Expert Systems With Applications 115 (2019) 386–402

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Robustness analysis of multi-criteria collaborative filtering algorithms against shilling attacks

Ahmet Murat Turk, Alper Bilge*

Department of Computer Engineering, Anadolu University, Eskisehir 26555, Turkey

ARTICLE INFO

Article history:
Received 2 February 2018
Revised 31 July 2018
Accepted 1 August 2018
Available online 3 August 2018

Keywords:
Collaborative filtering
Multi-criteria
Shilling attack
Profile injection
Robustness analysis
Mode attack

ABSTRACT

Collaborative filtering is an emerging recommender system technique that aims guiding users based on other customers preferences with behavioral similarities. Such correspondences are located based on preference history of users. A relatively new extension of traditional collaborative filtering schemes takes into account not only how much a user likes an item, but also why she likes the item by collecting multi-criteria preferences focusing on distinctive features of the items. These multi-criteria collaborative filtering systems have the potential to improve recommender system accuracy since they reveal multiple views of users on products. However, due to providing more insightful recommendations, such systems might be subjected to malicious attacks more substantially than the traditional ones. Attackers attempt to insert fake profiles to bias outputs of these systems in favor of a particular product or disrepute the system itself. Since outputs of expert systems directly dependent on input signals; interventions to the inputs coherently cause failures on productions of such systems. In this study, we examine shilling attack strategies against multi-criteria preference collections, how to extend well-known attack scenarios against these systems, and propose an alternative attacking scheme. We analyze the robustness of baseline multi-criteria recommendation algorithms regarding various similarity aggregation procedures against proposed attacking schemes by the extensive experimental investigation. Empirical results on real-world data demonstrate that these systems are highly vulnerable to manipulations and proper attack detection practices are needed to ensure recommendation quality. According to our findings, manipulative attempts at such expert systems mislead decision-making process.

1. Introduction

With increasing amount of information people daily face through the Internet, they are in need of filtering useful context. Decision-making for an individual desiring to choose between products, such as a movie to watch, is time-consuming. However, online retailers, stream services, and social networks offer automated recommendations to help users discover relevant or exciting products and retrieve valuable services. Recommender systems are intelligent tools in producing such qualified referrals that help users cope with the enormous amount of information they confront (Lakiotaki, Tsafarakis, & Matsatsinis, 2008) and support online service providers in increasing sales and boosting the popularity of their system (Jannach, Karakaya, & Gedikli, 2012). Such solutions are omnipresent on the web and included in diverse areas such as movies, music, articles, hotels, television, books, restaurant, e-shopping, and web search (Nilashi, bin Ibrahim, & Ithnin, 2014a). These expert systems essentially automate the fundamental human instinct of asking trusted ones for advice and mimic word-of-mouth.

Collaborative filtering (CF) (Herlocker, Konstan, Borchers, & Riedl, 1999) is one of the most popular recommendation techniques. CF systems offer personalized recommendations by constituting a collaboration among people who have similar tastes. The assumption behind such collaboration is that individuals who agree regarding their preferences or tastes in the past would likely to agree in the future, as well (Sanchez-Vilas, Ismoilov, Lousame, Sanchez, & Lama, 2011). Based on this assumption, CF systems try to estimate referrals on future inclinations of users by utilizing a database of preference collections. Such recommendations are based on a two-step process of locating like-minded users initially, referred to as neighbors, and estimating a prediction based on neighbors' opinions on the target item. Recommendation outputs might be presented as a top-*N* list of commendable items indicating the most inspiring products specifically for the user, as well as an explicit prediction score on a particular item designating an

* Corresponding author.
E-mail addresses: ahmetmuratturk@anadolu.edu.tr (A.M. Turk), abilge@anadolu.edu.tr (A. Bilge).

https://doi.org/10.1016/j.eswa.2018.08.001
0957-4174/© 2018 Elsevier Ltd. All rights reserved.

estimation of liking degree for the item. For a book recommender system, for instance, recommendations might motive the user into unread books of possible interest and predictions help the user to decide whether to purchase a book or not.

