



# Empowering reciprocal recommender system using contextual bandits and argumentation based explanations

Tulika Kumari<sup>1,2</sup> · Bhavna Gupta<sup>3</sup> · Ravish Sharma<sup>4</sup> · Punam Bedi<sup>1</sup>

Received: 19 November 2022 / Revised: 15 January 2023 / Accepted: 30 March 2023 /  
Published online: 29 May 2023

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

## Abstract

Reciprocal Recommender Systems (RRS) aim to recommend relevant matches to users based on the mutual agreement of their preferences. Explainability of reciprocal recommendations is important for developing a persuasive reciprocal recommender system, since it can improve the effectiveness and credibility of the reciprocal recommendation results. Explainable RRS provide an explanation highlighting why a recommendation would be relevant to the user. Explaining the rationale behind predictions with textual or visual artifacts help in increasing trustworthiness and transparency of the system which is crucial especially for models that are used in critical decision making. In this work, XSiameseBiGRU-UCB, a deep learning contextual bandits framework with post-hoc argumentation based explanations for RRS is proposed. XSiameseBiGRU-UCB is an explainable Siamese neural network-based framework that provides explanations to justify the generated reciprocal recommendations for both the parties involved. In RRS, dilemma between exploitation and exploration requires identifying the best possible recommendation from known information or collecting more information about the environment while generating reciprocal recommendations. To tackle this, we propose to use a contextual bandit policy with upper confidence bound, which adaptively exploits and explores user interests to achieve increased rewards in the long run. Experimental studies conducted with four real-world datasets validate the efficacy of the proposed approach.

---

✉ Bhavna Gupta  
bgupta@keshav.du.ac.in

Tulika Kumari  
tulika.cs.du.2013@gmail.com

Ravish Sharma  
ravish.sharma@pgdav.du.ac.in

Punam Bedi  
pbedi@cs.du.ac.in

<sup>1</sup> Department of Computer Science, University of Delhi, 110007 Delhi, India

<sup>2</sup> Shaheed Rajguru College of Applied Sciences for Women, University of Delhi, 110096 Delhi, India

<sup>3</sup> Keshav Mahavidyalaya, University of Delhi, 110034 Delhi, India

<sup>4</sup> P.G.D.A.V. College, University of Delhi, 110065 Delhi, India

**Keywords** Reciprocal recommender systems · Contextual bandits · Siamese neural network · Explanations · Argumentation · Semantic similarity

## 1 Introduction

Reciprocal Recommender Systems (RRS) that suggest users to each other can be found in many modern applications, from online dating and recruitment, to mentor-mentee matching [1–3]. RRS provide reciprocal recommendations based on the agreement between users' preferences. Reciprocal recommendation is one where likings of user being recommended and the user who receives recommendation are fulfilled. For example, in online dating, a good match can happen when both the parties agree upon each other with respect to their preferences such as attractiveness, hobbies, profession. Likewise, in the case of online recruitment, both the candidate as well as the recruiter has some preferences such as salary, expertise, skills required for the position advertised and possessed by the candidate that has to be satisfied.

Recent advancements in deep learning techniques have shown outstanding predictive performance and have widely been used in reciprocal recommender systems. However, deep learning is often subject to flak because of its 'black-box' nature, which refers to its inability to 'explain' its results in a form that is suitable for human understanding. Even if we are able to recognize the basic mathematical modelling of these deep learning architectures, it's not possible to get insight of the actual internal working of these models. We often need explicit reasoning tools that are able to explain the results of the model.

Generating explainable recommendations has become critical due to the demand of transparency in artificial intelligence applications, accountability and ethical practices. In recent years, it has become crucial to offer explanations to justify or explain how recommendations are generated by the algorithm. This enables users to decide judiciously whether or not they should accept the recommendations and also to what degree. Generating explainable recommendations is a necessary requirement to increase user's credence in the recommendations generated by the system. An explanation can be any information that dictates the fundamental reasons for the decision taken by the model.

Transparency in design of the model (also known as intrinsic) and post-hoc explanations are two common ways of generating explainable recommendations [4]. Intrinsic explanation refers to identifying how a model functions by understanding its structure, components and training algorithms. The post-hoc explanation focuses on explaining recommendation results i.e., why a particular outcome is inferred. It facilitates analytic statements, e.g., why an item is recommended to you; provides visualizations and explanations by example to support recommendation results. Figure 1 shows a general categorization of explainable artificial intelligence (XAI) approaches.

Many situations in real life involve choices that are difficult to make, such as choosing the optimal course of action based on available information (exploitation) or learning more about the environment (exploration). This is referred to as the exploitation vs. exploration trade-off, which necessitates balancing reward maximization based on the knowledge that the agent has previously obtained and trying new actions to further contribute knowledge. When generating recommendations, RRS is faced with a choice between exploiting the user's known preferences and exploring potential unexplored preferences. In essence, RRS must leverage its understanding of the previously selected people as potential matches for any user, while also looking into new options for the user, while making suggestions. Figure 2 shows a toy illustration of the exploitation-exploration people engage in when choosing a

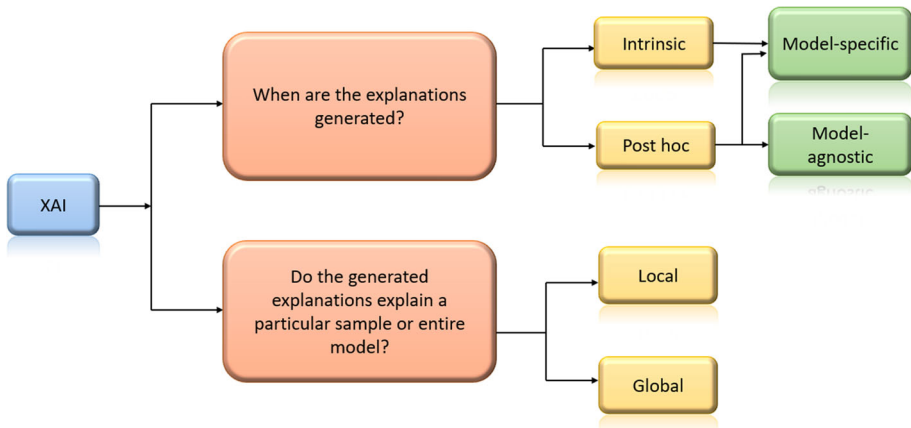


Figure 1 Taxonomy of XAI

compatible partner in online dating. The figure depicts that Alice is exploiting her knowledge and Mary is exploring more options to choose her partner. It highlights the fact that people may oscillate between these two approaches as they evaluate different potential partners and weigh the pros and cons of each option. Ultimately, the goal is to find the best possible match, which may require both exploiting existing knowledge and exploring new possibilities. The general RRS are not capable of solving the exploitation-exploration dilemma which limits their ability to present new or informative choices for the user.

With an aim to tackle this dilemma, we have formulated reciprocal recommender system as a contextual bandit [5]. In contextual bandit setting, a learner or agent perceives context, selects action, and obtains numerical loss or reward from the environment [6].

We propose XSiameseBiGRU-UCB: Siamese neural network model with argumentative explanations. Siamese architecture used in XSiameseBiGRU-UCB transforms raw

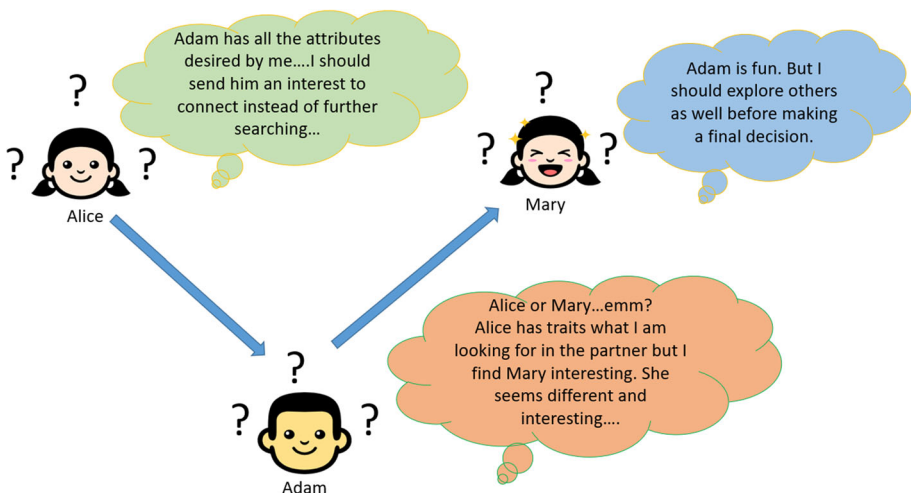


Figure 2 Toy example to illustrate exploitation-exploration in RRS

features and adds upper confidence bound (UCB) in the last layer of the network to address the exploitation vs exploration dilemma in reciprocal recommender systems. XSiameseBiGRU-UCB provides reciprocal recommendations along with the personalized and intuitive explanations. It uses argumentation to generate post-hoc explanations of the reciprocal matches generated. To generate arguments for supporting a reciprocal match, our proposed approach uses contextual information available in user profiles. We compute arguments' degree of support for the claim using Sorensen-Dice coefficient (SDC) and Siamese Bi-directional Gated Recurrent Units network semantic model based on Sentence-BERT (SBiGRU-SBERT). We chose SDC to compute similarity between binary vectors as it excludes negative co-occurrences and, in our case, the positive matches are more significant than the negative matches in the binary vectors. For computing the proximity between textual data from open ended descriptions in user profiles, we proposed SBiGRU-SBERT which is inspired by our SBiGRU model [7] with Sentence-BERT (SBERT).

Our proposed XSiameseBiGRU-UCB captures the predictive strength of deep learning along with associative reinforcement learning to find reciprocal recommendations. XSiameseBiGRU-UCB requires no other information apart from users' profile for generating recommendations. Thus, the system is free from the cold-start problem. Experiments to evaluate XSiameseBiGRU-UCB are performed on four datasets viz. speed experiment data<sup>1</sup>, anonymized dataset from an online dating site<sup>2</sup>, okCupid profiles<sup>3</sup> and online recruitment data<sup>4</sup>. Our proposed approach exhibits better performance in comparison to state-of-the-art algorithms.

The main contributions of XSiameseBiGRU-UCB are summarized below:

- XSiameseBiGRU-UCB provides argumentative explanations for the generated reciprocal recommendations using proposed Siamese Bi-directional Gated Recurrent Units network semantic model based on Sentence-BERT (SBiGRU-SBERT) and Sorensen-Dice coefficient (SDC).
- XSiameseBiGRU-UCB utilizes multiple aspects such as demographical information, primary goal, user's preferences, UCB exploratory strategy to recommend new and informative choices to the user.
- We provide a formalization using contextual bandits for the inherent exploitation-exploration dilemma of reciprocal recommender system.
- XSiameseBiGRU-UCB also addresses critical challenges faced by RRS namely cold-start problem, limited availability of users and popularity bias.
- An integrated loss function that combines binary cross entropy and contrastive loss is used in the training stage to encourage inter-class separability and intra-class compactness.
- Extensive experiments conducted on real-world datasets and comparisons with state-of-the-art approaches exhibit the efficacy of XSiameseBiGRU-UCB w.r.t. number of evaluation criteria.

Section 2 discusses related work followed by basic concepts in reciprocal recommender system, argumentation and contextual-bandits in Section 3. Section 4 describes our proposed approach XSiameseBiGRU-UCB. Datasets used, implementation and experimental results are discussed in Section 5. Lastly, we conclude in section 6 followed by references.

<sup>1</sup> <https://www.kaggle.com/datasets/annavictoria/speed-dating-experiment>

<sup>2</sup> [https://figshare.com/articles/dataset/Gender-specific\\_preference\\_in\\_online\\_dating/6429443/1](https://figshare.com/articles/dataset/Gender-specific_preference_in_online_dating/6429443/1)

<sup>3</sup> <https://www.kaggle.com/datasets/andrewmvd/okcupid-profiles>

<sup>4</sup> <https://insights.stackoverflow.com/survey>, <https://www.kaggle.com/samrat77/indeed-software-engineer-job-dataset>

## 2 Related work

The study of reciprocal recommender systems has grown rapidly due to the increasing interest in online dating, online recruitment and other social matching domains. Moreover, explainable artificial intelligence (XAI) has emerged as a field which aims at making black-box models understandable to augment their effectiveness. We provide a review of reciprocal recommender systems and explainable recommendations.

RECON [1] is one of the best-known preliminary studies for generating reciprocal recommendations in online dating. It computes unidirectional preference scores of users with other users and combines them into a reciprocity score using harmonic mean. Krzywicki et al. [8] present a recommendation algorithm in two-stages with collaborative filtering and re-ranking the recommendations using decision tree constructed with interactions data as a “critic”. Zheng et al. [9] present a multi-dimensional utility framework using multi-criteria ratings for speed-dating data. Zheng et al. [10] further present utility-based multi-stakeholder recommendation approach and techniques to learn user expectations if not available in the data. Tay et al. [11] estimate the agreement between users for relationship recommendation on regular social networks using deep learning. Neve et al. [12] present collaborative filtering-based reciprocal recommendations using latent factor models and further propose hybrid RRS for a social networking service [13]. Kumari et al. [5] propose a multifaceted reciprocal recommendation approach using contextual-bandits for online dating. Different from their previous work, in this paper, authors propose an explainable RRS which generates argumentative explanations using SBiGRU-SBERT and SDC. This work also improves the model SiameseNN-UCB proposed in [5] with bi-directional gated recurrent units and integrated loss function.

