, A. C. (2017). A unified view of gradient-based attribution methods for deep neural networks. In *CoRR*.
4. Anjomshoae, S., Najjar, A., Calvaresi, D., & Framling, K. (2019). Explainable agents and robots: Results from a systematic literature review. In *AAMAS*, Montreal, Canada, May 13–17 (pp. 1078–1088).
5. Ayub, M., Ghazanfar, M. A., Maqsood, M., & Saleem, A. (2018). A Jaccard base similarity measure to improve performance of CF based recommender systems (pp 1–6).
6. Buzcu, B., Varadhajaran, V., Tchappi, I., et al. (2022). *Explanation-based negotiation protocol for nutrition virtual coaching* (pp. 20–36). Springer.
7. Calvaresi, D., Calbimonte, J. P., Siboni, E., Eggenschwiler, S., Manzo, G., Hilfiker, R., & Schumacher, M. (2021). Erebots: Privacy-compliant agent-based platform for multi-scenario personalized health-assistant chatbots. *Electronics, 10*, 666.
8. Calvaresi, D., Carli, R., Piguet, J. G., Contreras, V. H., Luzzani, G., Najjar, A., & Schumacher, M. (2022) Ethical and legal considerations for nutrition virtual coaches. In *AI and ethics* (pp. 1–28).
9. Cemiloglu, D., Catania, M., & Ali, R. (2021). Explainable persuasion in interactive design. In *2021 IEEE 29th international requirements engineering conference workshops (REW)* (pp. 377–382).
10. Chen, M., Jia, X., Gorbonos, E., et al. (2020). Eating healthier: Exploring nutrition information for healthier recipe recommendation. *Information Processing & Management, 57*(6), 102051.
11. Chi, Y. L., Chen, T. Y., & Tsai, W. T. (2015). A chronic disease dietary consultation system using owl-based ontologies and semantic rules. *Journal of Biomedical Informatics, 53*, 208–219.
12. Corrado, S., Luzzani, G., Trevisan, M., & Lamastra, L. (2019). Contribution of different life cycle stages to the greenhouse gas emissions associated with three balanced dietary patterns. *Science of the Total Environment, 660*, 622–630.
13. El-Dosuky, M., Rashad, M. Z., Hamza, T., & El-Bassiouny, A. H. (2012). Food recommendation using ontology and heuristics. In *Advanced machine learning technologies and applications: First international conference, AMLTA 2012, Cairo, Egypt, December 8–10, 2012. Proceedings 1* (pp. 423–429). Springer.
14. Fanda, L., Cid, Y. D., Matusz, P. J., & Calvaresi, D. (2021). To pay or not to pay attention: Classifying and interpreting visual selective attention frequency features. In *International workshop on explainable, transparent autonomous agents and multi-agent systems* (pp. 3–17). Springer.
15. Forbes, P., & Zhu, M. (2011). Content-boosted matrix factorization for recommender systems: experiments with recipe recommendation. In *Proceedings of the fifth ACM conference on recommender systems* (pp. 261–264).

16. Freyne, J., & Berkovsky, S. (2010). Recommending food: Reasoning on recipes and ingredients (pp. 381–386).
17. Ge, M., Ricci, F., & Massimo, D. (2015). Health-aware food recommender system. In *Proceedings of the 9th ACM conference on recommender systems* (RecSys '15, pp. 333–334). Association for Computing Machinery.
18. Gibney, M. J., Forde, C. G., Mullally, D., & Gibney, E. R. (2017). Ultra-processed foods in human health: A critical appraisal. *The American Journal of Clinical nutrition, 106*(3), 717–724.
19. Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2018). A survey of methods for explaining black box models. *ACM Computing Surveys (CSUR), 51*(5), 1–42.
20. Gunning, D., & Aha, D. (2019). Darpa's explainable artificial intelligence (xai) program. *AI Magazine, 40*(2), 44–58.
21. Hammond, K. J. (1986). Chef: A model of case-based planning. In *AAAI*.
22. Harris, J. A., & Benedict, F. G. (1918). A biometric study of human basal metabolism. *Proceedings of the National Academy of Sciences, 4*(12), 370–373.
23. Hoffman, R. R., Mueller, S. T., Klein, G., & Litman, J. (2018). Metrics for explainable AI: Challenges and prospects. arXiv:1812.04608 [csAI]
24. Hulstijn, J., Tchappi, I., Najjar, A., & Aydogan, R. (2023). (2023) Metrics for evaluating explainable recommender systems. In *AAMAS, EXTRAAMAS 2023*. Springer.
25. Ishizaka, A., & Siraj, S. (2018). Are multi-criteria decision-making tools useful? an experimental comparative study of three methods. *European Journal of Operational Research, 264*(2), 462–471.
26. Jaccard, P. (1912). The distribution of the flora in the alpine zone 1. *New Phytologist, 11*(2), 37–50.
27. Jannach, D., Pu, P., Ricci, F., & Zanker, M. (2021). Recommender systems: Past, present, future. *AI Magazine, 42*, 3–6.
28. Lawo, D., Neifer, T., Esau, M., & Stevens, G. (2021) Buying the 'right' thing: Designing food recommender systems with critical consumers. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (CHI '21). Association for Computing Machinery.
29. Lazar, J., Feng, J., & Hochheiser, H. (2017). Research methods in human–computer interaction
30. Millecamp, M., Htun, N. N., Conati, C., & Verbert, K. (2019). 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* (pp. 397–407).
31. Miller, T. (2019). Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence, 267*, 1–38.
32. Mokdara, T., Pusawiro, P., Harnsomburana, J. (2018), Personalized food recommendation using deep neural network. In *2018 Seventh ICT international student project conference (ICT-ISPC)* (pp. 1–4). IEEE.
33. Mualla, Y., Tchappi, I., Kampik, T., Najjar, A., Calvaresi, D., Abbas-Turki, A., & Nicolle, C. (2022). The quest of parsimonious Xai: A human-agent architecture for explanation formulation. *Artificial Intelligence, 302*, 103573.
34. Padhiar, I., Seneviratne, O., Chari, S., Gruen, D., & McGuinness, D. L. (2021). Semantic modeling for food recommendation explanations (pp. 13–19).
35. Saarela, M., & Jauhiainen, S. (2021). Comparison of feature importance measures as explanations for classification models. *SN Applied Sciences, 3*, 1–12.
36. Samih, A., Adadi, A., & Berrada, M. (2019). Towards a knowledge based explainable recommender systems. In *Proceedings of the 4th international conference on big data and internet of things* (BDIoT'19). Association for Computing Machinery.
37. Shimazu, H. (2001). Expertclerk: Navigating shoppers' buying process with the combination of asking and proposing. In *Proceedings of the 17th international joint conference on artificial intelligence* (IJCAI'01, Vol. 2, pp. 1443–1448). Morgan Kaufmann Publishers Inc.
38. Sovrano, F., & Vitali, F. (2022). Generating user-centred explanations via illocutionary question answering: From philosophy to interfaces. *ACM Transactions on Interactive Intelligent Systems, 12*(4), 1–32.
39. Teng, C. Y., Lin, Y. R., & Adamic, L. A. (2012a). Recipe recommendation using ingredient networks. In *Proceedings of the 4th annual ACM web science conference* (WebSci '12, pp. 298–307)
40. Teng, C. Y., Lin, Y. R., & Adamic, L. A. (2012b). Recipe recommendation using ingredient networks. In *Proceedings of the 4th annual ACM web science conference* (pp. 298–307).
41. Tintarev, N., & Masthoff, J. (2015). Explaining recommendations: Design and evaluation. Recommender systems handbook (pp. 353–382).
42. Toledo, R. Y., Alzahrani, A. A., & Martinez, L. (2019). A food recommender system considering nutritional information and user preferences. *IEEE Access, 7*, 96695–96711.
43. Trang Tran, T. N., Atas, M., Felfernig, A., & Stettinger, M. (2018). An overview of recommender systems in the healthy food domain. *Journal of Intelligent Information Systems, 50*, 501–526.

44. Tran, T. N. T., Felfernig, A., Trattner, C., et al. (2021). Recommender systems in the healthcare domain: State-of-the-art and research issues. *Journal of Intelligent Information Systems, 57*(1), 171–201.
45. Ueda, M., Takahata, M., & Nakajima, S. (2011). User's food preference extraction for personalized cooking recipe recommendation. In *Workshop of ISWC, Citeseer* (pp. 98–105).
46. Ueda, M., Asanuma, S., Miyawaki, Y., & Nakajima, S. (2014). Recipe recommendation method by considering the users preference and ingredient quantity of target recipe. In *Proceedings of the international multiconference of engineers and computer scientists* (pp. 12–14).
47. Vultureanu-Albişi, A., & Bădică, C. (2021). Recommender systems: An explainable AI perspective. In *2021 International conference on innovations in intelligent systems and applications (INISTA)* (pp. 1–6).
48. van der Waa, J., Nieuwburg, E., Cremers, A., et al. (2021). Evaluating Xai: A comparison of rule-based and example-based explanations. *Artificial Intelligence, 291*, 103404.
49. Wang, L., Rodriguez-Dominguez, R. M., & Wang, Y. M. (2018). A dynamic multi-attribute group emergency decision making method considering experts' hesitation. *International Journal of Computational Intelligence Systems, 11*, 6.
50. Xu, Y., Collenette, J., Dennis, L., & Dixon, C. (2022). Dialogue-based explanations of reasoning in rule-based systems. In *3rd Workshop on explainable logic-based knowledge representation*.
51. Zhu, X., Wang, D., Pedrycz, W., et al. (2023). Fuzzy rule-based local surrogate models for black-box model explanation. *IEEE Transactions on Fuzzy Systems, 31*(6), 2056–2064.