Considering the two-step process of recommendation generation in CF systems, the overall success of the system is strictly bound to the neighborhood formation phase which relies on preference histories of users (Nilashi, bin Ibrahim, & Ithnin, 2014b). User accounts contain a single preference value for each item indicating the degree of appreciation which constitutes a base for calculating the similarity of tastes among users. However, such preference data is restricted and does not expose the perspective of the user to why a particular rating is placed for the item or which properties of the item are compelling for the user. Adomavicius and Kwon (2007) criticize such a limitation of user modeling and propose a new approach suggesting an evaluation of items along with their sub-aspects. Such a multi-perspective evaluation scheme is claimed to allow users to express their authentic preferences better. For instance, Yahoo!Movies platform collects feedback about movies on specific aspects of *story, acting, directing*, and *visual effects* along with an *Overall* score. Therefore, users can express more fine-grained preferences such as they like a movie but are not satisfied with acting, or they do not like it at all but the visual effects. Zagat Survey (Lee & Teng, 2007) allows people to rate restaurants based on *quality of food, décor*, and *service*; and provides recommendations of points of interest. The multi-perspective rating collection is acclaimed to potentially increase the recommendation accuracy since it enables better personalization opportunities compared to single-rating systems. CF systems utilizing such multi-criteria data are called Multi-Criteria Collaborative Filtering (MCCF) systems. MCCF framework presents an open research area where relations among criteria and the significance of sub-preferences on the recommendation model require in-depth analysis.

Outcomes of expert systems directly depend on training data employed in the model-building process which makes them fault intolerant to the properties of the input. Corrupted, noisy, or manipulated information misdirect these systems to poor decisions and uncertainty. Therefore, robustness is essential for ensuring highly reliable outcomes and research conducted in robust expert systems deals with noise removal in system input (Nguyen et al., 2015; Risnumawan, Shivakumara, Chan, & Tan, 2014). Recommender systems, as a variation of experts systems, suffer from such issues since they naïvely assume that collected preferences are always submitted with goodwill. However, the Internet is inhabited by malicious people who aim at exploiting such systems for their benefit. Therefore, robustness analyses are performed, and defense mechanisms are developed to defend expert systems against such malevolent attempts.

Most recommender systems, including MCCF systems, are publicly available to new users since they are dependent on the collection of qualified preference data (Yilmazel & Kaleli, 2016). However, due to their entailment on implicit or explicit rating data and the underlying assumption driving neighborhood formation procedure, they are vulnerable to the malicious action of a fake profile injection. Shilling attacks are performed by implanting such malicious profiles with the aim of either increasing the possibility of an item recurring in the recommendation lists, referred to as push attacks, or demoting competitors' products and causing the abatement of sales, referred to as nuke attacks (Mobasher, Burke, Bhaumik, & Williams, 2007). Essentially, the goal of inserting shilling profiles is to appear in the neighborhood of users with numerous fake profiles and manipulate recommendation outcomes. Researchers have extensively studied the effects of these attacks against traditional CF systems (Hurley, O'Mahony, & Silvestre, 2007), propose mechanisms to detect and avoid them (Burke, Mobasher, Williams, & Bhaumik, 2006) and develop robust recommendation algorithms rendering them futile (Cheng & Hurley, 2010b).

Due to their more complicated preference data structure and recommendation approach, MCCF systems might be subjected to more sophisticated shilling attempts. Although MCCF systems have been widely discussed in the literature, the effects of shilling attacks against such systems have not been analyzed yet. Also, well-known shilling attack strategies might be extended to multi-criteria platforms in a diverse way which makes them harder to detect and passivate. In this study, we mainly discuss multi-criteria profile injection strategies and scrutinize effects of shilling attack challenge against primary MCCF schemes.

Contributions of the study can be listed as follows:

1. Applicable extension methods of well-known shilling attack strategies are discussed against the multi-criteria domain.
2. A novel shilling attack scheme is proposed exploiting data distribution of MCCF systems.
3. A comprehensive empirical robustness analysis of primary MCCF algorithms is performed against various profile injection attacks.

The rest of the study is organized as follows: Section 2 introduces a brief background on multi-criteria recommendation approach and popular shilling attack types. Section 3 presents related and milestone studies in the fields of MCCF and shilling attacks research. Section 4 discusses shilling profile design paradigms in the multi-criteria domain, provides applicable extensions of well-known attacking strategies, and presents a novel attacking strategy against MCCF systems. Section 5 performs a real data-based extensive robustness analysis of primary MCCF systems against proposed attack models and interprets empirical results. Finally, the study concludes by discussing gained insights and related future works in Section 6.