Some of the existing works in the literature focuses on reciprocal recommendations for domains such as online recruitment [2], online learning [14], social network sites [15], skill sharing [13], mentor-mentee [3].

Equality and diversity are becoming increasingly significant in online dating or matrimonial. This motivates us to highlight these considerations within RRS. A wide variety of research within RRS focusses on online dating for opposite gender relationships (heterosexual dating), but not extended to homosexual dating. Our proposed approach can be applied on single class (e.g., homosexual online dating) as well as two-disjoint class (e.g., heterosexual online dating). We also distinguish reciprocity as direct and indirect.

Most of the existing works in explainable recommender system focus mainly on the non-reciprocal recommendations. These explanation approaches consider only the preferences of the user for whom recommendations are being generated [16]. In [17], a matrix factorization model is trained to generate association rules that interpret the recommendations. Existing work primarily employs global association rule mining to identify relationships between items, which is based on item co-occurrence across all user transactions. As a result, the explanations are not personalized, meaning that different users will receive the same explanation as long as they are recommended the same item and have similar purchase histories. This limits its applicability in modern recommender systems or reciprocal recommender systems, which strive to provide personalized services to users. Additionally, it's not possible to associate all recommended items with other items based on available historical interactions, which means that in some cases no explanation can be provided which limits the usefulness of the proposed method. Zhang et al. [18] introduce a news recommendation model that utilizes meta explanation triplets to provide user-centered and news-centered explanations. The model includes heterogeneous graphs to use various types of side information to

improve news recommendation. Explanations are categorized into two sets: one set centered around the user (e.g., “users living in the same city”) and another set centered around the product (e.g., “news with the same topic”). In this paper, however, we present personalized and intuitive explanations for both parties along with a degree of support which tells degree by which arguments justify the reciprocal recommendation generated using the proposed model, XSiameseBiGRU-UCB.

Explainable reciprocal recommendations have shown its effectiveness in maximizing the persuasiveness of the generated recommendations [19].

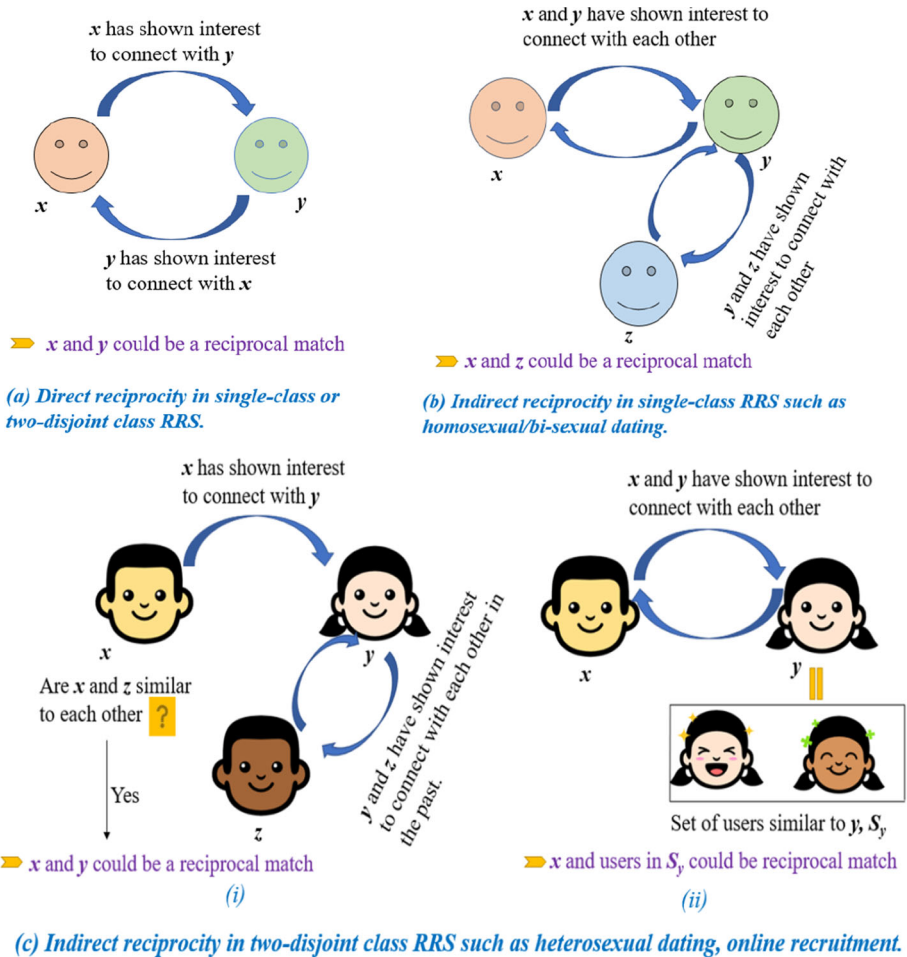
Over the years, argumentation has achieved substantial success in explainable artificial intelligence due to its strong explainability capabilities [20]. The explanations generated by argumentation are closer to the way humans reason or think. Some of the existing works in recommender systems use [21, 22] Defeasible Logic Programming [23]. Bedi et al. [24] propose an interest-based recommender system for generating personalised recommendations. Briguez et al. [22] details the conditions for movie recommendation using a set of predefined postulates. Naveed et al. [25] develop a formalisation of explanations for generated recommendations based on Toulmin’s model of argumentation [26]. Briguez et al. [27] generate argumentation-based recommendations after modelling rule-based user preferences using Defeasible logic programming. Argumentation is integrated in social recommender system [28] to provide reasoning of the generated recommendations using the preferences of neighbours. Existing works in the literature of recommender systems discussed above uses argumentation. However, to the best of our understanding, argumentation is not yet explored in reciprocal recommender systems. Different from the earlier works, in this paper, we use factual and evidence arguments based on contextual information such as demographic information, primary goal, user’s preferences, exploratory behaviour to provide natural language explanations which are customized and comprehensible. In this paper, we propose argument-based reasoning in XSiameseBiGRU-UCB to provide explanations using arguments in support of generated recommendation.

### 3 Preliminaries

#### 3.1 Reciprocal recommender systems

Reciprocity often plays a primal role in person-to-person relationships. It aids in forming social connection and mutual benefit. The ability to build a good match is usually dependent on both parties reciprocating. Although, reciprocity is a well-examined concept in various domains, namely, evolutionary psychology and economics to view the impact of cooperation in human populations. Our work considers reciprocity in RRS and distinguishes two kinds of reciprocity: direct and indirect reciprocity, both of which can be thought of as key mechanisms for mutual agreement. Direct reciprocity can be observed when both the parties involved use their own experiences/preferences to decide whether a request should be sent to connect with the other person or not. However, in indirect reciprocity they also consider the experiences of others. Figure 3 illustrates different kinds of direct or indirect reciprocity that can be observed in person-to-person reciprocal recommender system.

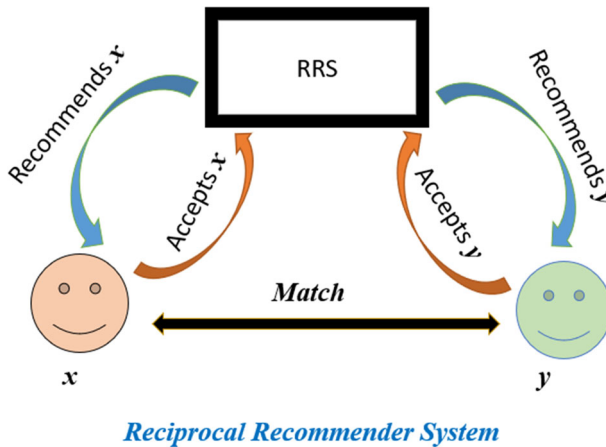
**Reciprocal Recommender Systems (RRS)** generate recommendations based on the mutual agreement of users’ preferences and are principally used in domains such as, online dating, online recruitment, academic or research collaborations. Figure 4 shows a general RRS.



**Figure 3** Illustration of different forms of reciprocity which can be observed in RRS

RRS intrinsically differs from conventional item-to-user Recommender Systems (RS) in following aspects:

- With a conventional RS, the emphasis is on recommending items that the user will find most interesting. A good recommendation for the RRS problem, however, should take into account the interests of both parties, and not just the user receiving the recommendation.
- In conventional RS, recommending the same item to a large number of users is not an issue. This contrasts with RRS as availability of persons is constrained. It is best to refrain from suggesting the same individual to many people. For example, in online dating, people select only few people or possibly one person to date. Also, the person who is recommended to too many people, becomes overawed by the attention and stops replying. Thus, overloading people should be evaded in RRS.
- Sometimes, users may quit the RRS site and never come back even after getting successful match. Because of this, the cold-start issue is critical in RRS. However, even after



**Figure 4** Reciprocal Recommender System

a successful recommendation is made, users may still remain a part of the system in conventional RS.

Depending upon the decomposability of the set of users, RRS can be further classified as follows:

**Single class RRS** If any user  $y$  is a potential match to any other user  $x$  where  $x, y \in U$  and  $x \neq y$ , then it is single-class RRS.

**Two-disjoint class RRS** Set of all users represented by  $U$  can be partitioned into two disjoint sets  $U_1$  and  $U_2$  such that if  $x \in U_1$  then  $y \in U_2$  and vice versa.

### 3.2 Basics of argumentation

Argumentation attempts to identify suitable conjecture and conclusions of reasoning. It is an important and commonly used form of human cognition. We often analyse some key arguments and counterarguments while making a decision.

An argument attempts to support or defend beliefs, decisions or claims. It tries to give a logical proof of the claim. In simpler terms, argumentation can be thought of as a justification of a knowledge through another knowledge. Knowledge can be expressed as judgments, decisions, hypotheses, concepts etc. Argument involves supporting, attacking and/or modifying claims so that decision makers may accept or deny the claim. It also helps in explaining the results of reasoning usually by identifying related information and generating an explanation. Claim refers to the knowledge that is being justified and knowledge that is used in justification of claim are called arguments, or reasons. For example, consider the following newspaper story (Associated Press 1993): "A recent study found that women are more likely than men to be murdered at work. 40% of the women who died on the job in 1993 were murdered. 15% of the men who died on the job during the same period were murdered." In this example, first sentence is a claim, and next two sentences provide reasons to justify this claim [29].

In the literature, argumentation is distinguished as monological and dialogical argumentation [30]. Monological argumentation involves justifying a claim from a knowledge base



containing defeasible premises, and dialogical argumentation refers to an interactive process where a set of entities or agents interact with each other to identify arguments in support of or against a particular claim. Newspaper article by an editor, speech by a politician in public rallies or talk by a resource person in a seminar are some of the situations where one can observe monological argumentation. Plaintiff making a claim and defendant resisting it in a civil lawsuit, opposition and ruling party politicians debating about a new legislation are some of the examples where dialogical argumentation occurs.

Argumentation can inherently be based on any kind of information which can be objective, subjective, or hypothetical [31]. Objective information refers to the information that is data-based, observable, or measurable by everyone involved in the argumentation. Subjective information is based on beliefs or opinions from some of those involved in the argumentation. Hypothetical information is assumed by hypothesis for the sake of constructing arguments. For example, suppose you are asked to give review of a restaurant food. There may be certain foods that you subjectively dislike i.e., foods that you personally don't like. But while critiquing food, you leave your subjective tastes aside and make decision on the basis of objective information such as how it is cooked and seasoned. Monological argumentation is used in various argument-based logics for defeasible reasoning. In this paper, we focus on monological argumentation based on objective information for generating explanations.

### 3.3 Contextual bandits

Bandit algorithms [32] are a class of learning algorithms that function well in uncertain environments. The name “bandit” comes from the idea of a slot machine, with multiple options (arms) and rewards for each. By balancing exploration and exploitation, the algorithm can learn the underlying reward distributions for each arm and thus achieve the highest cumulative reward over time. The goal is to achieve the best results by taking into account the trade-off between exploiting what you know and exploring for more knowledge about the possible options. At its core, a multi-armed bandit problem is a series of repeated trials in which the user is presented with a fixed number of options (known as “arms”) and receives a reward based on the choice they make. The goal is to balance the exploration of new options with the exploitation of options that have been found to be profitable in order to maximize the overall rewards over time. Contextual bandit is a particularly valuable variation of the multi-armed bandit problem, where the agent is presented with an N-dimensional context, or feature vector, before making a decision on which arm to pull. The objective is to gradually understand the association between the context vectors and rewards in order to make more accurate predictions on which action to take, given the context.

Contextual bandits are a good-fit for wide range of situations where exploitation and exploration trade-off occurs such as clinical trials, web search, and recommender systems. In such settings, an agent has to iteratively make choices that can maximize its expected rewards. However, the agent often lacks the required knowledge of the reward generation process. Thus, the agent has to explore in order to gain more insights of the reward generating process. This aforementioned exploration vs. exploitation dilemma is encountered in many sequential decision problems and can be modelled as contextual bandits [33].