Authors and Affiliations

Berk Buzcu[1,5] · Melissa Tessa[2] · Igor Tchappi[4] · Amro Najjar[3,4] · Joris Hulstijn[4] · Davide Calvaresi[5] · Reyhan Aydoğan[1,6,7]

✉ Berk Buzcu
berk.buzcu@hevs.ch

Melissa Tessa
im_tessa@esi.dz

Igor Tchappi
tchappi.haman@uni.lu

Amro Najjar
amro.najjar@list.lu

Joris Hulstijn
joris.hulstijn@uni.lu

Davide Calvaresi
davide.calvaresi@hevs.ch

Reyhan Aydoğan
reyhan.aydogan@ozyegin.edu.tr

1 Computer Science, Özyeğin University, Istanbul, Turkey

2 Computer Science, High National School of Computer Science ESI ex-INI, Algiers, Algeria

3 Luxembourg Institute of Science and Technology, Esch-sur-Alzette, Luxembourg

4 University of Luxembourg, Esch-sur-Alzette, Luxembourg

5 University of Applied Sciences and Arts Western Switzerland (HES-SO Valais-Wallis), Sierre, Switzerland

6 Interactive Intelligence, Delft University of Technology, Delft, The Netherlands

7 University of Alcala, Alcala de Henares, Spain

Applied Intelligence (2023) 53:29741–29757
https://doi.org/10.1007/s10489-023-05001-9

Conflict-based negotiation strategy for human-agent negotiation

Mehmet Onur Keskin[1] · Berk Buzcu[1] · Reyhan Aydoğan[1,2]

Accepted: 6 September 2023 / Published online: 3 November 2023

Abstract

Day by day, human-agent negotiation becomes more and more vital to reach a socially beneficial agreement when stakeholders need to make a joint decision together. Developing agents who understand not only human preferences but also attitudes is a significant prerequisite for this kind of interaction. Studies on opponent modeling are predominantly based on automated negotiation and may yield good predictions after exchanging hundreds of offers. However, this is not the case in human-agent negotiation in which the total number of rounds does not usually exceed tens. For this reason, an opponent model technique is needed to extract the maximum information gained with limited interaction. This study presents a conflict-based opponent modeling technique and compares its prediction performance with the well-known approaches in human-agent and automated negotiation experimental settings. According to the results of human-agent studies, the proposed model outpr erforms them despite the diversity of participants' negotiation behaviors. Besides, the conflict-based opponent model estimates the entire bid space much more successfully than its competitors in automated negotiation sessions when a small portion of the outcome space was explored. This study may contribute to developing agents that can perceive their human counterparts' preferences and behaviors more accurately, acting cooperatively and reaching an admissible settlement for joint interests.

Keywords Opponent modelling · Preference modelling · Human-agent negotiation · Automated negotiation

1 Introduction

Negotiation is an interaction among self-interested parties that have a conflict of interests and aim to achieve a joint agreement. It can occur daily basis when parties need to make decisions collectively on any matters such as personal activities (e.g., arranging holiday plans), professional procedures (e.g., job interviews, task or resource allocations), or societal matters(e.g., effective energy distribution). Depending on the complexity of the decisions, this process can be time-consuming and cumbersome for human stakeholders. Therefore, researchers in the field of Artificial Intelligence have put their effort into automating this process over the last decades [1, 8, 14]. Recently, there has been a high interest in human-agent negotiations in which intelligent agents negotiate with their human counterparts [4, 28]. Creating large-scale social impact by such intelligent systems requires understanding how human decisions are made and their preferences and interests [26]. That is, agents should be capable of understanding why their opponent made such offers and what is acceptable to their opponent so that it can adapt its bidding strategy accordingly to increase the chance of reaching mutually beneficial agreements. That shows the importance of the opponent modeling during the negotiation.

There are a variety of opponent modeling techniques proposed in automated negotiation literature [6]. As far as the existing opponent models to predict the opponent's preferences are concerned, it is observed that they attempt to learn a model from bid exchanges and mostly have some particular assumptions about both opponent's bidding behavior and preference model (e.g., having an additive utility function and employing time-based concession strategy). Even simple heuristic models such as the frequentist approach [19, 31] perform well in negotiation. Although there are relatively much fewer offer exchanges in human-agent negotiation in contrast to automated one (i.e., the number of

✉ Reyhan Aydoğan
R.Aydogan@tudelft.nl

Mehmet Onur Keskin
onur.keskin@ozu.edu.tr

Berk Buzcu
berk.buzcu@ozu.edu.tr

1 Özyeğin University, Istanbul, Turkey

2 Computer Science, Interactive Intelligence, TU Delft, Delft, Netherlands

offers typically does not exceed 20-30 offers in human-agent negotiation [22]), some studies adopt their variants in human-agent negotiation [25].

This study pursues an alternative way of modeling human opponents' preferences by searching for cause-effect relationships in human negotiators' bidding patterns. Here, the main challenge is to learn meaningful preference relations that enable our agents to generate offers that are more likely to be acceptable by their opponents despite the small number of offer exchanges. Accordingly, this study proposes a novel conflict search-based opponent modeling strategy mainly designed to learn human opponents' preferences in multi-issue negotiations to generate well-targeted offers leading to mutually beneficial agreements (i.e., high social welfare). The proposed opponent modeling approach has been evaluated experimentally concerning different performance metrics, such as the model's accuracy and the model's effect on the negotiation and negotiation outcome. To show the performance of the proposed approach, we conducted two human-agent negotiation experiments involving 70 participants in total and compared our agent performance with those of the aforementioned well-known frequentist approaches [19, 31]. Our results showed that the proposed conflict-based opponent model outperformed them dramatically in terms of their prediction accuracy. Furthermore, we studied the effect of the model on the negotiation outcome in automated negotiation by involving 15 state-of-the-art negotiation agents from the International Automated Negotiating Agents Competition (ANAC) [15] on six different negotiation scenarios. Our results showed that our agent gained the highest average individual utility and social welfare (i.e., both product of utilities and the sum of utilities) on average.

The rest of the paper is organized as follows. Section 2 reviews the related work on opponent modeling. The proposed opponent model is explained in Section 3, and the negotiation strategy of the agent utilizing the model is defined in Section 4. Section 5 presents our experimental setup and analysis of the results. Finally, we conclude our work with a discussion involving future work directions in Section 6.

2 Related work

Automated negotiation has been widely studied for several decades, and a variety of negotiation frameworks have been proposed so far [2, 8]. By their nature, automated agents try to find the most beneficial agreement for both parties by making many consecutive offers up to a particular deadline (time or round). As Hindriks, Jonker, and Tykhonov point out that agents can benefit from learning about their opponent during negotiation [12], a variety of opponent modeling approaches have been proposed in the negotiation community, such as opponent's preferences (e.g., [12, 19, 21, 25, 31, 32]), the acceptability of an offer (e.g., [20, 26, 29]) and negotiation strategy/attitude (e.g., [16, 23, 27]). The main opponent strategy is identifying the opponent's preferences by analyzing offer exchanges between parties. Afterward, the agent examines the opponent's negotiation offers with estimated opponent preferences to get an idea about its strategy/attitude. Various modeling techniques have been used in these strategies, such as kernel density, Bayesian learning, and frequentist models. While building up their model, those opponent modeling approaches rely on some assumptions such as having a predetermined deadline, capturing their preferences in the form of an additive utility function, and following a turn-taking negotiation protocols such as (Stacked) Alternating Offers Protocol [2] and conceding over time (e.g., time-based concession strategies). In the following part, we mention the most relevant works. A more detailed explanation about opponent modeling can be found in the survey [6].

Another common preference model technique in the literature is based on Bayesian learning [12, 32]. Hindriks et al. use Bayesian learning to predict the shape of the opponent's utility function, the corresponding rank of issue values, and issue weights [12]. As an extension of Hindrik's work, Yu et al. incorporate regression analysis into Bayesian learning by comparing the predicted future bids and actual incoming bids. Accordingly, they update the Bayesian belief model by considering both current and expected coming bids.

Recent studies' most common preference modeling strategies are variations of the frequentist models. The winner of the Second Automated Negotiating Agents Competition [15] called Hardheaded agent [19] uses a simple counting mechanism for each issue value and analyzes the contents of the opponent's consequent offers. The main heuristic is that the opponent would concede less on the essential issues while using the preferred values in its offers. Therefore, while analyzing the opponent's current and previous offer, if the value of an issue is changed, that issue's weight is decreased by a certain amount. While such a simple approach is initially intuitive, information loss seems inevitable. Due to the nature of the negotiation, the opponent may need to concede even on important issues. Those moves may mislead the model. In addition, the concession amount may vary during the negotiation, which the frequentist approach needs to capture.

Moreover, suppose the opponent repeats the same bid multiple times. In that case, the model may overvalue those repeated issue values while underestimating the unobserved values (e.g., converging a zero utility since it is not seen). Tunalı, Aydoğan, and Sanchez [31] aims to resolve those problems by comparing the windows of offers instead of consecutive pairs of offers and offering a more robust estimation of the opponent's behavior. It adopts a decayed weight update to avoid incorrect updates when opponents concede on the most critical issues. Furthermore, it smoothly increases the importance of issue values to avoid unbalanced issue value distributions when the opponent offers the same offer repeatedly. Although their approach outperforms the classical frequency approach, it still suffers from only counting the issue value appearance because it ignores the varying utility patterns of the opponent's offers.

Apart from the opponent modeling in automated negotiation, we review the opponent modeling approaches particularly designed for human-agent negotiations. Lin et al. introduce the QOAgent using kernel density estimation (KDE) for modeling opponent's preferences [21]. According to their results, the QOAgent can reach more agreements. In most cases, it achieves better agreements than the human counterpart playing the same role in individual utility. As an extension of QOAgent, Oshrat et al. present an agent unlike most other negotiating agents in the automated negotiation, the KBAgent attempts to utilize previous negotiations with other human opponents having the same preferences to learn the current human opponent [26]. This approach requires an essential assumption that human participants will behave similarly to each other. Thus, KBAgent builds a broad knowledge base from its previous opponents and accordingly offers based on a probabilistic model constructed from the knowledge base utilizing kernel density estimation. In their experimental comparison, the KBAgent outperformed the QOAgent. In these studies, the authors focus on the overall negotiation performance rather than the performance of the proposed opponent modeling approach. However, we examine the accuracy of the opponent modeling and the performance of the whole negotiation strategy.

Furthermore, Nazari, Lucas, and Gratch follow a similar intuition with frequentist models for human-agent negotiation [25]. However, they take into account only the preference ranking of the negotiation issues instead of estimating the overall utility of each outcome. For issue values, they consider a predefined ordering. However, those assumptions may not hold in negotiations where a human participant may have a different evaluation of issue values. In their negotiation, their agent considers the importance of the issues and the expected ordering of the issue values while generating their offers. A similar heuristic with the frequentist approach holds here. That is, an issue is more important if the opponent consistently asks for more on that issue. It leads to the same intrinsic problem of the frequentist approach.