2. Background

In this section, we briefly explain state-of-the-art MCCF algorithms and shilling attack concepts in CF systems. Besides the methods based on multi-criteria decision-making, MCCF systems exclusively utilize multi-criteria rating collections, and most of the proposed methods aim at improving the scheme described by Adomavicius and Kwon (2007). Also, since some of the proposed attacks in literature are not practical due to reliance on detailed and accurate information on preference collection (Gunes, Kaleli, Bilge, & Polat, 2014), we discuss effects of the most well-known attacking strategies.

2.1. Multi-criteria collaborative filtering

In CF systems, recommendation quality heavily relies on the quality of collected preferences. If the rating scheme of the recommender system restricts the user to provide an overall liking degree, users might not feel satisfied in expressing their opinions which might result in poor recommendations. People might criticize a certain aspect of a service even if they appreciate it in general. The multi-criteria rating scheme is more convenient to evaluate liking degree of a user experience. Furthermore, some people might be more interested in certain aspects of a product. Adomavicius and Kwon (2007) explain how multi-criteria preferences might affect recommendation results with an example provided in Table 1.

Considering the rating collection in Table 1, the indices represent the given rating for sub-criteria, and the *Overall* score rating is the average of those ratings. In traditional CF approach, if u_1 requests a prediction for i_5, u_2 and u_3 are located as the nearest

Table 1
A sample multi-criteria preference dataset.

	i_1	i_2	i_3	i_4	i_5
u_1	$5_{2,2,8,8}$	$7_{5,5,9,9}$	$5_{2,2,8,8}$	$7_{5,5,9,9}$	?
u_2	$5_{8,8,2,2}$	$7_{9,9,5,5}$	$5_{8,8,2,2}$	$7_{9,9,5,5}$	9
u_3	$5_{8,8,2,2}$	$7_{9,9,5,5}$	$5_{8,8,2,2}$	$7_{9,9,5,5}$	9
u_4	$6_{3,3,9,9}$	$6_{4,4,8,8}$	$6_{3,3,9,9}$	$6_{4,4,8,8}$	5
u_5	$6_{3,3,9,9}$	$6_{4,4,8,8}$	$6_{3,3,9,9}$	$6_{4,4,8,8}$	5

neighbors of u_1 based on their preference history. Therefore, the prediction result is expected for u_1 to like the item. Indeed, despite the identical *Overall* scores, these users are quite different in their tastes and views according to their sub-criteria preferences. Multi-criteria approach concludes that u_4 and u_5 are better matches for u_1 considering sub-aspects of the products. Hence, the prediction result is expected for u_1 to dislike the item.

2.1.1. Similarity-based approaches

Traditional CF approach relies on constructing neighborhood of the active user (a) by calculating similarities with other users within the system (Herlocker, Konstan, Terveen, & Riedl, 2004) via various metrics such as Pearson's correlation coefficient, adjusted-cosine similarity, and distance metrics. Such neighborhood formation in MCCF is performed by calculating similarities between a and any user u separately for all available criteria ratings and aggregating those similarities by (i) either averaging them or (ii) assuming the lowest value as given in Eqs. (1) and (2), respectively (Bilge & Kaleli, 2014; Hu, Chiu, Liao, & Li, 2015; Maneeroj, Samatthiyadikun, Chalermpornpong, Panthuwadeethorn, & Takasu, 2012).

$$sim_{\text{avg}}(a,u) = \frac{1}{k}\sum_{i=1}^{k} sim_i(a{,}u) \tag{1}$$

$$sim_{\text{min}}(a,u) = min_{i=1\ldots k}(sim_i(a{,}u)) \tag{2}$$

Finally, a prediction (p_{aq}) for a on the target item (q) is calculated as the simple arithmetic mean or weighted average of neighbors' ratings on q as given in Eq. (3).

$$p_{aq} = \overline{r_a} + \frac{\sum_{u\in N}(r_{u,q} - \overline{r_u}) \times sim(a,u)}{\sum_{u\in N} sim(a,u)} \tag{3}$$

where r_{uq} is the rating of u on q, $\overline{r_u}$ is the average rating of u, N is the set of a's neighbors, and sim_{au} is the similarity weight between a and u.