Contextual bandit problems often use optimistic algorithms, which make a deterministic choice at each round based on an optimistic estimate of future rewards. Upper confidence bound (UCB) [34] based on the “optimism in the face of uncertainty” principle, is an effective and well-known method to solve contextual bandit problems. These algorithms are popular due to their simplicity and effectiveness, and they are often preferred because of their ability

to discard sub-optimal actions and achieve better results. UCB-type algorithms are also easier to analyse, which makes them a good choice for complex settings. The UCB strategy balances the exploration and exploitation trade-off by setting an upper bound on the reward for each action, made up of two components. The first component is an estimate of the reward, and the second component reflects the level of confidence in that estimate. The strategy is to choose the action with the highest UCB. When the agent has a low confidence in its reward estimates, the second term becomes more prominent, promoting exploration. Conversely, when all the confidence terms are small, the algorithm focuses on exploiting the best action(s) available. It selects arms (actions) using an upper confidence bound value which is estimated with mean reward of an arm ( $r_{t,a}$ ) and confidence interval value ( $c_{t,a}$ ). LinUCB [6], a well-known contextual bandits algorithm, utilizes a linear model to predict the expected reward for each arm based on the context.  $r_{t,a}$  is expressed with the  $d$ -dimensional feature  $x_{t,a}$  of  $a$  and an unknown coefficient  $\theta$ ,  $x_{t,a}^T \theta_t$  and confidence interval value is given by  $\alpha_t \|x_{t,a}\|_{H_t^{-1}}$  with  $\alpha_t > 0$  (tuning parameter to control the exploration rate). LinUCB uses ridge regression to estimate the unknown coefficients based on previous trials. Its arm-selection strategy is expressed by the equation:

$$a_t = \operatorname{argmax}_{a \in A} \{x_{t,a}^T \theta_t + \alpha_t \|x_{t,a}\|_{H_t^{-1}}\} \tag{1}$$

where  $H_t$  matrix is computed as

$$H_t = \lambda I_d + \sum_{i=1}^t x_{i,a_i} \cdot x_{i,a_i}^T \tag{2}$$

## 4 Proposed work

### 4.1 Problem specifications

#### 4.1.1 Contextual-bandits framework for reciprocal recommendations

We propose contextual bandits formulation for RRS as follows:

*Agent:* Reciprocal Recommender System algorithm.

*Environment:* Set of users,  $U$ .  $U$  can be divided into two disjoint sets  $U_1$  and  $U_2$  in two-disjoint class RRS such as male and female users in heterosexual online dating or recruiters and job seekers in online recruitment. For single-class RRS such as bisexual or homosexual online dating, any user  $a$  can be recommended to  $u$  where  $a, u \in U$  and  $a \neq u$ .

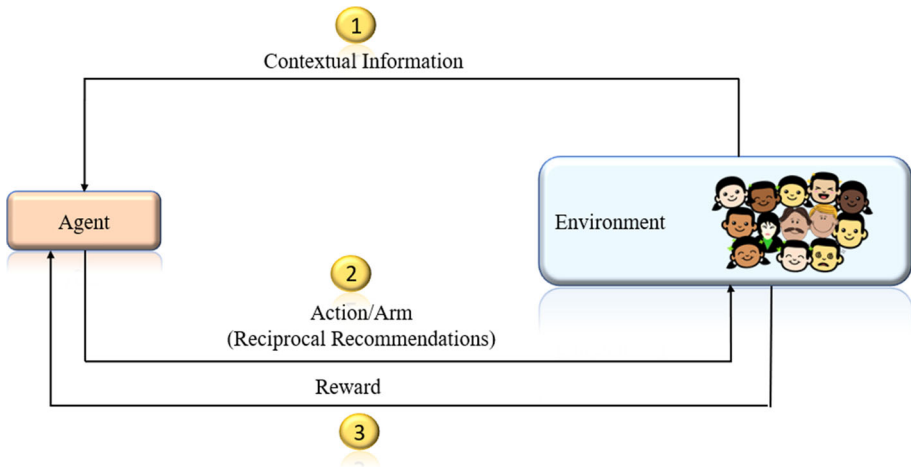
$$U = \left\{ \begin{array}{ll} U_1 \cup U_2 & \text{two-disjoint class RRS} \\ U & \text{single-class RRS} \end{array} \right\} \tag{3}$$

*Arm or action:* Reciprocal match  $a \in A$  of user  $u$ .

$$\text{Set of arms, } A = \left\{ \begin{array}{ll} U_2 & \text{two-disjoint class RRS and } u \in U_1 \\ U_1 & \text{two-disjoint class RRS and } u \in U_2 \\ U \setminus \{u\} & \text{single-class RRS} \end{array} \right\} \tag{4}$$

*Context:* Contextual information of user  $u$  and arm  $a$ .

*Reward:* Reward of an arm  $a$ ,  $r_{t,a} \in [0, 1]$  is 1 if  $u$  and  $a$  are reciprocal match to each other and 0 otherwise.



**Figure 5** Agent-environment for contextual bandits setting

We formulate RRS using the contextual bandit framework according to the specifications given below.

For each round  $t \in [T]$ , the agent

1. Observes the context  $s_t \in S$  where  $S$  denotes the context space.
2. Selects an arm  $a \in A$  on the basis of observed rewards in prior rounds and receives a reward  $r_{t,a}$  which depends on user as well as the arm.
3. Improves arm-selection strategy using  $s_t, a$  and  $r_{t,a}$ .

In each round  $t$ , agent observes  $u_t$  and  $a$  with their  $d$ -dimensional feature vectors,  $\langle x_{u_t} \rangle$  and  $\langle x_a \rangle$  respectively. An arm  $a$  is recommended to the user  $u_t$  and a reward  $r_{t,a}$  is received by the agent. Agent selects arms so as to maximize the cumulative reward or minimize the regret given by:

$$R_T = E[\sum_{t=1}^T (r_{t,a}^* - r_{t,a})] \tag{5}$$

where  $r_{t,a}^*$  denotes the true reward and  $r_{t,a}$  denotes the predicted reward at round  $t$ .

At any round  $t$ , reward,  $r_{t,a}$  for an arm  $a$  given feature vectors  $\langle x_{t,u_t} \rangle$  and  $\langle x_{t,a} \rangle$  can be expressed by  $f(\cdot; \langle x_{u_t} \rangle, \langle x_a \rangle, W)$ . Here,  $W$  denotes an unknown weight matrix.

$$r_{t,a} = f(\cdot; \langle x_{u_t} \rangle, \langle x_a \rangle, W) \tag{6}$$

Figure 5 depicts agent-environment in contextual bandits setting of RRS.

### 4.1.2 Argumentation for explanations of reciprocal recommendations

Argumentation provides reasoning to support a claim with the help of some premises. In this paper, we use argumentation to generate post-hoc explanations. The underlying idea is to build conclusion for the claim by generating series of arguments that support the claim. Support for a claim can mean that the claim may be inferred from existing arguments, or that the claim is evidential.



**Figure 6** Illustration of argument to justify a claim

For explanations of reciprocal recommendation, we generate arguments by exploiting contextual information based on various attributes. Factual and evidence arguments are given as explanations to justify the reciprocal recommendation. Factual argument is based on the explicit preferences of users and gives direct reasoning behind the recommendations. Evidence argument is based on the similarity between the user being recommended as reciprocal match and users liked in the past.

We define the notion of argumentation using: (1) claim that is being justified, (2) support by which the justification is performed (3) to what degree arguments justify the claim.

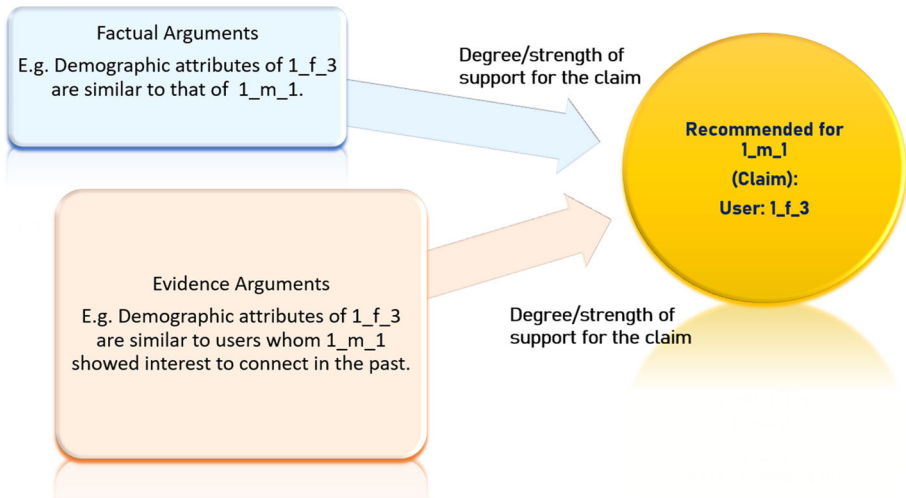
**Argumentation** Argumentation, in our usage, is defined as a unit comprising a claim, its support arguments and degree of strength for support ( $\langle \textit{claim} \rangle : \langle \textit{support} \rangle : \langle \textit{degree of support} \rangle$ ). Support arguments may be explicit or implicit. Figure 6 presents a diagrammatic representation of argumentation. Figure 7 illustrates an example of explanations using argumentation in our usage.

## 4.2 Proposed approach: XSiameseBiGRU-UCB

XSiameseBiGRU-UCB is a deep learning contextual bandits framework with post-hoc argumentation based personalized explanations for RRS. The proposed model has Siamese bi-directional Gated Recurrent Units to transform raw features and uses UCB strategy to tackle the inherent exploitation vs exploration dilemma in RRS. Siamese architecture aids in distance metric learning to drive a similarity metric. The reward function  $f(\cdot)$  is expressed in terms of feature vectors and UCB term.

For a pair of users  $u_t$  and  $a$  ( $u_t \in U$ ,  $a \in A$ ,  $a \neq u_t$ ), output of the Siamese bi-directional Gated Recurrent Units network,  $\varphi_W(x_{u_t}, x_a; W)$  is defined as follows:

$$\varphi_W(x_{u_t}, x_a; W) = (G_W(x_{u_t}) - G_W(x_a)) \quad (7)$$



**Figure 7** Illustration of argumentation-based personalized explanation

Here,  $W$  represents shared weight in the Siamese architecture which needs to be learnt.  $\varphi_W(x_{u_t}, x_a; W)$  is the difference between output from two identical bi-directional Gated Recurrent Units and is defined as:

$$G_W(x_{u_t}) = \chi(x_{u_t}) + \alpha_t \|\chi(x_{u_t}; W)\|_{H_t^{-1}} \tag{8}$$

$$G_W(x_a) = \chi(x_a) + \alpha_t \|\chi(x_a; W)\|_{H_t^{-1}} \tag{9}$$

$\chi(\cdot)$  is the  $L^{th}$  hidden layer output.

Arm-selection strategy to select the best-suited arm is then given by:

$$a_t = \operatorname{argmin}_{a \in A} \{\varphi_W(x_{u_t}, x_a; W)\} \tag{10}$$

$\|\cdot\|_H$  is Mahalanobis distance:

$$\|x\|_H = \sqrt{x^T H x} \tag{11}$$

$\alpha_t > 0$  is used to control exploration rate and  $H_t$  matrix is computed as:

$$H_t = \lambda I_d + \sum_{i=1}^t \chi(x_{i,u_i}; W_{i-1}) \cdot \chi(x_{i,a_i}; W_{i-1})^T \tag{12}$$

$\lambda > 0$  and  $I_d$  is an identity matrix of  $d$ -dimension. Algorithm 1 describes in detail the steps followed while generating reciprocal recommendations.

We integrated binary cross-entropy (18) with contrastive loss (19) to measure the prediction error while generating reciprocal recommendations. These measures are complementary to each other. Contrastive loss mainly focuses on minimizing intra-class variations and the cross-entropy loss guides in maximizing the inter-class variations. We wish to learn a function  $f(\cdot)$  that maps  $u_t$  and  $a$  into embedding vectors such that similar samples have similar embeddings and different samples have different embeddings. For a pair of inputs, contrastive loss tries to minimize the embedding distance when they belong to the same class but maximizes the distance otherwise. Contrastive loss thus reduces the intra-class variations by bringing

feature vectors from the same instance together. Our loss function (17) thus increases the inter-class separability in addition to reducing intra-class separability.