Instead of learning an explicit preference model, some studies focus on understanding what offers would be acceptable for their opponent. Sanchez et al. use Bayesian classifiers to learn the acceptability of partial offers for each team member in a negotiation team [29]. They present a model for negotiation teams that guarantees unanimous decisions consisting of predictable, compatible, and unforeseen issues. The model maximizes the probability of being accepted by both sides. While their model relies on predictable issues such as price, our model is designed to handle unpredictable discrete issues. Lastly, it is good to mention the reinforcement learning approach proposed for human-agent negotiation [20]. Lewis et al. collect a large dataset consisting of offers represented in natural language from 5808 sessions on Amazon's Mechanical Turk. They present a reinforcement learning model to maximize the agent's reward against human opponents. Accordingly, they aim to estimate the negotiation states acceptable for their human counterparts.

More recently, researchers have been trying to incorporate deep learning models in opponent modeling. For instance, Sengupta et al. has implemented a reinforcement learning-based agent that can adapt to unknown agents per experiences with other agents. In order to model the opponent, they have applied the Recurrent Neural Networks model, specifically LSTM, since they use time series data from the negotiation steps. However, they switched their implementation to a 1D-CNN classifier instead due to data limitations. They observe an opponent agent's bidding strategy according to the agent's self-utility and try to cast it into a class of known behaviors. According to this classification, the agent swaps negotiation strategy within the runtime [30].

Meanwhile, Hosokowa and Fujita expand upon the classical frequentist approach through the addition of the ratio of offers within specified slices of the negotiation timeline, and they implement a weighting function to stabilize the ratios as time passes to capture the change of an opponent's concession toward the end of negotiation [13].

3 Proposed conflict-based opponent modelling (CBOM)

Our opponent modeling called *Conflict-Based Opponent Modeling* (CBOM) aims to estimate the opponent's preferences represented utilizing an additive utility function shown in (1) where w_i represents the importance of the negotiation issue I_i (i.e., issue weight), o_i represents the value for issue i in offer o, and V_i is the valuation function for issue i, which returns the desirability of the issue value. Without losing generality, it is assumed that $\sum_{i \in n} w_i = 1$ and the domain of V_i is (0,1) for any i. The higher the V_i is, the more preferred an issue value is.

$$\mathcal{U}(o) = \sum_{i=1}^{n} w_i \times V_i(o_i) \tag{1}$$

Regarding the issue valuation/weight functions (i.e., preferences on issue values), targeting to learn these functions directly from the opponent's offer history may not be a reliable approach since the opponent's negotiation strategy may mislead us. Although the contents of the opponent's offers give insight into which values are more preferred over others, depending on the employed strategy, we may end up with a different model estimation. For instance, the frequency of the issue value appearance might be a good indicator for understanding the ranking of issue values. However, it is not sufficient to deduce to what extent each issue value is preferred. Fluctuations in the opponent's offers or repeating the exact offers often mislead the agent into accurately estimating the additive utility function. Therefore, we aim first to detect the preference ordering pattern rather than quantifying an evaluation function directly and then interpolate it.

As most of the existing opponent models in the literature do, our model assumes the opponent concedes over time. Initially, the agent does not know anything about its opponent's preferences; therefore, it creates a template estimation model according to the given domain configuration. In other words, the agent starts with an initial belief in ranking the issue values (V_i) and issues (W_i). The agent may assume that the opponent's value function is the opposite of its value function. For instance, If the agent prefers $V_1 >$ to V_2, it may consider that its opponent prefers V_2 to V_1. Alternatively, it may consider an arbitrary ordering for the opponent. Consider that we have n issues and for each issue i there are possible issue values denoted by $D_i = \{v_1^i, \ldots, v_m^i\}$. Assuming an arbitrary preference ordering for the opponent, (2) and (3) shows how the initial valuation values and issue weight are initiated. The agent keeps the issues and issue values in order in line with the estimated opponent preferences. Meaning that v_k^i is preferred over v_j^i by the opponent where where $k > j$. Accordingly, (2) assigns compatible evaluation values via max normalization. Similarly, the issue weights are initialized by sum normalization, where each issue weight is in the [0, 1] range, and their sum is equal to 1. Equation (3) ensures that their sum equals one.

$$v_j^i = \frac{j}{|D_i|} \tag{2}$$

$$W_i = \begin{cases} W_{i-1} + \frac{W_{i-1}}{n} & i > \frac{n}{2} \\ W_{i+1} - \frac{W_{i+1}}{n} & i < \frac{n}{2} \\ \frac{1}{n} & i = \lceil \frac{n}{2} \rceil \end{cases} \tag{3}$$

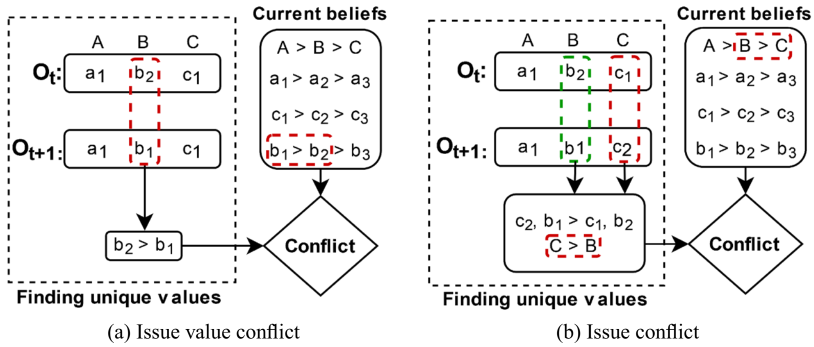
As the agent receives the opponent's offers during the negotiation, it updates its belief incrementally based on the inconsistency between the current model and the opponent's offers. To achieve this, it stores all bids made by the opponent so far, and when a new offer arrives, the current offer is compared with the previous bids with respect to any conflicting ordering. In particular, common and different values in the offer contents are detected. For different values, the system checks whether there is any conflicting situation with the current model. Recall that the current offer is expected to have the less preferred values since the model assumes that the opponent concedes over time. However, according to the learned model, the ordering may not match the expectation. In such a case, the model is updated.

Two types of conflict in the estimated model could be detected: *issue value conflict* and *issue conflict*. To illustrate those conflicts, let us examine some examples where agents negotiate over three issues (i.e., A, B, and C) in Fig. 1. As current belief indicates $b_1 \succ b_2 \succ b_3$ where b_i denotes a possible issue value for the issue B, b_1 is preferred b_2. In the given negotiation dialogue in Fig. 1a, it can be seen that the opponent's previous offer and current offers are $O_t =< a_1, b_2, c_1 >$ and $O_{t+1} =< a_1, b_1, c_1 >$, respectively. Agents can examine the contents of the offers and find unique value changes to make some inferences on the preferences. For our case, the only difference in the offers is the value of the issue B. Relying on the assumption that the human negotiator leans towards concession over time, the agent could infer $b_2 \succ b_1$. Recall that the most preferred values would appear early. As seen clearly, this preference ordering conflicts with that of the agent's belief. We call this type of conflict "issue value conflict" in our study.

The latter conflict type is about the importance of the issues. In the given an example in Fig. 1b, the consecutive offers involve more than one issue value difference, particularly on issues B and C. Then, the agent can deduce $(c_1, b_2) > (c_2, b_1)$ by relying on the concession assumption mentioned above. Individually, ordering in issue C is consistent with the belief (i.e., $c_1 \succ c_2$); however, the ordering on issue B is conflicting (i.e., $b_2 > b_1$). Therefore, in order to have $(c_1, b_2) \succ (c_2, b_1)$, the importance of the issue C should be higher than that of B (i.e., $C \succ B$). This inference conflicts with the current belief of the agent, which says B is more important than C.

Algorithm 1 shows how the ranking of the issue values is extracted. When the opponent makes an offer (O_c), the agent compares the content of the current offer with that of each offer in the opponent's offer history to find the unique values and consequently extract some preferential comparisons (Lines 1–4). Afterward, the agent keeps all those comparisons in a dictionary called CM (Line 3). By reasoning on each comparison in this set by considering the current belief set, each conflict is extracted and stored in AC (Line 5-7). It is worth noting that the method can find issue value conflicts consisting of multiple issues. After keeping track of all possible conflicts, the agent must determine how to update its beliefs. Counting the number of conflicts on each issue value pair, it considers the issue value orderings having the least conflicts and updates its belief accordingly (Lines 8-16). The agent detects issue value pairs in the conflict set for each issue and compares their occurrences to determine which one to stick on. For instance, if the agent observes conflicting information, the more frequent ordering becomes more dominant, and the agent adapts its beliefs accordingly. After updating the beliefs about issue value orderings, it does the same kind

Fig. 1 Preference conflict extraction example



of updates for the issue ordering (Lines 18-25). After finalizing the updates on the rankings, it estimates the utility space of the opponent by utilizing the update operations in (2) & (3).

To illustrate this, we trace the negotiation in Fig. 2, where we can observe how this opponent model works. Following the same domain, we first arbitrarily set our initial beliefs about preference ordering (e.g., $a_3 \succ a_2 \succ a_1$ for the values of issue A). The agent keeps track of offers made by the

Algorithm 1 Conflict-based Opponent Model (CBOM).

O: Offer history, O_c: Current Offer, B_c: Current Belief, I: Issues, V_i: Values;
UV: A list of unique values for a given offer pair;
CM: Comparison map of unique values per each offer pair;
AC: All conflicts extracted from comparison map;
VC: A list of conflicted value pairs for an issue;
IC: A list of conflicted issue pairs;
U_{opp}: The estimated opponent utility space;

1 **for each** $O_i \in O$ **do**
2 $UV \leftarrow$ findUniqueValues(O_i, O_c) ;
3 CM.append(UV);
4 **end**
5 **for each** $c \in CM$ **do**
6 AC.append(extractConflictsFrom(c, B_c));
7 **end**
8 **for each** $i \in I$ **do**
9 $VC \leftarrow$ getValueConflictbyIssue(AC, i);
10 **for each** $(p, r) \in VC$ **do** // $p, r \in D_i$
11 **if** $\|VC(r, p)\| > \|VC(p, r)\|$ **then**
12 $B_c(i) \leftarrow r \succ p$;
13 **else**
14 $B_c(i) \leftarrow p \succ r$;
15 **end**
16 **end**
17 **end**
18 $IC \leftarrow$ getIssueConflict(AC) ;
19 **for each** $(A, B) \in IC$ **do** // $A, B \in I$
20 **if** $\|IC(A, B)\| > \|IC(B, A)\|$ **then**
21 $B_c(I) \leftarrow A \succ B$;
22 **else**
23 $B_c(I) \leftarrow B \succ A$;
24 **end**
25 **end**
26 $U_{opp} \leftarrow$ estimateOppUtilitySpace(B_c);

opponent so far. In our example, you can see the offer history at time $t+2$. Following, the agent compares all previous offers with each other (i.e., pairwise comparison) and tries to extract an ordering relation. Here, O_t and O_{t+1} denote the first and second offer made by the opponent. Since the agent believes that the opponent's earlier offers are more preferred over the later offers, it extract that $a_1, b_2 \succ a_2, b_1$. This knowledge does not give any novel insight to update our beliefs, but we store this ordering for future analysis in the following rounds. When the opponent makes the offer O_{t+2}, the model compares it with all the previous offers pairwisely as well as the previously extracted information (e.g., the a_1, $b_2 \succ a_2$, b_1 relation). Starting from the first offer in the offer history (O_t-O_{t+2}), the model acquires the information of $a_1 \succ a_2$, since there is only one issue with a different value. When it compares O_{t+1} with O_{t+2}, it extracts ($b_1 \succ b_2$) and updates its beliefs accordingly. Similarly, the extracted information could be utilized to reason about the ordering of the negotiation issues (e.g., $A \succ B$) based on the contradiction between (a_1, $b_2 \succ a_2, b_1$) and ($b_1 \succ b_2$). Consequently, the agent deduces that the importance of issue A is more than issue B considering the assumption that one-issue comparisons are more reliable than multi-issue comparisons, which conflicts with the current belief and updates its belief accordingly.