---

**Algorithm 1** Algorithm for Reciprocal Recommendations.

---

**Require** Number of rounds  $T$ , a deep Siamese bi-directional Gated Recurrent Units network that defines  $f(\cdot; x_t, u_t, x_a, W)$  and set of users  $U$ . **Output** Reciprocal recommendations,  $RR$ .

```

1: Partition  $U$  in the case of two-disjoint class as  $U_1$  and  $U_2$ , otherwise  $U$  only i.e.,
   
$$U = \begin{cases} U_1 \cup U_2 & \text{two-disjoint class RRS} \\ U & \text{single-class RRS} \end{cases}$$

2: for  $t = 1, 2, 3, \dots, T$  do
3:   Extract  $u_t$ 's profile  $P_{u_t}$ 
4:   if two-disjoint class RRS then
5:     if  $u_t \in U_1$  then
6:       Extract  $a$ 's profile  $P_a \forall a \in A$  where  $A = U_2$ 
7:     end if
8:     if  $u_t \in U_2$  then
9:       Extract  $a$ 's profile  $P_a \forall a \in A$  where  $A = U_1$ 
10:    end if
11:  end if
12:  if single-class RRS then
13:    Extract  $a$ 's profile  $P_a \forall a \in A$  where  $A = U \setminus \{u_t\}$ 
14:  end if
15:  Feature vector representations  $x_{u_t}$  and  $x_a$  of  $u_t$  and  $a$  respectively are fed to Siamese bi-directional Gated Recurrent Units network.
   //Upper confidence bound value is added in the last layer
16:   $G_w(x_{u_t}) = \chi(x_{u_t}) + \alpha_t \|\chi(x_{u_t}; W)\|_{H_{t-1}^{-1}}$ 
17:   $G_w(x_a) = \chi(x_a) + \alpha_t \|\chi(x_a; W)\|_{H_{t-1}^{-1}}$ 
18:  Reward of an arm,  $\varphi_W(x_{u_t}, x_a; W) = (G_w(x_{u_t}) - G_w(x_a))$ 
19:   $H_t = H_{t-1} + \chi(x_t, u_t; W_{t-1}) \cdot \chi(x_t, a; W_{t-1})^T$ 
20:  Update the weight parameters.
21: end for
22: Reciprocal compatibility score  $rCS(u_t, a)$  between  $u_t$  and  $a$ ,  $rCS(u_t, a) = E[r_a]$  where  $E[r_a] = \varphi_W(x_{u_t}, x_a; W)$ 
23: Let  $Y_u$  represents list of users who are potential matches to  $u$ .
24: Popularity score for a user  $a$ ,
   
$$pop_a = \sum_{u \in U} \frac{\mathbb{1}(a \in Y_u)}{|Y_u|}$$

   where  $\mathbb{1}(a \in Y_u) = \begin{cases} 1 & \text{if } a \in Y_u \\ 0 & \text{otherwise} \end{cases}$ 
25: Popularity-aware reciprocal score,
   
$$ParS(u_t, a) = rCS(u_t, a)(1 - \mathbb{1}(pop_a - \phi))(1 - pop_a)$$

   where  $\mathbb{1}(pop_a - \phi) = \begin{cases} 1 & \text{if } (pop_a - \phi) \geq 0 \\ 0 & \text{otherwise} \end{cases}$ 
   //Popularity-aware reciprocal recommendations
26:  $RR_u \leftarrow \{v_1, v_2, \dots, v_i : v_i \in A \text{ and } ParS(u, v_i) \geq ParS(u, v_{i+1}) \forall i\}$ 
27: Return  $RR_u$ 

```

---

With the proposed approach XSiameseBiGRU-UCB, after generating a list of reciprocal recommendations for users, users are assisted with argumentation that provides intuitive and personalized explanations. We propose to utilize post-hoc explanation approach that attempts to provide the explainability of reciprocal recommendation results.

The proposed methodology for explaining the reciprocal recommendations is represented by means of two algorithms (Algorithm 2 and 3). Algorithm 2 gives a detailed description of the proposed post-hoc argumentation-based explanations. It provides explanations

as to why a user  $u_2$  is a potential match to another user  $u_1$ . To generate arguments in support of  $u_2$ , algorithm 2 computes proximity scores between multi-criteria attributes desired by  $u_1$  and multi-criteria attributes possessed by  $u_2$  ( $ArgDec_{u_1,u_2}^{(m)}$ ), demographic attributes ( $ArgDec_{u_1,u_2}^{(d)}$ ), availability (using goal and frequency of going on the event) of  $u_1$  and  $u_2$  ( $ArgDec_{u_1,u_2}^{(fg)}$ ) and hobbies ( $ArgDec_{u_1,u_2}^{(hobbies)}$ ). It also computes the proximity scores between multi-criteria, demographic and availability attributes of  $u_2$  and users to whom  $u_1$  shown interest in the past ( $ArgDec_{u_1,u_2}^{(mDated)}$ ,  $ArgDec_{u_1,u_2}^{(dDated)}$ ,  $ArgDec_{u_1,u_2}^{(fgDated)}$ ). On the basis of these computed proximity scores, it generates arguments in support for  $u_2$ . Proximity measures used in algorithm 2 for computing the degree of strength for support:

**Algorithm 2** ‘eXA’: Algorithm for generating argumentation in support of a potential match recommended by algorithm 1.

```

Require Users  $u_1, u_2$ .
Output Explanation to justify the claim of  $u_2$  being a potential match for  $u_1$ .
1: Let  $P_{u_1}$  and  $P_{u_2}$  denote profile of  $u_1$  and  $u_2$  respectively.
2: for each multi-criteria attribute  $m \in F_m$  list of multi-criteria attributes do
3:    $x_{u_1}^{(mp)} = \bigcup_{m \in F_m} < v_{u_1}^{(mp)} >$ 
4:    $x_{u_2}^{(ms)} = \bigcup_{m \in F_m} < v_{u_2}^{(ms)} >$ 
   where  $v_{u_1}^{(mp)}$  denotes  $u_1$ 's multi-criteria preference rating  $m$  and  $v_{u_2}^{(ms)}$  denotes multi-criteria attributes in  $u_2$ .
5: end for
6: for each demographic attribute  $d \in F_d$  list of demographic attributes do
7:    $x_{u_1}^{(d)} = \bigcup_{d \in F_d} < v_{u_1}^{(d)} >$ 
8:    $x_{u_2}^{(d)} = \bigcup_{d \in F_d} < v_{u_2}^{(d)} >$ 
   where  $v_{u_1}^{(d)}$  denotes  $u_1$ 's demographic attributes and  $v_{u_2}^{(d)}$  denotes demographic attributes in  $u_2$ .
9: end for
10:  $x_{u_1}^{(fg)} = < v_{u_1}^{(goal)}, v_{u_1}^{(freq)} >$ 
   where  $v_{u_1}^{(goal)}$  represents goal and  $v_{u_1}^{(freq)}$  represents frequency of going on such events of  $u_1$ .
11:  $x_{u_2}^{(fg)} = < v_{u_2}^{(goal)}, v_{u_2}^{(freq)} >$ 
   where  $v_{u_2}^{(goal)}$  represents goal and  $v_{u_2}^{(freq)}$  represents frequency of going on such events of  $u_2$ .
12:  $ArgDec_{u_1,u_2}^{(m)} = proximity(x_{u_1}^{(mp)}, x_{u_2}^{(ms)})$ 
13:  $ArgDec_{u_1,u_2}^{(d)} = proximity(x_{u_1}^{(d)}, x_{u_2}^{(d)})$ 
14:  $ArgDec_{u_1,u_2}^{(fg)} = proximity(x_{u_1}^{(fg)}, x_{u_2}^{(fg)})$ 
   // proximity is computed using equation 14 and SBiGRU-SBERT
15: Let  $D_{u_1}$  be the list of users to whom  $u_1$  has shown interest to connect in the past.
16:  $ArgDec_{u_1,u_2}^{(mDated)} = \frac{\sum_{\forall u_i \in D_{u_1}} proximity(x_{u_2}^{(ms)}, x_{u_i}^{(ms)})}{|D_{u_1}|}$ 
17:  $ArgDec_{u_1,u_2}^{(dDated)} = \frac{\sum_{\forall u_i \in D_{u_1}} proximity(x_{u_2}^{(d)}, x_{u_i}^{(d)})}{|D_{u_1}|}$ 
18:  $ArgDec_{u_1,u_2}^{(fgDated)} = \frac{\sum_{\forall u_i \in D_{u_1}} proximity(x_{u_2}^{(fg)}, x_{u_i}^{(fg)})}{|D_{u_1}|}$ 
19:  $p_{u_1,attr} = \{(v, n) : \forall \text{ unique discrete values } v \text{ of } attr$ 
   //where  $n$  is the count of occurrence of  $v$  in  $D_{u_1}$ .
20: for each attribute  $attr \in F$  list of attributes do
21:    $w_{attr} = \frac{n}{\sum_{n \in p_{u_1,attr}}$ 
22: end for
23:  $ArgDec_{u_1,u_2}^{(hobbies)} = corr(H_{u_1}, H_{u_2})$ 
   where  $H_{u_1}$  and  $H_{u_2}$  represent hobbies of  $u_1$  and  $u_2$ .
    
```

```

24: if  $ArgDec_{u_1, u_2}^{(i)} < \delta \forall i \in \{m, d, fg, mDated, fgDated, dDated\}$  then
25:   You may explore  $u_2$  to experience new and informative choices.
26: end if
27: Arguments in support for  $u_2$  to be a potential match are as follows:
28: (i) <Claim: $u_2$ , Support:  $x_{u_1}^{(mp)}$  and  $x_{u_2}^{(ms)}$ , Degree of support:  $ArgDec_{u_1, u_2}^{(m)} >$ 
29: (ii) <Claim: $u_2$ , Support:  $x_{u_1}^{(d)}$  and  $x_{u_2}^{(d)}$ , Degree of support:  $ArgDec_{u_1, u_2}^{(d)} >$ 
30: (iii) <Claim: $u_2$ , Support:  $x_{u_1}^{(fg)}$  and  $x_{u_2}^{(fg)}$ , Degree of support:  $ArgDec_{u_1, u_2}^{(fg)} >$ 
31: (iv) <Claim: $u_2$ , Support:  $x_{u_2}$  has multi-criteria attributes similar to users whom you showed interest in the
    past, Degree of support:  $w_{attr}^{(m)} * ArgDec_{u_1, u_2}^{(mDated)} >$ 
32: (v) <Claim: $u_2$ , Support:  $x_{u_2}$  has demographic attributes similar to users whom you showed interest in the
    past, Degree of support:  $w_{attr}^{(d)} * ArgDec_{u_1, u_2}^{(dDated)} >$ 
33: (vi) <Claim: $u_2$ , Support:  $x_{u_2}$  has goal and availability similar to users whom you showed interest in the
    past, Degree of support:  $w_{attr}^{(fg)} * ArgDec_{u_1, u_2}^{(fgDated)} >$ 
34: (vii) <Claim: $u_2$ , Support: You both have similar hobbies, Degree of support:  $ArgDec_{u_1, u_2}^{(hobbies)} >$ 

```

**Algorithm 3** Algorithm for explanations of reciprocal recommendations.

```

Require A user  $u_1$ , ‘eXA’ method (algorithm 2), reciprocal recommendations generated by algorithm 1.
Output Explanation of reciprocal recommendations to  $u_1$ .
1: for each  $u_i \in RR$  list of reciprocal recommendations generated by Algorithm 1 do
2:   eXA( $u_1, u_i$ ) // Algorithm 2
3:   eXA( $u_i, u_1$ ) // Algorithm 2
4: end for

```

1. Sorensen-Dice coefficient (SDC): SDC was developed by Sorensen [35] and Dice [36]. We use SDC as proximity measure for binary vectors. Table 1 shows the 2-by-2 contingency table corresponding to two binary vectors  $x_1$  and  $x_2$  where  $f_{11}$  is the number of binary features or bits equal to 1 in both  $x_1$  and  $x_2$ ,  $f_{10}$  is the number of binary features or bits equal to 1 in  $x_1$  and 0 in  $x_2$ ,  $f_{01}$  is the number of binary features or bits equal to 1 in  $x_2$  and 0 in  $x_1$  and  $f_{00}$  is the number of binary features or bits equal to 0 in both  $x_1$  and  $x_2$ . SDC can then be expressed over binary vectors  $x_1$  and  $x_2$  as follows:

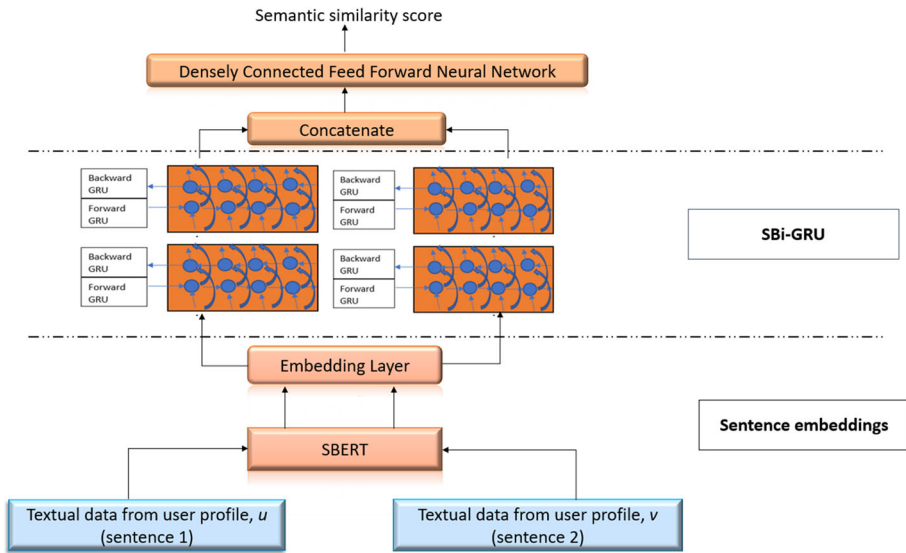
$$S_D(x_1, x_2) = \frac{2f_{11}}{2f_{11} + f_{10} + f_{01}}. \tag{13}$$

2. SBiGRU-SBERT (Siamese Bi-directional Gated Recurrent Units network semantic model based on Sentence-BERT): We use SBiGRU model [7] with Sentence-BERT (SBERT) for computing the semantic similarity between textual data from open ended descriptions in user profiles. SBiGRU-SBERT takes embedded vectors from SBERT model as its input and predicts the semantic similarity of the input text pairs. It is a siamese neural network having two identical networks with shared weights and parameters. It has

**Table 1** Contingency table for binary vectors  $x_1$  and  $x_2$  ( $1 \leq i \leq |x_1|$ )

	Attribute $i$ of $x_2$ assuming a value of 1	Attribute $i$ of $x_2$ assuming a value of 0
Attribute $i$ of $x_1$ assuming a value of 1	$f_{11}$	$f_{10}$
Attribute $i$ of $x_1$ assuming a value of 0	$f_{01}$	$f_{00}$





**Figure 8** Siamese Bi-directional Gated Recurrent Units network semantic model based on Sentence-BERT(SBiGRU-SBERT)

an embedding layer, a 2-layer bi-directional GRU, a concatenate layer and densely connected feed forward neural network. Representations of each data point obtained from bi-directional GRU are concatenated and then passed through a series of two hidden layers. At last, semantic similarity score between the input data points is calculated with sigmoid activation function. Figure 8 illustrates the framework for Siamese Bi-directional Gated Recurrent Units network semantic model based on Sentence-BERT.

The goal of Algorithm 3 is to generate explanations for reciprocal recommendations to explain why both the parties would be benefitted by the match. For each user  $u_i$  in the list of reciprocal recommendations of  $u_1$  generated by algorithm 1, it uses Algorithm 2 to generate arguments in support to justify the claim  $u_i$  ( $eXA(u_1, u_i)$ ) being a potential match for  $u_1$  and vice-versa ( $eXA(u_i, u_1)$ ).

Figure 9 shows workflow of XSiameseBiGRU-UCB. Figures 10, 11, 12 and 13 show snapshots of explanations generated by XSiameseBiGRU-UCB on datasets D1 to D4.

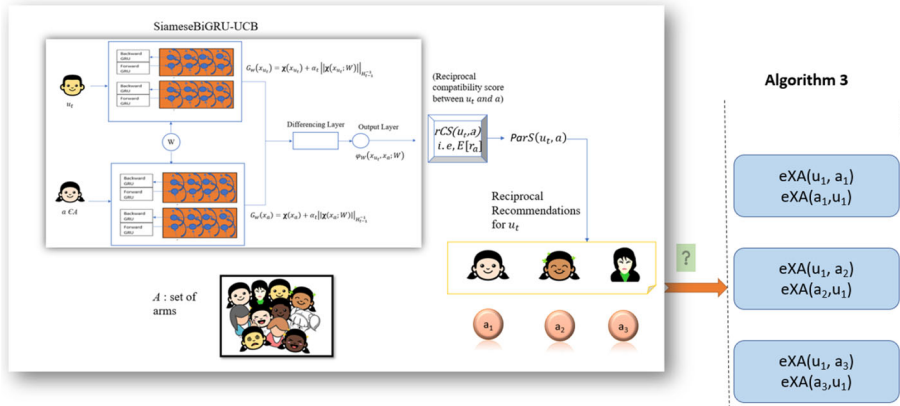
## 5 Experimental study

### 5.1 Dataset used

We conducted experiments with the following real-world datasets in order to validate XSiameseBiGRU-UCB.

- [D1] Speed-dating experiment data: The data is collated by Ray Fisman and Sheena Iyengar [37]. The dataset has 8,378 records containing demographic details of users along with the expectations and experiences of their dating behaviour.
- [D2] Anonymized dataset from heterosexual online dating site [38]. There are 548,395 total users with 344,552 male and 203,843 female users. Each user profile has 35 attributes,

Algorithm 1



Algorithm 2

### Explanations for reciprocal recommendation generated by XSiameseBi-GRU

XSiameseBi-GRU recommends user a1 to u1 as a potential reciprocal match because::

Argumentation in support for a1::

- <Claim: a1, Support: Multi-criteria attributes, Degree of support: \_\_ >
- <Claim: a1, Support: Demographic attributes, Degree of support: \_\_ >
- <Claim: a1, Support: Availability/frequency of going on dating and goal, Degree of support: \_\_ >
- <Claim: a1, Support: Multi-criteria attributes are similar to users whom you showed interest to connect in the past, Degree of support: \_\_ >
- <Claim: a1, Support: Availability/frequency of going on dating and goal are similar to users whom you showed interest to connect in the past, Degree of support: \_\_ >
- <Claim: a1, Support: Demographic attributes are similar to users whom you showed interest to connect in the past, Degree of support: \_\_ >
- <Claim: a1, Support: Similar hobbies, Degree of support: \_\_ >

And because following arguments suggest that u1 is also a good match for a1::

- <Claim: u1, Support: Multi-criteria attributes, Degree of support: \_\_ >
- <Claim: u1, Support: Demographic attributes, Degree of support: \_\_ >
- <Claim: u1, Support: Availability/frequency of going on dating and goal, Degree of support: \_\_ >
- <Claim: u1, Support: Multi-criteria attributes are similar to users whom you showed interest to connect in the past, Degree of support: \_\_ >
- <Claim: u1, Support: Availability/frequency of going on dating and goal are similar to users whom you showed interest to connect in the past, Degree of support: \_\_ >
- <Claim: u1, Support: Demographic attributes are similar to users whom you showed interest to connect in the past, Degree of support: \_\_ >
- <Claim: u1, Support: Similar hobbies, Degree of support: \_\_ >

Proximity Measures used (in Algorithm 2)

- SBiGRU-SBERT
- Sørensen–Dice coefficient

Figure 9 Workflow of XSiameseBiGRU-UCB

## Explanations for reciprocal recommendation generated by XSiameseBi-GRU

XSiameseBi-GRU recommends user 1\_f\_3 to 1\_m\_1 as a potential reciprocal match because::

```

Argumentation in support for 1_f_3::
<Claim: 1_f_3, Support: Multi-criteria attributes, Degree of support: 0.96 >
<Claim: 1_f_3, Support: Demographic attributes, Degree of support: 0.57 >
<Claim: 1_f_3, Support: Availability/frequency of going on dating and goal, Degree of support: 1.0 >
<Claim: 1_f_3, Support: Multi-criteria attributes are similar to users whom you showed interest to connect in the past, Degree of support: 0.81 >
<Claim: 1_f_3, Support: Availability/frequency of going on dating and goal are similar to users whom you showed interest to connect in the past, Degree of support: 0.28 >
<Claim: 1_f_3, Support: Demographic attributes are similar to users whom you showed interest to connect in the past, Degree of support: 0.3 >
<Claim: 1_f_3, Support: Similar hobbies, Degree of support: 0.16 >

And because following arguments suggest that 1_m_1 is also a good match for 1_f_3::
<Claim: 1_m_1, Support: Multi-criteria attributes, Degree of support: 0.98 >
<Claim: 1_m_1, Support: Demographic attributes, Degree of support: 0.57 >
<Claim: 1_m_1, Support: Availability/frequency of going on dating and goal, Degree of support: 1.0 >
<Claim: 1_m_1, Support: Multi-criteria attributes are similar to users whom you showed interest to connect in the past, Degree of support: 0.42 >
<Claim: 1_m_1, Support: Availability/frequency of going on dating and goal are similar to users whom you showed interest to connect in the past, Degree of support: 0.28 >
<Claim: 1_m_1, Support: Demographic attributes are similar to users whom you showed interest to connect in the past, Degree of support: 0.14 >
<Claim: 1_m_1, Support: Similar hobbies, Degree of support: 0.16 >

```

Figure 10 A snapshot of explanations generated by XSiameseBiGRU-UCB for speed experiment data [D1]

namely, user ID, gender, birth year, work location, education level, mate requirements etc.

[D3] Anonymized results of the 2021 Stack Overflow Developer Survey are used for candidate profiles. Candidate profiles consist of information such as current employment status, location of residence, information pertaining to education, work, career, technology, and demographic information. Job profiles are obtained from Indeed.com containing job offers related to Software Engineer of 10,000 job profiles from different companies. Each job profile contains name, company, city, ratings, summary and date.

## Explanations for reciprocal recommendation generated by XSiameseBi-GRU

XSiameseBi-GRU recommends user uid 18296 to 491530 as a potential reciprocal match because::

```

Argumentation in support for uid 18296::
<Claim: uid 18296, Support: Multi-criteria attributes, Degree of support: 1.0 >
<Claim: uid 18296, Support: Demographic attributes, Degree of support: 0.72 >
<Claim: uid 18296, Support: Multi-criteria attributes are similar to users whom you showed interest to connect in the past, Degree of support: 0.75 >
<Claim: uid 18296, Support: Demographic attributes are similar to users whom you showed interest to connect in the past, Degree of support: 0.55 >

And because following arguments suggest that uid 491530 is also a good match for uid 18296::
<Claim: uid 491530, Support: Multi-criteria attributes, Degree of support: 1.0 >
<Claim: uid 491530, Support: Demographic attributes, Degree of support: 0.72 >
<Claim: uid 491530, Support: Multi-criteria attributes are similar to users whom you showed interest to connect in the past, Degree of support: 0.51 >
<Claim: uid 491530, Support: Demographic attributes are similar to users whom you showed interest to connect in the past, Degree of support: 0.35 >

```

Figure 11 A snapshot of explanations generated by XSiameseBiGRU-UCB for online dating data [D2]

### Explanations for reciprocal recommendation generated by XSiameseBi-GRU

XSiameseBi-GRU recommends candidate uid 6 to company uid 1446 as a potential reciprocal match because::

```

Argumentation in support for candidate uid 6::

<Claim: uid 6 , Support: Main branch of the candidate and job position, Degree of support: 0.77 >
<Claim: uid 6, Support: Main branch of the candidate is similar to candidates whom you showed interest to connect in the past, Degree of support: 0.81 >
<Claim: uid 6, Support: Similarity in candidate profile and job summary:
(C;#;C++;HTML/CSS;Java;JavaScript;Node.js;PowerShell;Python;SQL;Swift;MySQL;PostgreSQL;SQLite;Express;Flask;jQuery;React.js;Keras;NumPy;Pandas;TensorFlow;Torch/PyTorch),(Fluency with Java, SQL, Perl/Python, or C++.Fast prototyping skills, familiarity with scripting languages such as bash, perl, awk, python.)),Degree of support: 0.69 >

And because following arguments suggest that company uid 1446 is also a good match for candidate uid 6::

<Claim: uid 1446 , Support: Main branch of the candidate and job position, Degree of support: 0.77 >
<Claim: uid 1446, Support: Job position offered by the company is similar to companies whom you showed interest to connect in the past, Degree of support: 0.63 >
<Claim: uid 1446, Support: Similarity in candidate profile and job summary:
(C;#;C++;HTML/CSS;Java;JavaScript;Node.js;PowerShell;Python;SQL;Swift;MySQL;PostgreSQL;SQLite;Express;Flask;jQuery;React.js;Keras;NumPy;Pandas;TensorFlow;Torch/PyTorch),(Fluency with Java, SQL, Perl/Python, or C++.Fast prototyping skills, familiarity with scripting languages such as bash, perl, awk, python.)),Degree of support: 0.69 >

```

Figure 12 A snapshot of explanations generated by XSiameseBiGRU-UCB for online recruitment [D3]

[D4] okCupid profiles: The data has 59947 rows and 31 columns containing information such as open text essays related to an individual’s preferences, personal descriptions, profession, diet, education, languages spoken etc.

Table 2 shows the specifics of the datasets used in this paper.