4 Proposed conflict-based negotiation strategy

This section presents our negotiation strategy employing the opponent model mentioned above. This strategy incorporates the estimated opponent modeling into the Hybrid strategy [16], which estimates the target utility of the current offer based on time and behavior-based concession strategies.

The Algorithm 2 elaborates how the agent makes its decisions during the negotiation. In each round, it calculates a target utility by employing the hybrid bidding strategy. Consequently, it generates candidate offers that were not offered by the agent (i.e., CBOM Agent). Its utility is in the range of lower and upper target utility (i.e., TU_{cbom} - ϵ and TU_{cbom} + ϵ) (Lines 1–9). If there is no such an offer, the boundary is

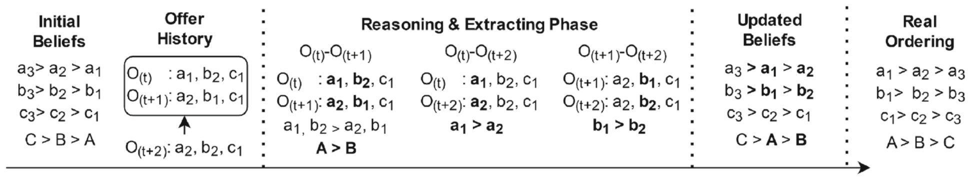


Fig. 2 Example process of the conflict-based opponent model (CBOM)

enlarged with a dynamically generated small number according to the domain size (Line 8). We define a round count n where we believe the agent has enough offers from its opponent to estimate their preferences. The value of n may vary depending on whether the agent negotiates with a human or agent negotiator. If the number of received offers from the opponent is less than n, the agent picks the offer maximizing its utility among potential offers (Lines 10–12). Shortly, the system does not engage the opponent model until there are enough offers accumulated in the history of opponent offers. Otherwise, the agent selects the offer whose estimated utility product is the maximum (Line 14). If the opponent made an offer with a utility higher than our lowest utility bid and the utility of the current candidate's offer (Line 16), the agent accepted its opponent's offer instead of making the offer. This acceptance condition is slightly more cooperative than the AC_{next} acceptance strategy. Otherwise, it makes the chosen offer (Line 19).

$$TU_{Hybrid} = (t^2) \times TU_{Times} + (1 - t^2) \times TU_{Behavior} \tag{4}$$

$$TU_{Times} = (1-t)^2 \times P_0 + [2 \times (1-t) \times t \times P_1] + t^2 \times P_2 \tag{5}$$

$$TU_{Behavior} = U(O_j^{t-1}) - \mu \times \Delta U \tag{6}$$

$$\Delta U = \sum_{i=1}^{4} [W_i \times (U(O_h^{t-i}) - U(O_h^{t-i-1}))] \tag{7}$$

$$\mu = P_3 + t \times P_3 \tag{8}$$

Equation (4) outlines how the agent computes the target utility for its upcoming offer, according to [16]. The concession function (TU_{Times}), represented by (5), incorporates t, the scaled time ($t \in [0, 1]$), and P_0, P_1, P_2, which correspond to the curve's maximum value, curvature, and minimum value, respectively. Note that the values of P_0, P_1, and P_2 in our experiment are 0.9, 0.7, and 0.4, respectively. The behavioral aspect of the "Hybrid" strategy involves scaling the overall utility change by a time-varying parameter, μ, to estimate the target utility, as demonstrated in (4). $U(O_a^{t-1})$ signifies the agent's utility for its preceding offer. Positive changes imply that the opponent has made concessions; hence, the agent should also make concessions. In (6), $U(O_a^{t-1})$ again represents the agent's utility for its prior offer. Positive changes indicate that the opponent has conceded, prompting the agent to make concessions. Considering the opponent's previous n bids, where W_i represents the weights of each utility difference, the behavior-based approach determines overall utility changes, as demonstrated in (7). Equation (8) reveals that the value of the coefficient μ is determined by the current time and P_3, which controls the degree of mimicry. Initially, the agent decreases or increases the target utility less than its opponent; subsequently, the degree of mimicry rises over time. Therefore, the "HybridAgent" strategy can smoothly conform with domains of varying sizes and harmonize with distinctive opponents utilizing behavior-based components of the HybridAgent. As an extension of the bidding strategy, CBOM agent also generates a target utility value by combining different p-values for various domain sizes. It cares about the social welfare score for both parties, choosing the most agreeable offer

Algorithm 2 Conflict-based negotiation strategy.

O_{space}: Offer space, n: Minimum number of round required for CBOM;
O_{opp}, O_{cbom}: Opponent's & CBOM agent's offer history, respectively ;
$U_{cbom}(o)$: The utility of an offer o for the CBOM agent ;
U_{opp}: The estimated utility space of the opponent ;
ϵ: Parameter controlling the bid utility window ;
TU_{Hybrid}: Calculated target utility with Hybrid Bidding Strategy ;

1 $O_{potential} \leftarrow \{\}$;
2 **while** $\|O_{potential}\| \succ 0$ **do**
3   **for each** $o \in O_{space}$ **do**
4   **if** $(U_{cbom}(o) \in [TU_{Hybrid} - \epsilon, TU_{Hybrid} + \epsilon])$ & $(o \notin O_{cbom})$ **then**
5       $O_{potential} \leftarrow O_{potential} + o$;
6   **end**
7   **end**
8   $\epsilon \leftarrow \epsilon + 0.01$;
9 **end**
10 **if** $\|O_{opp}\| < n$ **then**
11   $O_{cbom}^t \leftarrow \arg\max_o U_{cbom}(O_{potential})$;
12 **else**
13   $U_{opp} \leftarrow CBOM(O_{opp})$;
14   $O_{cbom}^t \leftarrow \arg\max_o U_{cbom}(O_{potential}) * U_{opp}(O_{potential})$;
15 **end**
16 **if** $(min(U(O_{cbom} \cup O_{cbom}^t)) \leq U(O_{opp}^t))$ **then**
17   Accept O_{opp}^t ;
18 **else**
19   *Return* O_{cbom}^t
20 **end**

from the list of offers that the opponent model creates. Thus, it is expected that CBOM agent can achieve higher utility while maximizing social welfare and finding quicker mutual agreement.

In accordance, Fig. 3 illustrates an example of the selected offer of the CBOM Agent according to its boundaries. The figure is structured with the y-axis representing the agent's utility and the x-axis representing the opponent's utility for the potential outcomes in the domain. Each outcome is depicted by blue dots on the graph. The red dot represents the target utility offer at a given time, indicating the preferred outcome for the agent. The red circle is drawn by adding the target utility to an epsilon value. Within this boundary, the agent selects the offer closest to the Nash offer (d_N), depicted in green. When the domain size is limited, only a few bids may remain within a specific utility range. Without enlarging the epsilon, the agent might end up repeating certain offers. The number of offers within the offer window should be increased in such situations. Expanding this boundary allows the agent to explore more offers, which helps avoid sending repetitive final offers to the opponent while still adhering to the target window. Examining other offers within the same window allows the agent to identify a more appropriate choice while upholding its target utility.

5 Experimental analysis

We first examine the performance of the proposed Conflict-based Opponent Model (CBOM) by conducting two different human experiments (Section 5.1) and extend this evaluation by considering the performance of the proposed strategy using this opponent model through agent-based negotiation simulations (Section 5.2).

5.1 Evaluation of opponent modeling via human-agent experiments

To show how well the proposed opponent modeling approach predicts the human opponent's preferences, we conducted experiments where participants negotiated with our agent on a given scenario to find a consensus within limited rounds by following the Alternating Offers Protocol (Section 3).

We consider the performance metrics to assess the quality of the predictions: Spearman's correlation and root-mean-square error (RMSE). The former metric indicates the accuracy of the predicted order of the outcomes according to the learned utility function, whereas the latter measures how accurate estimated utilities are. For correlation estimation, possible outcomes are sorted concerning the learned opponent model, and this ranking is compared with the actual ordering. Consequently, the Spearman correlation is calculated between the actual outcome ranking and the estimated one. The correlation would be high when both orderings are similar to each other. The correlation coefficient **r** ranges between -1 and 1, where the sign of the coefficient shows the direction, and the magnitude is the strength of the relationship. For RMSE, the utility of each outcome is estimated according to the learned model, and the error in the prediction is calculated (See (9)). When the estimated utility values are close to the actual utility values, the RSME values would be low. In summary, low RSME and high correlation values are desired in our case.

$$RMSE = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}\left(U(o_i) - \hat{U}(o_i)\right)^2} \tag{9}$$

Baarslag et al. compare the performance of the existing opponent models in automated negotiation [5]. Their results show that frequentist-based opponent modeling approaches are the most effective among the existing ones despite the approach's simplicity. Therefore, we use a benchmark involving two different state-of-the-art frequentist opponent modeling approaches widely used in automated negotiation employed in HardHeaded [19] and Scientist [31] agents to evaluate the performance of the proposed opponent model. Frequentist opponent modeling techniques mostly rely on heuristics, assuming that the opponent would concede less on the essential issues and the preferred values appear more often than less preferred ones. Consequently, they check the frequency of each issue value's appearance in the offers. Furthermore, they compare the content of the consecutive offers and find out the issues with changed values. In other words, if the value of an issue is changed in the opponent's consecutive offers, the weights of those issues are decreased by a certain amount (i.e., becoming less critical). In the Scientist Agent, Tunalı et al. aims to resolve some update problems and enhance the model by comparing a group of offer exchanges instead of only consecutive pairs of offers and adopting a decayed weight update mechanism. Each opponent model is fed and updated in each round by simulating the negotiation data obtained from human-agent negotiation experiments. At the end of each negotiation, the estimated models are evaluated according to the RMSE and Spearman correlation metrics explained above.