### Explanations for reciprocal recommendation generated by XSiameseBi-GRU

XSiameseBi-GRU recommends user uid 3577 to uid 3946 as a potential reciprocal match because::

```

Argumentation in support for uid 3577::

<Claim: uid 3577 , Support: Demographic attributes, Degree of support: 0.82 >
<Claim: uid 3577, Support: Demographic attributes are similar to users whom you showed interest to connect in the past, Degree of support: 0.76 >
<Claim: uid 3577, Support: Similarity in open ended description in your profiles:
[(i am an out going individual who has been overly engulfed with work and school latly.trying to find my way. i always assumed i would have a clear goal or quest in mind for my life. but like most great questions, any answers found just give rise to more questions.making people believe i am physick, fire dancng and throwing garbage nowhere near the rubbish can. and ya rubbish is a great word that people in america donot use enough. also i am great at taking any side of an argument, i love to play devils advocate.),(i am a outgoing girl who love to travel and see the world, i love life and trying new things. i love to laugh and to be tickled. i am slow to anger and a trustworthy friend. i am not always easy to figure out but if you take the time then i will know your worth mines.i am discovering new things about myself everyday, it's always a learning process.i am really good at being their when i am needed, so give me a call or text if you just need a ear.love friendship air nature art music i spend time thinking about my next adventure in travel.on a typical friday night i am out with friends.tell u when we meet.if your are looking to meet someone who is seeking something more.)), Degree of support: 0.72 >

And because following arguments suggest that uid 3946 is also a good match for uid 3577::

<claim: uid 3946, Support: Demographic attributes, Degree of support: 0.82 >
<Claim: uid 3946, Support: Demographic attributes are similar to users whom you showed interest to connect in the past, Degree of support: 0.61 >
<Claim: uid 3946, Support: Similarity in open ended description in your profiles:
[(i am an out going individual who has been overly engulfed with work and school latly.trying to find my way. i always assumed i would have a clear goal or quest in mind for my life. but like most great questions, any answers found just give rise to more questions.making people believe i am physick, fire dancng and throwing garbage nowhere near the rubbish can. and ya rubbish is a great word that people in america donot use enough. also i am great at taking any side of an argument, i love to play devils advocate.),(i am a outgoing girl who love to travel and see the world, i love life and trying new things. i love to laugh and to be tickled. i am slow to anger and a trustworthy friend. i am not always easy to figure out but if you take the time then i will know your worth mines.i am discovering new things about myself everyday, it's always a learning process.i am really good at being their when i am needed, so give me a call or text if you just need a ear.love friendship air nature art music i spend time thinking about my next adventure in travel.on a typical friday night i am out with friends.tell u when we meet.if your are looking to meet someone who is seeking something more.)), Degree of support: 0.72 >

```

Figure 13 A snapshot of explanations generated by XSiameseBiGRU-UCB for okCupid [D4]

**Table 2** Specifics of the datasets used

Dataset	Number of samples	Number of attributes
D1	8,378 (202 male users and 190 female users)	195
D2	548,395 (344,552 male users and 203,843 female users)	35
D3	83439 (candidate profiles) 10,000 (job profiles)	48 (candidate profiles) 6 (job profiles)
D4	59947	31

## 5.2 Dataset preprocessing

Data preprocessing is a crucial step to transform the raw data into a form suitable to be fed into a model.

### 5.2.1 Data discretization and one-hot encoding

Data discretization and one-hot encoding was performed on categorical attributes of datasets D1 to D4 described in section 5.1. Data discretization refers to the conversion of continuous values into a finite set of intervals. We discretised demographic attributes. This was followed by one-hot encoding to transform discretized attributes into one-hot encoded vectors.

One-hot encoding refers to the process of transforming nominal or categorical features into a vector by placing the value 1 or 0 in each column, depending on whether the value present in original categorical value matches the categorical column header. We used scikit-learn's OneHotEncoder for one-hot encoding. Since we discretized the categorical or nominal attributes before one-hot encoding, it reduced the problem of creation of too many new dummy variables. We selected one-hot encoding scheme because there is no quantitative relationship between the discretized attributes' values, thus, using ordinal encoding schemes which allow the model to assume an ordering between attributes' values can result in unexpected outcomes and poor performance. A common practice while using one-hot encoding is to drop one of the one-hot encoded columns from each categorical feature. However, dropping a column from one-hot encoded columns introduces a bias towards the dropped variable in regularization and also multicollinearity is rarely an issue with neural networks [39]. Thus, we have kept all the one-hot encoded columns from each categorical feature.

### 5.2.2 Textual data preprocessing and transformation

Textual data preprocessing includes converting text data to lower case, removing punctuation, line breaks, whitespaces, stop words. Short-forms and abbreviations were expanded, followed by stemming and lemmatization.

After the pre-processing phase, Sentence-BERT (SBERT) proposed by Nils Reimers and Iryna Gurevych [40] was used for sentence embedding. SBERT is a modified BERT network having siamese and triplet networks that is able to obtain sentence embeddings which are semantically meaningful. A pooling operation is added to the output of BERT to get fixed size sentence embedding using SBERT. Figure 14 depicts SBERT architecture.

We used 10-folds cross-validation method to construct train and test sets by repeatedly splitting the data into training and testing sets. The data was divided into 10 equal subsets and the model was trained on 9 of these subsets and tested on the remaining one. This process

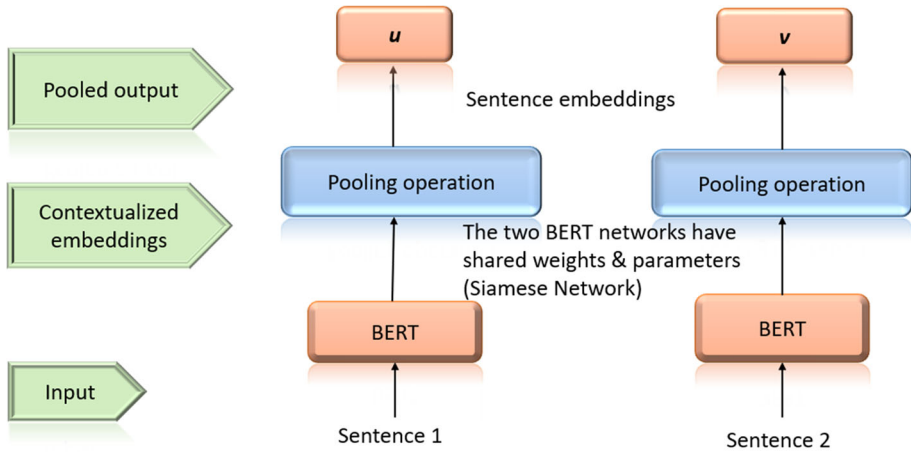


Figure 14 Architecture of SBERT

was repeated until each subset was used as the test set, and the average performance of the model on all 9 test sets was used as the cross-validated performance.

### 5.3 Experimental study

#### 5.3.1 Evaluation metrics

Precision@K, recall@K and F1-score@K are used to assess the reciprocal recommendations obtained with our proposed model.

Precision@K is calculated as the fraction of retrieved reciprocal recommended users in top-K set which are relevant. Recall@K is the fraction of relevant reciprocal matches in top-K retrieved reciprocal recommendations.

$$Precision@K = \frac{|relevant \cap retrieved|}{|retrieved|} \tag{14}$$

$$Recall@K = \frac{|relevant \cap retrieved|}{|relevant|} \tag{15}$$

F1-score@K is given by:

$$F1 - score@K = \frac{2 * Precision@K * Recall@K}{Precision@K + Recall@K} \tag{16}$$

Our loss function is mathematically formulated as follows:

$$\mathcal{L} = \alpha \mathcal{L}_{bcel} + \beta \mathcal{L}_{cont} \tag{17}$$

where  $\alpha$  and  $\beta$  are hyperparameters to weigh the contributions of the losses  $\mathcal{L}_{bcel}$  and  $\mathcal{L}_{cont}$  such that  $\alpha + \beta = 1$ .

For a round  $t$ , algorithm perceives context-action pair  $(s_t, a)$ , predicts a reciprocal compatibility score between  $u_t$  and  $a$  denoted by  $rCS(u_t, a) \in [0, 1]$ , then receives a reward  $r_{t,a}^*$ . Each input paired sample  $(u_t, a)$  has a corresponding reward  $r_{t,a}^*$ . Binary cross-entropy loss

is given by:

$$\mathcal{L}_{bcel} = (-1)/|U| \sum_{\forall u_t \in U} r_{t,a}^* \log(rCS(u_t, a)) + (1 - r_{t,a}^*) \log(1 - rCS(u_t, a)) \tag{18}$$

where set of all users is represented by  $U$  and  $|U|$  denotes total number of users.

Contrastive loss is formulated as follows:

$$\mathcal{L}_{conl} = \frac{1}{2M} \sum_{u_t \in U, a \in A} \mathbb{1}[u_t \sim a] \|f(u_t) - f(a)\|_2^2 + \mathbb{1}[u_t \not\sim a] \max(0, (m - \|f(u_t) - f(a)\|_2)) \tag{19}$$

In the above equation,  $M$  represents batch size of input-pair data.  $m$  is a hyperparameter that defines lower bound distance between different samples.  $\mathbb{1}$  stands for indicator or characteristic function. If the two candidate samples  $(u_t, a)$  are reciprocal match, it returns 1. Otherwise, it returns 0.

Beyond-accuracy metrics, we measured aggregate diversity and Gini coefficient for coverage and concentration respectively [41]. Aggregate Diversity and Gini coefficient are used to assess the results of XSiameseBiGRU-UCB in mitigating popularity bias.

Aggregate Diversity measures total number of distinct users among topN users recommended across all potential arms and is formally defined as:

$$\text{Aggregate Diversity} = \frac{|\bigcup_{u \in U} RR_u|}{|A|} \tag{20}$$

where  $A$  refers to the set of arms out of which recommendations can be predicted for  $u$  and  $RR_u$  represents list of users recommended to  $u$ .

In an ideal situation where each user is recommended equally Gini coefficient is 0, and 1 for extreme inequality of recommendation frequency for different users. The Gini coefficient calculates the frequencies of user  $v_i$  and user  $v_j$  being recommended out of  $n$  number of recommended users and formally defined as follows:

$$\text{Gini} = \frac{\sum_{i=1}^n \sum_{j=1}^n |v_i - v_j|}{2 \sum_{i=1}^n \sum_{j=1}^n v_j} \tag{21}$$

We propose mean exploration precision and mean exploration recall to assess XSiameseBiGRU-UCB regarding exploration i.e., reciprocal match discovery - how our model helps users discover new reciprocal matches. For any user  $u$ , suppose  $RR_u$  is the list of reciprocal recommendations generated and  $neX_u$  is the list of reciprocal recommendations which cannot be explained using factual or evident arguments.

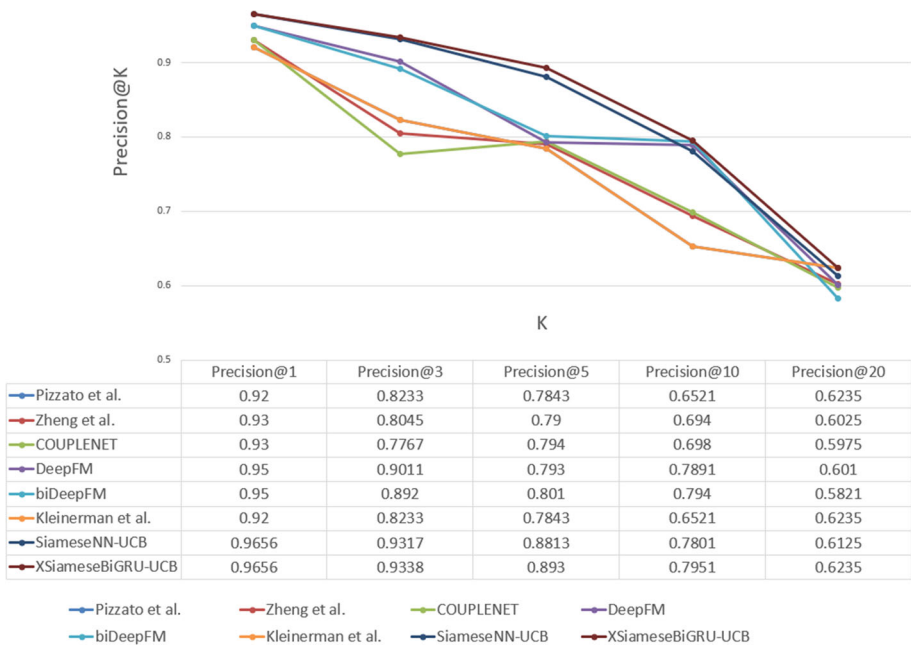
$$\text{Mean Exploration Precision (MEP)} = \frac{\sum_u \frac{|neX_u \cap RR_u|}{|RR_u|}}{|U|} \tag{22}$$

$$\text{Mean Exploration Recall (MER)} = \frac{\sum_u \frac{|neX_u \cap RR_u|}{|neX_u|}}{|U|} \tag{23}$$

To evaluate the results of explainability, we assessed the generated explanations using feature matching ratio [42] which tells us whether a feature is included in the generated explanation or not. It is defined as:

$$\text{Feature Matching Ratio} = \frac{1}{N} \sum_{u,i} \mathbb{1}(f_{u,a} \in eX_{u,a}) \tag{24}$$

Here,  $N$  denotes total number of explanations,  $eX_{u,a}$  denotes generated explanation for user-arm pair (having degree of strength of support  $\geq$  threshold-value) and  $f_{u,a}$  is the given feature.  $\mathbb{1}(x)$  is the indicator function.



**Figure 15** Precision@K of Pizzato et al. [1], Zheng et al. [9], COUPLENET [11], DeepFM [44], biDeepFM [45], Kleinerman et al. [43], SiameseNN-UCB [5] and XSiameseBiGRU-UCB

### 5.3.2 Hyperparameter sensitivity analysis

We performed experiments to examine the effect of batch size, loss function, optimizer, and learning rate of XSiameseBiGRU-UCB.

#### A. Batch Size

We experimented with different batch sizes (Table 3). With 64 batch size, validation accuracy (val\_acc) was highest and validation loss (val\_loss) was lowest across all the datasets used except D1.

#### B. Loss function

We combined binary cross-entropy and contrastive loss to improve the discriminative capability of our proposed model. Loss weights  $\alpha$  and  $\beta$  were varied in the range [0-1] to find an optimal balance between both the losses. We empirically set the loss weights  $\alpha$  and  $\beta$  to 0.3 and 0.7 since we got better performance with these values.

**Table 3** Ablation study of batch size in XSiameseBiGRU-UCB

Batch size	D1		D2		D3		D4	
	val_acc	val_loss	val_acc	val_loss	val_acc	val_loss	val_acc	val_loss
16	96.8812	0.7420	93.3211	0.8089	94.3221	0.7123	94.4367	0.6878
32	97.2122	<b>0.3912</b>	96.8423	0.6823	96.1121	0.6078	96.8878	0.5723
<b>64</b>	<b>97.3611</b>	0.4124	<b>97.8902</b>	<b>0.3223</b>	<b>97.3245</b>	<b>0.3867</b>	<b>96.9678</b>	<b>0.4313</b>
128	95.8323	0.5145	94.1089	0.9112	94.2082	0.7634	93.8688	0.8223



**Table 4** Ablation study with various optimizers and learning rates in XSiameseBiGRU-UCB

Optimizer	Learning rate	D1		D2		D3		D4	
		val_acc	val_loss	val_acc	val_loss	val_acc	val_loss	val_acc	val_loss
Stochastic Gradient Descent	$10^{-3}$	96.1103	0.7910	96.6720	0.7334	95.2087	0.8103	95.6667	0.7890
	$10^{-2}$	96.6667	0.4512	97.1232	0.3867	96.8322	0.6734	96.8875	0.6104
	$10^{-1}$	95.8876	0.8112	95.1078	0.7487	94.9767	0.8145	95.0123	0.7734
Adam	$10^{-3}$	96.8902	0.4667	97.0123	0.4443	97.3321	<b>0.3124</b>	96.8887	0.4323
	<b><math>10^{-2}</math></b>	<b>97.8789</b>	<b>0.3867</b>	<b>98.1211</b>	<b>0.2820</b>	<b>97.7823</b>	0.3209	<b>97.8876</b>	<b>0.3911</b>
	$10^{-1}$	95.3423	0.6123	95.1088	0.5834	94.2821	0.7212	95.0223	0.6012
Nadam	$10^{-3}$	97.1478	0.5623	96.8223	0.6687	96.7876	0.6423	96.6667	0.6212
	$10^{-2}$	97.2123	0.5334	96.2345	0.5876	96.8767	0.5743	97.0212	0.5645
	$10^{-1}$	95.4512	0.72087	95.0123	0.8214	95.8217	0.7427	95.2310	0.7805

C. Optimizer and learning rate

Table 4 shows the validation accuracy and loss values of XSiameseBiGRU-UCB with various optimizers and learning rates. We observed that the highest accuracy was achieved with Adam optimizer for learning rate of  $10^{-2}$ . Validation loss was lowest for datasets D1, D2 and D4 with Adam optimizer and  $10^{-2}$  learning rate.

5.3.3 Experimental results

XSiameseBiGRU-UCB was compared with the following state-of-the-art algorithms:

- RECON [1]: It is a content-based RRS which computes reciprocal compatibility scores of users on the basis of their profiles and actions.
- Zheng et al. [9]: A content-based RRS model on the basis of multi-criteria utility theory.
- COUPLENET [11]: A deep learning Siamese architecture with Gated Recurrent Units (GRU) encoders.
- Kleinerman et al. [43]: Authors present two explain methods namely, transparent and correlation-based to provide explanations for the recommendations generated by RECON.
- DeepFM [44]: Feature learning and factorization machines are integrated for generating recommendations.
- biDeepFM [45] is a multi-objective learning strategy that jointly evaluates the likelihood that an applicant will send an interest to the employer and the employer will also reciprocate. Harmonic mean is used to combine the unilateral preferences  $y'_u$  and  $y'_v$  obtained from biDeepFM [46].
- SiameseNN-UCB [5]: It follows Siamese architecture and uses UCB to tackle exploitation-exploration dilemma.

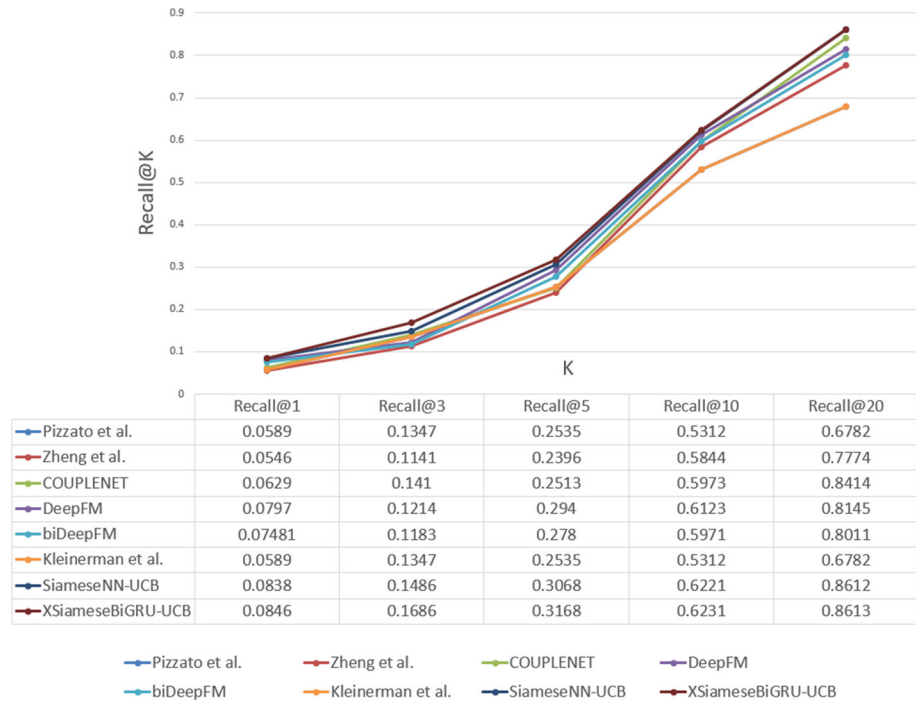
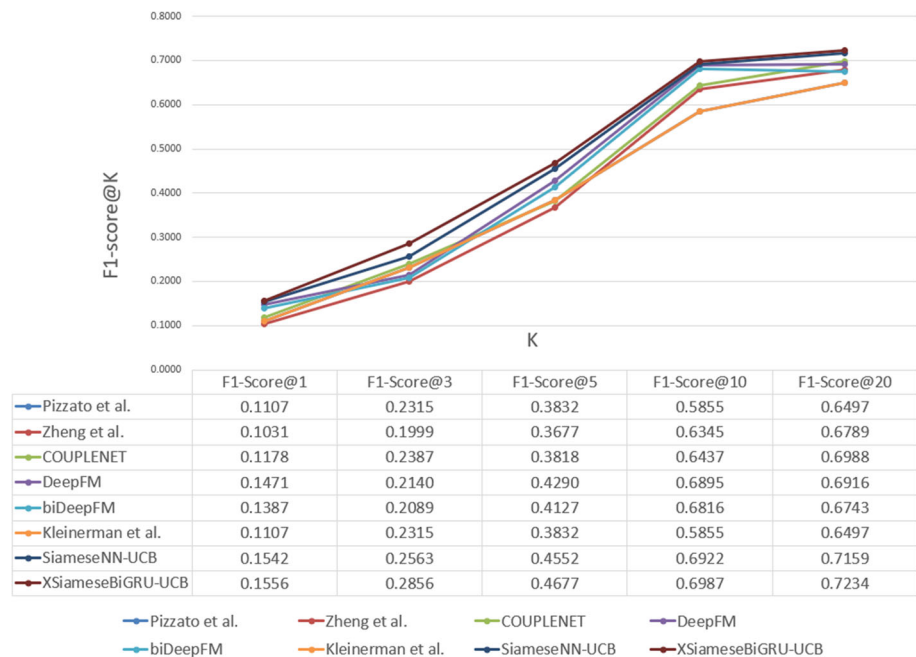


Figure 16 Recall@K of Pizzato et al. [1], Zheng et al. [9], COUPLENET [11], DeepFM [44], biDeepFM [45], Kleinerman et al. [43], SiameseNN-UCB [5] and XSiameseBiGRU-UCB

XSiameseBiGRU-UCB, COUPLENET [11], DeepFM [44], biDeepFM [45] and SiameseNN-UCB [5] were executed using tensorflow in Python programming language. XSiameseBiGRU-UCB, COUPLENET, DeepFM and SiameseNN-UCB were trained using Adam [47] optimizer and  $10^{-2}$  as the learning rate because we were able to obtain better results consistently in these approaches with these values. Batch size was tuned to 64. Early stopping call-back function was used to stop training when no improvement was observed in the validation loss values for fifteen consecutive epochs (patience value). A dropout of 0.3 was used for all the above mentioned deep neural network-based models. For DeepFM, a 3-layer perceptron with relu activation function was used. Dropout value of 0.8 and “constant” network shape was used as mentioned by authors in their paper [44]. Same architecture was adapted for biDeepFM also, with additional sigmoid extension. biDeepFM was trained using Weighted LogLoss with gradient descent as in [45].

We computed precision@K, recall@K and F1-score@K for a threshold value of 0.7 i.e., top-K reciprocal recommendations generated by the above approaches having reciprocal compatibility score greater than or equal to 0.7 for different values of K (1, 3, 5 and 10). The threshold value was reduced to 0.6 to obtain top-20 (K=20) recommendation results. As evident from the Figures 15, 16 and 17, there was a trade-off between precision and recall values. From Figures 15-17, it is evident that XSiameseBiGRU-UCB outperforms in terms of precision@K, recall@K and F1-score@K.

Generating accurate recommendations only is not always effective and profitable. Recommending only popular users as reciprocal match for other users may give high accuracy, but it also results in a decline of other desirable aspects such as recommendation diversity. Previous studies have shown that beyond accuracy, metrics such as diversity and gini coefficient should also be assessed for an effective recommender system [48]. To examine the effectiveness of XSiameseBiGRU-UCB we also computed aggregate diversity and Gini coefficient against baselines. A low value of aggregate diversity indicates that the reciprocal recommender system recommends only a small fraction of the users to target users, which in turn may negatively impact other users. It is important to have



**Figure 17** F1-score@K of Pizzato et al. [1], Zheng et al. [9], COUPLENET [11], DeepFM [44], biDeepFM [45], Kleinerman et al. [43], SiameseNN-UCB [5] and XSiameseBiGRU-UCB

**Table 5** Performance comparison results with respect to aggregate Diversity and Gini coefficient

Approach	D1		D2		D3		D4	
	Aggregate Diversity	Gini coefficient	Aggregate Diversity	Gini coefficient	Aggregate Diversity	Gini coefficient	Aggregate Diversity	Gini coefficient
Pizzato et al. [1]	0.2823	0.8400	0.2321	0.8870	0.2012	0.8434	0.2887	0.8422
heng et al. [9]	0.2321	0.8123	0.2334	0.8656	0.2153	0.8812	0.2410	0.8932
COUPLET [11]	0.2412	0.8503	0.2167	0.9223	0.2362	0.8612	0.2188	0.9177
DeepFM [44]	0.2321	0.8478	0.2624	0.8667	0.2134	0.8723	0.2400	0.8678
biDeepFM [45]	0.2016	0.8867	0.2563	0.9103	0.2126	0.8788	0.2023	0.8902
Kleiner et al. [43]	0.2823	0.8400	0.2321	0.8870	0.2012	0.8434	0.2887	0.8422
SiameseNN-UCB [5]	0.2914	0.7438	0.3087	0.7667	0.3240	0.6867	0.3124	0.6687
XSiameseBiGRU-UCB	<b>0.3124</b>	<b>0.6667</b>	<b>0.3467</b>	<b>0.6867</b>	<b>0.3620</b>	<b>0.6401</b>	<b>0.3640</b>	<b>0.6387</b>

**Table 6** Performance comparison results with respect to Mean Exploration Precision and Mean Exploration Recall

Approach	D1		D2		D3		D4	
	MEP	MER	MEP	MER	MEP	MER	MEP	MER
Pizzato et al. [1]	0.1221	0.0045	0.1767	0.0052	0.1311	0.0056	0.1213	0.0046
Zheng et al. [9]	0.1024	0.00312	0.1120	0.0026	0.1847	0.0062	0.1221	0.0034
COUPLETNET [11]	0.2011	0.0068	0.1980	0.0051	0.2212	0.0105	0.1893	0.0058
DeepFM [44]	0.1231	0.0028	0.165	0.0049	0.1123	0.0052	0.1876	0.0049
biDeepFM [45]	0.1238	0.0067	0.1562	0.0078	0.1271	0.012	0.1286	0.0073
Kleinerman et al. [43]	0.2815	0.01123	0.2323	0.0108	0.2013	0.0201	0.2764	0.0167
SiameseNN-UCB [5]	0.2989	0.02814	0.3137	0.0282	0.2112	0.0323	0.2934	0.0208
XSiameseBiGRU-UCB	<b>0.3109</b>	<b>0.0311</b>	<b>0.3363</b>	<b>0.0372</b>	<b>0.2912</b>	<b>0.0423</b>	<b>0.3034</b>	<b>0.0278</b>

high aggregate diversity values for a more even distribution of users in the recommendation lists. Gini coefficient is another widely used metric to examine whether the recommendations have fair distribution across all the recommended users or not. A greater value of Gini coefficient indicates a stronger concentration of the recommendations, e.g., on popular users. Table 5 demonstrates the results of aggregate diversity and Gini coefficient on the datasets used. Comparison results in Table 5 highlight that XSiameseBiGRU-UCB outperforms with respect to these metrics when compared with the baselines. With XSiameseBiGRU-UCB, we are able to achieve better coverage and a better distribution as shown by its low Gini coefficient compared to the baselines.