5.1.1 Study 1: human-agent negotiation in deserted Island scenario

We analyzed and utilized the negotiation log data collected during the human-agent negotiation experiments in

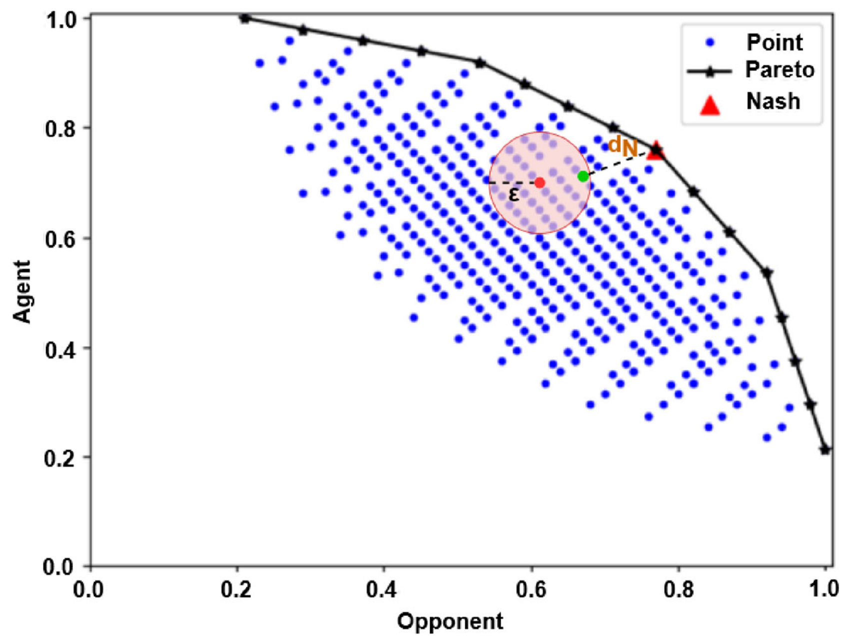


Fig. 3 Offer selection example of the CBOM Agent according to ϵ boundary

Table 1 Agent's and participants' preference profiles in deserted Island scenario

Items	First Negotiation Agent	First Negotiation Participant	Second Negotiation Agent	Second Negotiation Participant
Compass	13	5	6	13
Container	22	20	13	5
Food	17	7	20	22
Hammer	6	13	5	10
Knife	5	10	10	17
Match	20	22	7	6
Medicine	7	6	17	7
Rope	10	17	22	20

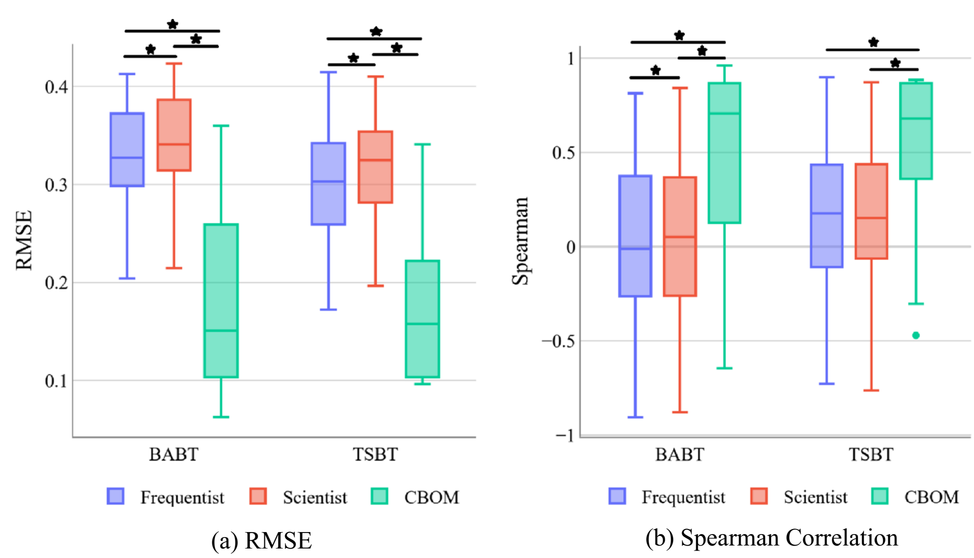


Fig. 4 RMSE & Spearman Correlations for the experiment in Island Scenario ($\star$ represents $p < 0.001$)

Table 2 Preference profiles for grocery negotiation sessions

Items	First Negotiation Agent	First Negotiation Participant	Second Negotiation Agent	Second Negotiation Participant
Watermelon	4	12	12	4
Banana	1	8	8	1
Orange	12	4	4	12
Apple	8	1	1	8

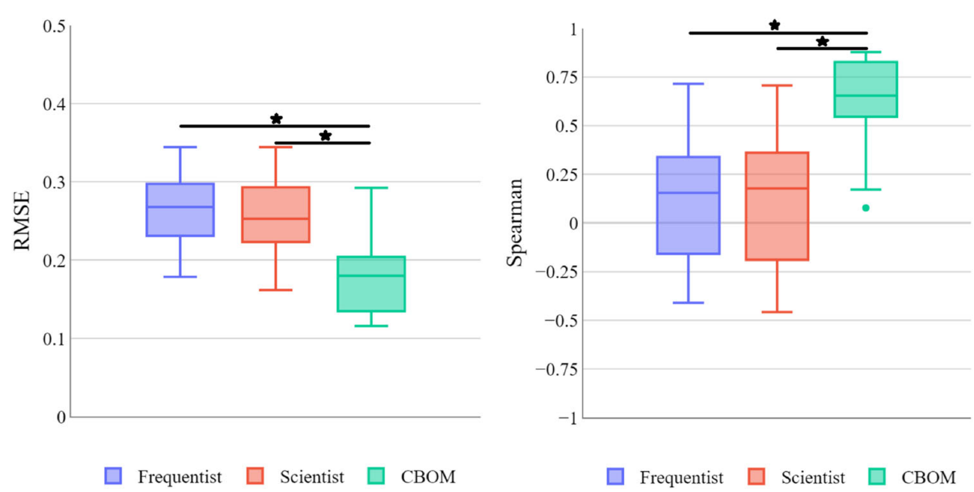
[4], where the participants negotiated on a particular scenario called "Deserted Island". They negotiated resource allocation based on the division of eight survival products by two partners who fell on the deserted island. Each participant attended two negotiation sessions where the utility distributions of the issues were the same, but the orderings differed. Table 1 shows the preference profiles for both sessions. During the experiments, participants only know their preferences, and so does the agent. In this study, 42 participants (21 men, 21 women, median age: 23) were included and asked to negotiate with our agent on a face-to-face basis, and the agent made counteroffers. Offer exchanges in both sessions were recorded separately for each session. At the end of this data collection process, 46 sessions using the time-based stochastic bidding tactic (TSBT) and 38 sessions using the behavior-based adaptive bidding tactic (BABT) were obtained. The average negotiation rounds to reach an agreement was 14.84, with a standard deviation of 5.2.

Figure 4 shows box plots for each opponent modeling technique's RMSE and Spearman correlation values. As far as the correlation values are concerned, it can be said that *CBOM*'s ranking predictions are better than Scientist and Frequentist (See Fig. 4b). To apply the appropriate statistical significance test, we first check the normality of the data distribution via the Kolmogorov-Smirnov normality test and then the homogeneity of variance via Levene's Test. We applied the dependent sample t-test or the Wilcoxon Signed Rank test, depending on the results. If the data distribution passes these tests, the paired t-test is applied; otherwise, a non-parametric statistics test, namely the Wilcoxon-Signed Rank test. All statistical test results are given at the 99% confidence interval (i.e., $\alpha = 0.01$). When we apply the statistical tests, it is seen that *CBOM*'s ranking performance is statistically significantly better than others ($p < 0.01$). Similarly, the errors on the estimated utilities via *CBOM* are lower than the errors via other approaches (see Fig. 4a). Furthermore, it is seen that *Scientist* statistically significantly performed better than *Frequentist* for both metrics except when the agent employs the TSBT strategy.

5.1.2 Study 2: human-agent negotiation in grocery scenario

In this part, we analyzed and utilized the negotiation log data collected during another human-agent negotiation experiment in [16] where the participants negotiated on a particular scenario called "Grocery". Different from the first study, the negotiation domain does not consist of binary resource items (i.e., *allocate* or *not allocate*). Instead, the negotiation parties negotiate on the number of items to be allocated (i.e., how many items will be allocated). In this scenario, there are four types of fruits, where each participant can have up to four

Fig. 5 RMSE & Spearman Correlations for the experiment in Grocery Scenario (⋆ represents $p < 0.001$)



(a) RMSE

(b) Spearman Correlation

Table 3 Negotiation scenarios in automated negotiation settings

Domain (ANAC Year)	# of Values for Each I	Total Bids	Opposition
Car (2015)	3, 4, 4, 5	240	0.209
Smart Energy Grid (2016)	5, 5, 5, 5	625	0.362
Grocery (2011)	4, 4, 4, 5, 5	1,600	0.354
Party (2012)	3, 4, 4, 4, 4, 4	3,072	0.191
Politics (2015)	2, 3, 3, 4, 4, 4, 4, 5	23,040	0.221
Supermarket (2012)	4, 6, 6, 7, 7, 7	49,392	0.155

of each, and the opponent gets the rest. The participants aim to find an adequate division of the fruits. Table 2 shows the agent and participant's preference profiles for both sessions. It is worth noting that each party only knows its scores. In this experiment, the participants negotiated against an agent employing the hybrid strategy where TSBT and BABT strategies are used together for bidding. 28 participants attended two negotiation sessions where all negotiation sessions ended with an agreement, thus, totaling up to 56 negotiation sessions against the agent. The average negotiation rounds to reach an agreement was 19.39, with a standard deviation of 11.82.