To evaluate XSiameseBiGRU-UCB in terms of exploration or explainability of users who are recommended, we used the proposed mean exploration precision and mean exploration recall. Table 6 shows the results of mean exploration precision and mean exploration recall on all the datasets used. We found empirically that the mean exploration precision and mean exploration recall were better in XSiameseBiGRU-UCB compared to the baselines.

Feature Matching Ratio is used to evaluate the generated explanation at the feature level. Table 7 shows the results of generated explanations with respect to feature matching ratio. The results show the values of feature matching ratio of generated explanations with  $k=1, 3, 5, 10$  and  $20$ . As shown in Table 7, more than 62% features are included in the generated explanations.

## 6 Conclusion

An effective reciprocal recommender system should be able to generate reciprocal recommendations along with personalized and intuitive explanations for the generated recommendations. In this paper, we proposed XSiameseBiGRU-UCB which provided post-hoc argumentation-based explanations for the reciprocal recommendations generated using contextual bandits framework. Contextual bandits

**Table 7** Feature Matching Ratio of XSiameseBiGRU-UCB

XSiameseBiGRU-UCB	D1	D2	D3	D4
Top-1	0.6343	0.6532	0.6212	0.6467
Top-3	0.6403	0.7667	0.6783	0.7240
Top-5	0.7134	0.7803	0.6304	0.6778
Top-10	0.7289	0.7889	0.7345	0.7211
Top-20	0.7845	0.8302	0.7630	0.7543

framework was used to address exploitation and exploration dilemma in RRS. We provided personalized explanations using arguments in support of the claim i.e., generated reciprocal recommendations. To generate arguments in support of a reciprocal recommendation, we used contextual information available in user profiles. We computed arguments' degree of support for the claim with Sorensen-Dice coefficient (SDC) and Siamese Bi-directional Gated Recurrent Units network semantic model based on Sentence-BERT (SBiGRU-SBERT). With XSiameseBiGRU-UCB, we addressed single and two-disjoint class RRS with direct as well as indirect reciprocity while incorporating various aspects such as demographics, interests, exploratory behaviour and key intent along with the user's given preferences.

XSiameseBiGRU-UCB was compared against state-of-the-art approaches on four real-world datasets. The results highlight its effectiveness w.r.t. number of evaluation criteria such as precision, recall, F1-score, aggregate diversity, Gini coefficient, mean exploration precision, mean exploration recall and feature matching ratio.

**Author Contributions** Ms Tulika Kumari: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Resources, Software, Validation, Visualization, Roles/Writing - original draft, Writing - review and editing. Dr. Bhavna Gupta: Conceptualization, Methodology, Supervision, Review, Editing, Project Administration, Resources, Validation. Dr. Ravish Sharma: Conceptualization, Methodology, Supervision, Review, Editing, Project Administration, Resources, Validation. Prof. Punam Bedi: Conceptualization, Methodology, Supervision, Review, Editing, Project Administration, Resources, Validation

**Funding** \*No funding was received to assist with the preparation of this manuscript.

**Data Availability** No datasets were generated. Datasets used in the current study are publicly available.

## Declarations

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

**Ethical standard** The authors have no competing interests to declare that are relevant to the content of this article.

## References

1. Pizzato, L., Rej, T., Chung, T., Koprinska, I., Kay, J.: Recon: a reciprocal recommender for online dating. In: Proceedings of the Fourth ACM Conference on Recommender Systems, pp. 207–214 (2010)
2. Malinowski, J., Keim, T., Wendt, O., Weitzel, T.: Matching people and jobs: a bilateral recommendation approach. In: Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06), vol. 6, pp. 137–137 (2006). IEEE
3. Li, C.-T.: Mentor-spotting: recommending expert mentors to mentees for live trouble-shooting in code-mentor. *Knowl. Inf. Syst.* **61**(2), 799–820 (2019)
4. Xu, F., Uszkoreit, H., Du, Y., Fan, W., Zhao, D., Zhu, J.: Explainable ai: a brief survey on history, research areas, approaches and challenges. In: CCF International Conference on Natural Language Processing and Chinese Computing, pp. 563–574 (2019). Springer
5. Kumari, T., Sharma, R., Bedi, P.: A contextual-bandit approach for multifaceted reciprocal recommendations in online dating. *J. Intell. Inf. Syst.* **59**(3), 705–731 (2022)
6. Li, L., Chu, W., Langford, J., Schapire, R.E.: A contextual-bandit approach to personalized news article recommendation. In: Proceedings of the 19th International Conference on World Wide Web, pp. 661–670 (2010)
7. Kumari, T., Sharma, R., Bedi, P.: Siamese bi-directional gated recurrent units network for generating reciprocal recommendations in online job recommendation. In: International Conference on Innovative Computing and Communications, pp. 257–269 (2023). Springer

8. Krzywicki, A., Wobcke, W., Kim, Y.S., Cai, X., Bain, M., Mahidadia, A., Compton, P.: Collaborative filtering for people-to-people recommendation in online dating: data analysis and user trial. *Int. J. Human-Comput. Stud.* **76**, 50–66 (2015)
9. Zheng, Y., Dave, T., Mishra, N., Kumar, H.: Fairness in reciprocal recommendations: a speed-dating study. In: *Adjunct Publication of the 26th Conference on User Modeling, Adaptation and Personalization*, pp. 29–34 (2018)
10. Zheng, Y., Pu, A.: Utility-based multi-stakeholder recommendations by multi-objective optimization. In: *2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pp. 128–135 (2018). IEEE
11. Tay, Y., Tuan, L.A., Hui, S.C.: Couplenet: paying attention to couples with coupled attention for relationship recommendation. In: *Twelfth International AAAI Conference on Web and Social Media* (2018)
12. Neve, J., Palomares, I.: Latent factor models and aggregation operators for collaborative filtering in reciprocal recommender systems. In: *Proceedings of the 13th ACM Conference on Recommender Systems*, pp. 219–227 (2019)
13. Neve, J., Palomares, I.: Hybrid reciprocal recommender systems: integrating item-to-user principles in reciprocal recommendation. In: *Companion Proceedings of the Web Conference 2020*, pp. 848–853 (2020)
14. Prabhakar, S., Spanakis, G., Zaiane, O.: Reciprocal recommender system for learners in massive open online courses (moocs). In: *International Conference on Web-Based Learning*, pp. 157–167 (2017). Springer
15. Guy, I.: People recommendation on social media. In: *Social Information Access*, pp. 570–623 (2018). Springer
16. Shimizu, R., Matsutani, M., Goto, M.: An explainable recommendation framework based on an improved knowledge graph attention network with massive volumes of side information. *Knowl.-Based Syst.* **239**, 107970 (2022)
17. Peake, G., Wang, J.: Explanation mining: post hoc interpretability of latent factor models for recommendation systems. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2060–2069 (2018)
18. Zhang, M., Wang, G., Ren, L., Li, J., Deng, K., Zhang, B.: Metonr: a meta explanation triplet oriented news recommendation model. *Knowl.-Based Syst.* **238**, 107922 (2022)
19. Kleinerman, A., Rosenfeld, A., Ricci, F., Kraus, S.: Supporting users in finding successful matches in reciprocal recommender systems. *User Modeling and User-Adapted Interaction* **31**(3), 541–589 (2021)
20. Vassiliades, A., Bassiliades, N., Patkos, T.: Argumentation and explainable artificial intelligence: a survey. *Knowl. Eng. Rev.* **36** (2021)
21. Chesñevar, C., Maguitman, A.G., González, M.P.: Empowering recommendation technologies through argumentation. In: *Argumentation in Artificial Intelligence*, pp. 403–422 (2009). Springer
22. Briguez, C.E., Budan, M.C., Deagustini, C.A., Maguitman, A.G., Capobianco, M., Simari, G.R.: Argument-based mixed recommenders and their application to movie suggestion. *Expert Syst. Appl.* **41**(14), 6467–6482 (2014)
23. García, A.J., Simari, G.R.: Defeasible logic programming: an argumentative approach. *Theory Pract. Log. Program.* **4**(1–2), 95–138 (2004)
24. Bedi, P., Vashisth, P.: Argumentation-enabled interest-based personalised recommender system. *J. Exp. Theor. Artif. Intell.* **27**(2), 199–226 (2015)
25. Naveed, S., Donkers, T., Ziegler, J.: Argumentation-based explanations in recommender systems: conceptual framework and empirical results. In: *Adjunct Publication of the 26th Conference on User Modeling, Adaptation and Personalization*, pp. 293–298 (2018)
26. Toulmin, S.E.: *The Uses of Argument*. Cambridge University Press, (2003)
27. Briguez, C.E., Budán, M.C., Deagustini, C.A., Maguitman, A.G., Capobianco, M., Simari, G.R.: Towards an argument-based music recommender system. In: *Computational Models of Argument*, pp. 83–90 (2012). IOS Press
28. Heras, S., Navarro, M., Botti, V., Julián, V.: Applying dialogue games to manage recommendation in social networks. In: *International Workshop on Argumentation in Multi-Agent Systems*, pp. 256–272 (2009). Springer
29. Van Eemeren, F.H., Jackson, S., Jacobs, S.: *Argumentation*. In: *Argumentation Library*, pp. 3–25 (2015)
30. Governatori, G., Sartor, G.: Burdens of proof in monological argumentation. In: *Legal Knowledge and Information Systems*, pp. 57–66 (2010)
31. Besnard, P., Hunter, A.: *Elements of Argumentation* vol. 47. MIT Press Cambridge, (2008)
32. Lattimore, T., Szepesvári, C.: *Bandit Algorithms*. Cambridge University Press, (2020)
33. Auer, P.: Using confidence bounds for exploitation-exploration trade-offs. *J. Mach. Learn. Res.* **3**(Nov), 397–422 (2002)
34. Auer, P., Cesa-Bianchi, N., Fischer, P.: Finite-time analysis of the multiarmed bandit problem. *Mach Learn* **47**(2), 235–256 (2002)

35. Sorensen, T.A.: A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on danish commons. *Biol. Skar.* **5**, 1–34 (1948)
36. Dice, L.R.: Measures of the amount of ecologic association between species. *Ecology* **26**(3), 297–302 (1945)
37. Fisman, R., Iyengar, S.S., Kamenica, E., Simonson, I.: Gender differences in mate selection: evidence from a speed dating experiment. *Q. J. Econ.* **121**(2), 673–697 (2006)
38. Su, X., Hu, H.: Gender-specific preference in online dating. *EPJ Data Sci* **8**(1), 12 (2019)
39. Veaux, R.D.D., Ungar, L.H.: Multicollinearity: a tale of two nonparametric regressions. In: *Selecting Models from Data*, pp. 393–402 (1994)
40. Reimers, N., Gurevych, I.: Sentence-bert: sentence embeddings using siamese bert-networks. [arXiv:1908.10084](https://arxiv.org/abs/1908.10084) (2019)
41. Anelli, V.W., Bellogín, A., Di Noia, T., Jannach, D., Pomo, C.: Top-n recommendation algorithms: a quest for the state-of-the-art. [arXiv:2203.01155](https://arxiv.org/abs/2203.01155) (2022)
42. Li, L., Zhang, Y., Chen, L.: Generate neural template explanations for recommendation. In: *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pp. 755–764 (2020)
43. Kleinerman, A., Rosenfeld, A., Kraus, S.: Providing explanations for recommendations in reciprocal environments. In: *Proceedings of the 12th ACM Conference on Recommender Systems*, pp. 22–30 (2018)
44. Guo, H., Tang, R., Ye, Y., Li, Z., He, X.: Deepfm: a factorization-machine based neural network for ctr prediction. *arXiv preprint* [arXiv:1703.04247](https://arxiv.org/abs/1703.04247) (2017)
45. Yıldırım, E., Azad, P., Öğüdücü, ŞG.: Bideepfm: a multi-objective deep factorization machine for reciprocal recommendation. *Eng. Sci. Technol. Int. J.* **24**(6), 1467–1477 (2021)
46. Kumari, T., Sharma, R., Bedi, P.: Multifaceted reciprocal recommendations for online dating. In: *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*, pp. 1–6 (2021). IEEE
47. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. [arXiv:1412.6980](https://arxiv.org/abs/1412.6980) (2014)
48. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B., Malthouse, E.: User-centered evaluation of popularity bias in recommender systems. In: *Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization*, pp. 119–129 (2021)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.