Similar to the previous study, we update each opponent modeling by using the offer exchanges by the human participants and calculate the error and correlation values with the final model at the end of each negotiation. Figure 5 shows box plots for RMSE and Spearman correlation values per each opponent modeling technique in this scenario. We can conclude that *CBOM* statistically significantly outperformed others, whereas *Frequentist* performs better than *Scientist* when we analyze the statistical test results. Those results are in line with the first study and strongly show the success of the proposed opponent modeling in human-agent negotiations. It is worth noting that the prediction error in the grocery scenario is lower than in the island scenario. Although the number of possible outcomes in these scenarios is the same (256), the number of issues in grocery scenarios is lower than in the island scenario. Therefore, one can intuitively think it is easier to predict the evaluation values in the grocery scenario compared to island scenarios. In addition, this study's average number of rounds is higher (19.39 versus 14.84). When we receive more offers, the model's accuracy may increase depending on the model.

5.2 Evaluation of the CBOM agent via automated negotiation experiments

In this section, we evaluate the performance of our agent employing the proposed *CBOM* opponent modeling by comparing its performance with that of the state-of-the-art negotiating agents available in automated negotiation literature. We built a rich benchmark of 15 successful negotiating agents who competed in the International Automated Negotiating Agents Competition ANAC [15] between 2011 and 2017. We ran negotiation tournaments in Genius, where each agent bilaterally negotiated on various negotiation scenarios. Six negotiation scenarios were used during the tournament, and the details of those scenarios are given in Table 3. As can be seen, the size and opposition degree of preference profiles in the given scenario is different. The size of the scenarios determines the search space. The larger the search space is, the more difficult it might be to estimate an accurate model based on the opponent's offers exchanges. Next, the opposition is valuable information regarding understanding the domain's capacity to satisfy both parties [5]. That is, it indicates how difficult it is to find a consensus. Taking the opposition of the preference profiles into account while analyzing the negotiation results may help us get an insight into how well the proposed negotiation strategy is in terms of social welfare with varying difficulties in finding an agreement.

Table 4 Average Spearman and RMSE results for six domains

Domains	SPEARMAN			RMSE		
	CBOM	Scientist	Hardheaded	CBOM	Scientist	Hardheaded
Car	**0.73 ± 0.1**	0.29 ± 0.0	0.33 ± 0.0	**0.15 ± 0.0**	0.31 ± 0.0	0.27 ± 0.0
Energy Grid	**0.68 ± 0.1**	0.25 ± 0.0	0.25 ± 0.0	**0.16 ± 0.0**	0.22 ± 0.0	0.30 ± 0.0
Grocery	0.81 ± 0.0	**0.85 ± 0.0**	0.82 ± 0.0	**0.12 ± 0.0**	0.17 ± 0.0	0.20 ± 0.0
Party	**0.90 ± 0.0**	0.82 ± 0.0	0.50 ± 0.0	**0.10 ± 0.0**	0.13 ± 0.1	0.26 ± 0.0
Politics	0.82 ± 0.0	**0.86 ± 0.0**	0.79 ± 0.0	**0.20 ± 0.0**	0.22 ± 0.0	0.16 ± 0.0
Supermarket	**0.66 ± 0.0**	0.58 ± 0.0	0.56 ± 0.0	**0.21 ± 0.0**	0.18 ± 0.0	0.30 ± 0.0

Table 5 Average agent individual utility

Agent Name	Car	Energy Grid	Grocery	Party	Politics	Supermarket	Average ± Std
CBOMAgent	0.833	**0.777**	**0.846**	0.838	0.594	0.868	0.793 ± 0.09
Atlas3	0.875	0.758	0.828	0.809	0.578	0.848	0.783 ± 0.10
AgentKN	0.832	0.728	0.793	0.820	**0.609**	0.756	0.756 ± 0.07
NiceTitForTat	0.804	0.680	0.767	**0.848**	0.569	0.853	0.753 ± 0.10
HardHeaded	**0.882**	0.670	0.795	0.832	0.533	0.741	0.742 ± 0.12
ParsCat	0.800	0.726	0.792	0.833	0.501	0.767	0.736 ± 0.11
CUHKAgent	0.750	0.684	0.763	0.800	0.543	0.765	0.717 ± 0.09
IAMcrazyHaggler	0.836	0.546	0.819	0.771	0.416	**0.872**	0.710 ± 0.17
IAMhaggler2012	0.782	0.496	0.796	0.841	0.491	0.792	0.700 ± 0.15
PonPokoAgent	0.773	0.606	0.785	0.783	0.427	0.802	0.696 ± 0.14
Caduceus	0.731	0.546	0.801	0.703	0.489	0.711	0.663 ± 0.11
ParsAgent2	0.779	0.540	0.713	0.698	0.424	0.805	0.660 ± 0.14
Boulware	0.712	0.611	0.671	0.752	0.460	0.642	0.641 ± 0.09
AgentX	0.672	0.582	0.655	0.801	0.347	0.642	0.616 ± 0.14
YXAgent	0.770	0.553	0.721	0.625	0.397	0.553	0.603 ± 0.12
Conceder	0.598	0.505	0.391	0.649	0.306	0.335	0.464 ± 0.13

We formed a pool of agents involving our Conflict-based agent and the ANAC finalists in different categories. We ran a tournament in Genius where each agent bilaterally negotiated with each other on scenarios described in Table 3. The ANAC agents used in this evaluation are listed as follows:

- **Boulware** and **Conceder** are baseline agents available in Genius framework.
- **Hardheaded** [19] was the winner of individual utility category in ANAC 2011.
- **NiceTitForTat** [7] was the finalist of individual utility category in ANAC 2011.
- **CUHKAgent** [11] was the winner of individual utility category in ANAC 2012.
- **IAmHaggler2012** [15] was the winner of the Nash category in ANAC 2012.

Table 6 Average Nash Distances

Agent name	Car	Energy Grid	Grocery	Party	Politics	Supermarket	Average ± Std
CBOMAgent	**0.077**	**0.104**	**0.068**	**0.120**	0.311	**0.058**	**0.123 ± 0.09**
Atlas3	0.183	0.216	0.105	0.230	0.253	0.198	0.197 ± 0.05
AgentKN	0.139	0.194	0.166	0.213	0.250	0.229	0.198 ± 0.04
ParsCat	0.133	0.173	0.130	0.156	0.438	0.178	0.201 ± 0.11
NiceTitForTat	0.154	0.209	0.178	0.230	**0.187**	0.250	0.201 ± 0.03
IAMhaggler2012	0.115	0.500	0.095	0.152	0.363	0.145	0.228 ± 0.15
HardHeaded	0.142	0.357	0.183	0.198	0.487	0.279	0.274 ± 0.12
AgentX	0.210	0.273	0.230	0.267	0.418	0.262	0.277 ± 0.07
Boulware	0.217	0.271	0.254	0.278	0.326	0.334	0.280 ± 0.04
IAMcrazyHaggler	0.156	0.495	0.176	0.156	0.686	0.064	0.289 ± 0.22
Caduceus	0.190	0.462	0.178	0.238	0.431	0.254	0.292 ± 0.11
PonPokoAgent	0.201	0.392	0.177	0.205	0.640	0.154	0.295 ± 0.17
CUHKAgent	0.361	0.428	0.280	0.205	0.457	0.261	0.332 ± 0.09
ParsAgent2	0.197	0.474	0.274	0.306	0.576	0.210	0.340 ± 0.14
YXAgent	0.268	0.526	0.350	0.376	0.718	0.450	0.448 ± 0.15
Conceder	0.321	0.360	0.534	0.459	0.458	0.652	0.464 ± 0.11

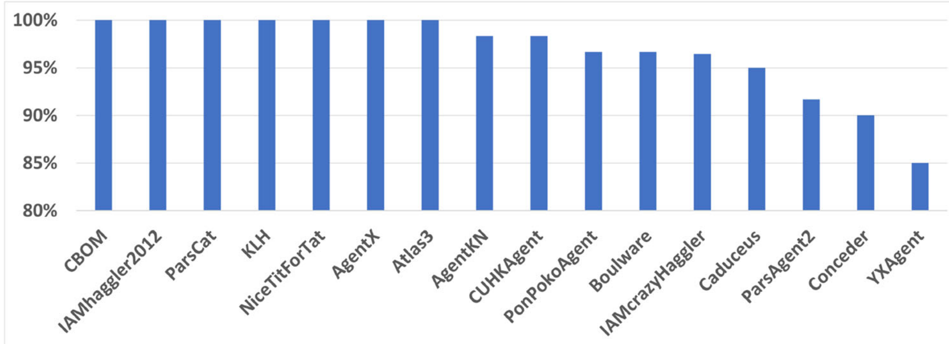


Fig. 6 Agreement rates of agents

- **Atlas3** [24] was the winner of individual utility category in ANAC2015, .
- **ParsAgent2** [17] was the winner of the Nash category in ANAC 2015.
- **AgentX** [9] was fourth of the Nash category in ANAC 2015.
- **Caudeceus** [10] was the winner of individual utility category in ANAC 2016.
- **YXAgent** [3] was the second of individual utility category in ANAC 2016.
- **PonPoko Agent** [3] was winner of individual utility category in ANAC 2017.
- **AgentKN** [3] was the second of the Nash category in ANAC 2017.

In order to study how well our opponent model performs when it negotiates with automated negotiating agents, we compare the performance of opponent models used in Conflict-based (CBOM), Scientist, and HardHeaded by integrating those opponent models into our negotiation strategy. We calculated the Spearman correlation between the actual and estimated ranks of the outcomes per each scenario and reported their averages. Note that the higher correlation is, the better the prediction is. Table 4 shows those Spearman correlations and RMSE in the utility calculations where the best scores are boldfaced. It is seen that CBOM is more successful than others in terms of Spearman correlation, except for the results obtained in the grocery and politics domains. Furthermore, RMSE results show that the CBOM is more successful in all domains.

Next, we analyze the performance of the proposed negotiation strategy relying on the CBOM opponent modeling against the ANAC finalists. The most widely used performance metric in negotiation is the final received utility, which is intuitive and in line with Kiruthika's approach to Multi-Agent Negotiation systems [18]. There are other metrics,

Table 7 Average agreement rounds

Agent name	Car	Energy Grid	Grocery	Party	Politics	Supermarket	Average ± Std
AgentX	**205**	**816**	**440**	**477**	**1,868**	**162**	**661 ± 580**
Conceder	1,222	1,826	2,626	2,017	2,559	3,095	2,224 ± 611
IAMhaggler2012	2,765	4,393	2,096	2,829	4,146	2,378	3,101 ± 864
ParsCat	2,899	3,988	2,741	3,034	4,254	2,567	3,247 ± 639
CBOMAgent	3,506	3,859	2,978	3,452	4,407	2,403	3,434 ± 633
IAMcrazyHaggler	3,526	4,202	2,984	3,104	4,489	2,553	3,476 ± 682
PonPokoAgent	3,673	4,321	3,330	3,564	4,489	3,202	3,763 ± 481
Boulware	3,604	4,005	3,650	3,705	4,317	3,560	3,807 ± 270
NiceTitForTat	3,367	4,002	4,061	3,654	4,142	4,053	3,880 ± 277
AgentKN	3,612	4,054	3,809	3,898	4,362	3,712	3,908 ± 246
Atlas3	4,252	4,473	2,298	4,355	4,369	3,973	3,953 ± 756
YXAgent	3,907	4,171	3,597	4,053	4,565	3,754	4,008 ± 312
ParsAgent2	3,658	4,249	3,870	4,024	4,527	3,736	4,011 ± 301
Caduceus	4,070	4,501	3,628	4,148	4,473	4,012	4,139 ± 296
HardHeaded	4,099	4,399	3,968	4,196	4,646	4,243	4,258 ± 217
CUHKAgent	4,301	4,458	4,512	4,272	4,659	4,236	4,406 ± 150

such as nearness to Pareto optimal solutions/Kalai point/the Nash point, the sum of both agents' agreement utility (i.e., social welfare), and the product of those agreement utilities. Accordingly, we evaluate the performances regarding average individual received utility, Nash distance, and social welfare.

First, we analyze the average individual utilities received by each agent. Table 5 shows those utilities per each agent in each negotiation scenario where the highest scores are boldfaced. The last column shows the average scores of each agent in all domains. Our agent took in the first top three agents. We noticed that the worst performance of our agent was in the supermarket domain, where the outcome space is too large to search. Moreover, our agent performed well in the smart energy grid and grocery scenarios, whose opposition levels are high.

Table 6 shows the average Nash distance for each agent in all scenarios separately, and the final column indicates the average of all scenarios. Here, the lower the Nash distance is, the fairer the agent's outcomes are. Our conflict-based agent outperformed the ANAC finalist agents except for Nice TitForTat, which is known for maximizing social welfare in the Politics scenario (See Table 8 in Appendix). Similar results were obtained when we analyzed the social welfare in terms of summation of both agents' agreement utilities (See Appendix). Overall results support the success of our agent, and the reason may stem from the fact that our agent aims to learn its opponent's preferences over time and aims to find win-win solutions for both sides.

When we investigate the overall agreement rate, it can be seen that most of the agents have a high acceptance rate, and the leading ones, like ours and Atlas3, found agreements in all negotiations, as seen in Fig. 6. The final metric that we investigated is the average rounds to reach an agreement. In our experiments, the deadline is set to 5000 rounds per negotiation scenario. Table 7 shows the average rounds that the agent reached their agreement. It can be observed that the the size of the outcome space and the opposition level may influence the agreement round. In large and competitive scenarios, agents needed more rounds to reach an agreement. Among all agents, Agent X tended to reach a consensus sooner than all other agents. Furthermore, IAMHaggler2012 and ParsCat agents tend to explore the offered space as much as they can in the given time. Therefore, these are the agents least affected by the size of the outcome space and its competitiveness. Our conflict-based agent could reached an agreement sooner than more than half of the agents but it is worth noting that it took more time in terms of seconds due to its computational complexity similar to AgentKN (See Table 9 in Appendix).

As a result of all the automated negotiations, we determined the six most successful agents in both the individual and fairness category. Figure 7 shows clearly that our agent gains the highest individual gain while having a fairer win-win solution (i.e., minimum distance to Nash solution). It is worth noting that while having high utility, our agent lets its opponent gains relatively high utility in contrast to other top agents.

6 Conclusion and future work

In conclusion, this work presents a conflict-based opponent modeling approach and a bidding strategy employing this model for bilateral negotiations. Apart from evaluating the performance of the proposed opponent model in two different human-negotiation experiment settings, the proposed strategy was also tested against the finalist of the ANAC agents considering various performance metrics such as individual utility and distance to the Nash solution. Our results show that the proposed approach outperformed the state of the negotiating agents, and the proposed opponent model performed better than other frequency-based models. The contribution of this study is twofold: (1) introducing a novel opponent modeling approach to learn human negotiators' preferences from limited bid exchanges and (2) presenting a suitable bidding strategy relying on the proposed opponent model for both collaborative and competitive negotiation settings.

Due to the algorithm's complexity, the agent's performance decreases when the outcome space becomes more extensive or the number of generated offers made by the opponent increases in automated negotiation. We are planning to reduce the computational complexity of the opponent modeling by adopting dynamic programming properties and local search. The upcoming study will focus on opponent model strategies that decrease the human-agent negotiation duration with the optimal number of rounds. It would be interesting to create stereotype profiles by mining the previous negotiation history and matching the current opponent's profile based on their recent offer exchanges.

Understanding and discovering the opponent's preferences over negotiation may play a key role in adopting strategic bidding strategies to find mutually beneficial agreements. However, as stated before, it is challenging to create a mental model for the opponent's preferences based on a few bid exchanges. In contrast to automated negotiation, the number of exchanged bids is limited in human-agent negotiation. That requires a bidding strategy smartly exploring the potential bids and building upon an opponent model, capturing the

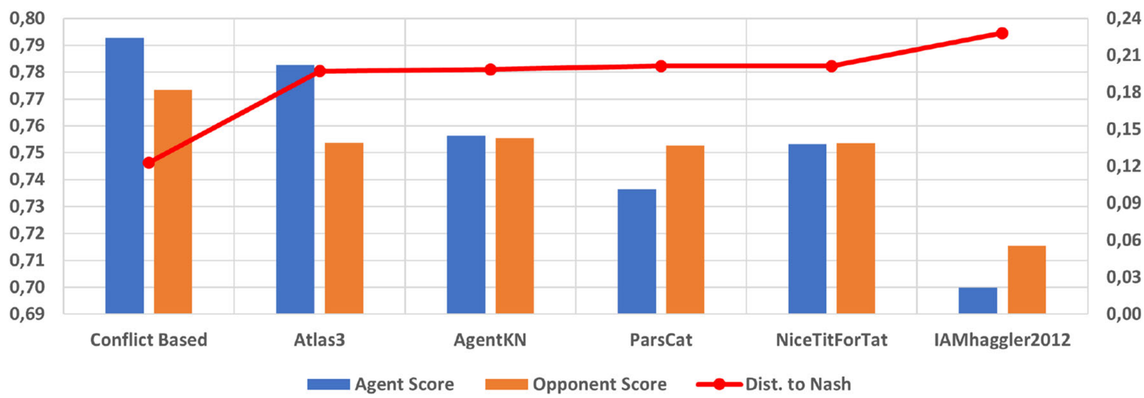


Fig. 7 Best six agents in all automated negotiation results

critical components of the opponent's preferences. While creating such modeling is not trivial with limited bid exchanges, the agent can exploit its previous negotiation experiences and take advantage of repeated patterns. As future work, it would be interesting to create different mental models from previous negotiation experiences by applying our model and trying to detect which mental model fits better for the current human negotiators. Consequently, instead of starting to learn from scratch, our model can enhance the chosen model by analyzing the current bid exchanges. Furthermore, the agent can exploit different types of inputs, such as the opponent's arguments and facial expressions, to enhance opponent modeling.

Appendix

Table 8 Average social welfare score

Agent Name	Car	Energy Grid	Grocery	Party	Politics	Supermarket	Average ± Std
CBOMAgent	165,655	**157,227**	150,312	**167,216**	106,945	**172,876**	**153,372 ± 21,995**
AgentX	162,310	149,437	**157,913**	157,453	124,845	159,043	151,834 ± 12,678
Atlas3	159,891	151,089	153,578	153,295	126,653	157,466	150,329 ± 10,976
AgentKN	166,027	150,985	133,303	156,921	126,087	150,364	147,281 ± 13,615
ParsCat	166,225	152,253	145,902	164,745	85,448	161,411	145,997 ± 27,999
NiceTitForTat	165,437	144,536	129,573	152,589	**127,955**	145,374	144,244 ± 12,914
IAMhaggler2012	161,990	93,551	156,376	165,018	93,827	164,709	139,245 ± 32,337
HardHeaded	**166,258**	120,624	131,202	161,969	85,146	142,000	134,533 ± 27,260
Boulware	152,574	134,493	118,818	141,937	116,171	127,207	131,867 ± 12,750
IAMcrazyHaggler	155,615	95,329	136,363	157,082	58,579	172,290	129,210 ± 39,828
PonPokoAgent	149,828	108,912	132,648	154,698	62,888	158,976	127,992 ± 33,601
Caduceus	151,600	100,644	130,134	144,040	93,133	146,992	127,757 ± 22,888
CUHKAgent	130,850	120,066	114,833	157,693	92,766	145,382	126,932 ± 21,067
ParsAgent2	150,379	100,984	115,091	135,542	69,915	154,718	121,105 ± 29,599
Conceder	136,634	128,370	81,617	112,968	114,208	68,503	107,050 ± 24,320
YXAgent	139,677	93,292	111,212	121,324	54,423	107,581	104,585 ± 26,484

Table 9 Average agreement run times

Agent Name	Car	Energy Grid	Grocery	Party	Politics	Supermarket	Average ± Std
AgentX	**9,831**	**18,626**	**3,413**	**7,771**	**81,721**	**23,662**	**24,171 ± 26,610**
Conceder	17,312	29,410	19,058	27,922	114,666	213,279	70,275 ± 72,241
Boulware	40,855	50,588	32,838	43,204	142,281	189,926	83,282 ± 60,379
ParsCat	36,405	66,436	30,499	41,597	172,065	184,158	88,527 ± 64,418
PonPokoAgent	47,276	66,870	40,068	47,167	169,917	177,437	91,456 ± 58,747
HardHeaded	46,853	57,890	34,569	53,148	165,997	198,421	92,813 ± 64,299
Atlas3	52,283	72,105	18,138	66,174	161,367	199,162	94,872 ± 63,694
CUHKAgent	50,303	55,542	38,336	53,566	168,014	214,496	96,709 ± 68,404
YXAgent	55,307	75,352	46,004	59,929	194,091	162,102	98,797 ± 57,485
IAMcrazyHaggler	57,375	80,281	42,139	44,002	199,154	175,726	99,780 ± 63,581
IAMhaggler2012	78,696	124,397	63,920	88,375	226,875	231,055	135,553 ± 68,526
ParsAgent2	154,667	164,868	46,388	117,370	227,899	254,998	161,032 ± 68,806
Caduceus	48,768	70,825	62,566	101,401	521,733	508,281	218,929 ± 209,987
NiceTitForTat	48,707	90,214	118,381	163,795	494,087	748,441	277,271 ± 256,374
CBOMAgent	149,535	248,868	116,199	183,300	626,341	937,576	376,970 ± 302,796
AgentKN	240,108	322,627	104,224	149,689	776,897	863,015	409,427 ± 299,331

Funding This work has been supported by a grant of The Scientific and Research Council of Turkey (TÜBİTAK) with grant number 118E197 and partially supported by the CHIST- ERA grant CHIST-ERA-19-XAI-005, the Scientific and Research Council of Turkey (TÜBİTAK, G.A. 120N680).

Declarations

Conflicts of interest The authors declare that they have no conflict of interest.

References

1. Amini M, Fathian M, Ghazanfari M (2020) A boa-based adaptive strategy with multi-party perspective for automated multilateral negotiations. Appl Intell 50. https://doi.org/10.1007/s10489-020-01646-y
2. Aydoğan R, Festen D., Hindriks K, Jonker C (2017) Alternating Offers Protocols for Multilateral Negotiation, pp 153–167. Studies Comput Intell Springer Sci. https://doi.org/10.1007/978-3-319-51563-2_10
3. Aydoğan R, Fujita K, Baarslag T, Jonker CM, Ito T (2021) ANAC 2017: Repeated multilateral negotiation league. Advances in automated negotiations. Springer Singapore, Singapore, pp 101-115
4. Aydoğan R, Keskin MO, Çakan U (2022) Would you imagine yourself negotiating with a robot, jennifer? why not? IEEE Trans Hum Mach Syst 52(1):41–51. https://doi.org/10.1109/THMS.2021.3121664
5. Baarslag T, Hendrikx M, Hindriks K, Jonker C (2013) Predicting the performance of opponent models in automated negotiation. In: 2013 IEEE/WIC/ACM International joint conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), vol 2, pp 59–66. https://doi.org/10.1109/WI-IAT.2013.91
6. Baarslag T, Hendrikx MJC, Hindriks K, Jonker C (2016) A survey of opponent modeling techniques in automated negotiation. In: AAMAS
7. Baarslag T, Hindriks K, Jonker C (2013) A tit for tat negotiation strategy for real-time bilateral negotiations. In: Complex automated negotiations: theories, models, and software competitions. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 229–233. https://doi.org/10.1007/978-3-642-30737-9_18
8. Fatima S, Kraus S, Wooldridge M (2014) Principles of automated negotiation. Cambridge University Press. https://doi.org/10.1017/CBO9780511751691
9. Fujita K, Aydoğan R, Baarslag T, Hindriks K, Ito T, Jonker C (2017) The Sixth automated negotiating agents competition (ANAC 2015). Springer International Publishing, Cham, pp 139–151 (2017). https://doi.org/10.1007/978-3-319-51563-2_9
10. Güneş TD, Arditi E, Aydoğan R (2017) Collective voice of experts in multilateral negotiation. PRIMA 2017: Principles and practice of multi-agent systems. Springer International Publishing, Cham, pp 450–458
11. Hao J, Leung Hf (2014) CUHKAgent: An Adaptive negotiation strategy for bilateral negotiations over multiple items. Springer Japan, Tokyo, pp 171–179. https://doi.org/10.1007/978-4-431-54758-7_11
12. Hindriks K, Jonker C, Tykhonov D (2009) The benefits of opponent models in negotiation. In: 2009 IEEE/WIC/ACM International joint conference on Web Intelligence and Intelligent Agent Tech-

nology (WI-IAT). pp 439–444. https://doi.org/10.1109/WI-IAT.2009.192
13. Hosokawa Y, Fujita K (2020) Opponent's preference estimation considering their offer transition in multi-issue closed negotiations. IEICE Trans Inf Syst E103.D, 2531–2539 (2020). https://doi.org/10.1587/transinf.2020SAP0001
14. Jonge Dd, Sierra C (2015) NB3: a multilateral negotiation algorithm for large, non-linear agreement spaces with limited time. Auton Agents Multi-Agent Syst 29(5):896–942. https://doi.org/10.1007/s10458-014-9271-3
15. Jonker C, Aydogan R, Baarslag T, Fujita K, Ito T, Hindriks K (2017) Automated negotiating agents competition (ANAC). Proceedings of the AAAI Conference on Artificial Intelligence 31(1). https://ojs.aaai.org/index.php/AAAI/article/view/10637
16. Keskin MO, Çakan U, Aydoğan R (2021) Solver agent: Towards emotional and opponent-aware agent for human-robot negotiation. In: Proceedings of the 20th international conference on Autonomous Agents and MultiAgent Systems, AAMAS 21. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, pp 1557–1559
17. Khosravimehr Z, Nassiri-Mofakham F (2017) Pars agent: hybrid time-dependent, random and frequency-based bidding and acceptance strategies in multilateral negotiations. Springer International Publishing, Cham, pp 175–183. https://doi.org/10.1007/978-3-319-51563-2_12
18. Kiruthika U, Somasundaram TS, Subramanian KS (2020) Lifecycle model of a negotiation agent: A survey of automated negotiation techniques. Group Decision and Negotiation 29:1–24. https://doi.org/10.1007/s10726-020-09704-z
19. van Krimpen T, Looije D, Hajizadeh S (2013) HardHeaded. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 223–227. https://doi.org/10.1007/978-3-642-30737-9_17
20. Lewis M, Yarats D, Dauphin Y, Parikh D, Batra D (2017) Deal or no deal? end-to-end learning of negotiation dialogues. In: Proceedings of the 2017 conference on empirical methods in natural language processing. Assoc Comput Ling, Copenhagen, Denmark, pp 2443–2453. https://doi.org/10.18653/v1/D17-1259
21. Lin R, Kraus S, Wilkenfeld J, Barry J (2008) Negotiating with bounded rational agents in environments with incomplete information using an automated agent. Artif Intell 172(6–7):823–851
22. Mell J, Gratch J, Aydoğan R, Baarslag T, Jonker CM (2019) The likeability-success tradeoff: Results of the 2nd annual human-agent automated negotiating agents competition. In: 2019 8th international conference on affective computing and intelligent interaction (ACII). pp 1–7. https://doi.org/10.1109/ACII.2019.8925437
23. Mell J, Lucas GM, Mozgai S, Gratch J (2020) The effects of experience on deception in human-agent negotiation. J Artif Intell Res 68:633–660
24. Mori A, Ito T (2017) Atlas3: A negotiating agent based on expecting lower limit of concession function. Springer International Publishing, Cham, pp 169–173. https://doi.org/10.1007/978-3-319-51563-2_11
25. Nazari Z, Lucas G, Gratch J (2015) Opponent modeling for virtual human negotiators. In: International conference on intelligent virtual agents. pp 39–49. https://doi.org/10.1007/978-3-319-21996-7_4
26. Oshrat Y, Lin R, Kraus S (2009) Facing the challenge of human-agent negotiations via effective general opponent modeling. Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS 2:377–384. https://doi.org/10.1145/1558013.1558064
27. Peled N, Gal YK, Kraus S (2015) A study of computational and human strategies in revelation games. Auto Agents Multi-Agent Syst 29(1):73–97
28. Rosenfeld A, Zuckerman I, Segal E, Drein O, Kraus S (2014) Negochat: a chat-based negotiation agent. In: AAMAS. pp 525–532
29. Sanchez-Anguix V, Aydoğan R, Julian V, Jonker C (2014) Unanimously acceptable agreements for negotiation teams in unpredictable domains. Elec Comm Res Appl 13(4):243–265
30. Sengupta A, Mohammad Y, Nakadai S (2021) An autonomous negotiating agent framework with reinforcement learning based strategies and adaptive strategy switching mechanism. In: Proceedings of the 20th international conference on autonomous agents and multiAgent systems, AAMAS '21. pp 1163–1172
31. Tunalı O, Aydoğan R, Sanchez-Anguix V: Rethinking frequency opponent modeling in automated negotiation. In: International. pp 263–279 (2017). https://doi.org/10.1007/978-3-319-69131-2_16
32. Yu C, Ren F, Zhang M (2013) An adaptive bilateral negotiation model based on bayesian learning. Springer, vol 435, pp 75–93. https://doi.org/10.1007/978-3-642-30737-9_5

Mehmet Onur Keskin obtained his B.Sc. in Computer Science and MSc. in Artificial Intelligence from Özyegin University, Istanbul. As a research assistant at the OZU Interactive Intelligence Systems Laboratory since 2017, Keskin has been focused on advancing human-agent/robot interaction frameworks for complex decision-making problems. His academic journey was enriched with research internships at well-known institutions such as the University of Cambridge, Brown 2 University, TU Delft, and the Oregon State University Collaborative Robotics Institute. He won IBM's Human multi-agent immersive negotiation competition at IJCAI 2020 and achieved second place in two categories at the Supply Chain Management League at IJCAI 2021. His research effort was recognized with the Richard E. Merwin Student Scholarship by the IEEE Computer Society and the Best Research Assistant Award by Özyeğin University in 2022. Keskin was one of the main organizers of the International Automated Negotiating Agents Competition 2023 (ANAC'23) at AAMAS.

Berk Buzcu earned his Bachelor of Science in Computer Science from Özyeğin University. Subsequently, he pursued further academic education as a Master's Student at the same institution. During this time, he actively participated in the CHIST-ERA (locally Tübitak) funded project, "EXPECTATION: Personalized Explainable AI for decentralized agents with heterogeneous knowledge," His focus within the scope of the project was the application of explainability techniques in recommender systems which aligned with his research interests. The main goal was to enhance the success of recommender systems in achieving user goals via sociability. In addition to his thesis research, Berk developed a human-robot negotiation environment while touching on other multiagent-related topics involving Blockchain and drone cooperation.

Reyhan Aydoğan is an assistant professor in Computer Science at Özyeğin University and a research fellow in the Interactive Intelligence Group at Delft University of Technology. She received her M.Sc. and Ph.D. degrees in Computer Engineering from Boğaziçi University. Her primary research focuses on modeling, developing, and analyzing agent technology that integrates different aspects of intelligence, such as reasoning, decision-making, and learning. She applies AI techniques such as machine learning and semantic reasoning in designing and developing agent-based decision support systems, e.g., negotiation support systems and automated negotiation tools. She is particularly interested in supporting human decision-makers in complex and dynamic environments, which also requires the design of effective human-computer interaction (e.g., preference elicitation). She serves as a program committee member in conferences such as AAMAS, IJCAI, and ECAI and is a reviewer for journals such as JAAMAS and JAIR. Dr. Aydoğan is one of the main organizers of the International Automated Negotiating Agents Competition (ANAC).