2.1.2. Aggregation function-based approaches

Based on the assumption that recognition of each criterion for a user can be extracted from multi-criteria ratings, these methods investigate user profiles to identify weights of each sub-criterion on the *Overall* rating as in the form $r_0 = w_1 r_1 + w_2 r_2 + \cdots + w_{k-1} r_{k-1}$ where r_0 is the *Overall* rating, r_i and w_i $(i = 1, \ldots, k-1)$ denote the rating and weight of each sub-criterion, respectively. Such aggregation function can be obtained by statistical techniques such as linear regression (Adomavicius & Kwon, 2007) and support vector regression (Jannach et al., 2012), and machine learning techniques such as neural networks (Nilashi et al., 2014a; Nilashi, bin Ibrahim, Ithnin, & Sarmin, 2015). Once the aggregation function is formed, predictions are generated for each sub-criterion using traditional CF techniques and aggregated by the obtained function.

2.2. Shilling attack profiles

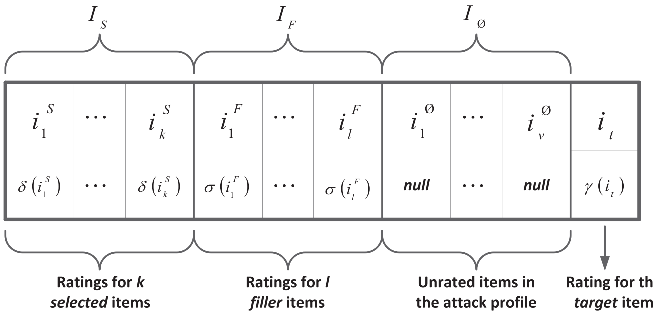


Fig. 1. Generalform of an attack profile (Gunes et al., 2014).

The general form of a shilling profile is the combination of four item sets as shown in Fig. 1 (Gunes et al., 2014). Selected items set, I_S, defines the characteristics of the attack and could be the popular items or the items that have common features. Filler items set, I_F, aims to disguise malicious profile to obstruct detection. Finally, $I_\varnothing$ and i_t denote the set of unrated items and rating for the targeted item, respectively.

Shilling attacks can be categorized based on their motivation behind as all attacks aim to either push or nuke a targeted item's popularity to gain an economic advantage over competitors. Push attacks intend to manipulate the recommender system in favor of a product to be recommended more often. Consequently, nuke attacks aim to reduce recommendation frequency of a competitor product. Based on the intent of the attack, i_t can be either maximum or minimum available rating in the system, respectively. Construction parameters and possible intents of attacks that are investigated within MCCF context are summarized in Table 2.

In Table 2, r_{max} and r_{min} represent the highest and the lowest possible votes in the system, respectively. Also, note that assigned ratings to i_t define the intent of the attack and some attacks can be mounted for both intents.

3. Related work

Traditional CF schemes rely on users opinions on items to discover relations among users and items, and estimate predictions based on like-minded users preferences on the target item using a weighted average or provide recommendations consisting of the most commendable items (Herlocker et al., 1999). Although CF dates back to more than two decades and is an extensively studied research area (Herlocker et al., 2004), MCCF is a relatively new approach branched within traditional CF literature. Initially, multi-criteria recommendation generation problem is perceived as a multi-criteria decision-making approach (Adomavicius, Manouselis, & Kwon, 2011; Manouselis & Costopoulou, 2007a). Primarily, Adomavicius and Tuzhilin (2005) proposed multidimensionality concept in recommender systems to adjust context of recommendations and collection of ratings in multiple aspects of items. Following these extension proposals, Adomavicius and Kwon (2007) developed an MCCF framework based on two different approaches where the former aims at extending conventional methods to multi-criteria preference domain and the latter originally focus on building a preference model based on relations and aggregations over criteria. Such framework is the first to bring out refining recommendations using sub-criteria.

In addition to multidimensional similarity calculation approach of Adomavicius and Kwon (2007), Manouselis and Costopoulou (2007b) comprehensively studies on conventional extension techniques of CF algorithms into MCCF domain. These methods focus on various algorithmic steps of CF recommendation such as calculating similarities, forming a neighborhood, feature weighting, and similarity merging.

Table 2
Features of the attack schemes.

Attack type	Selected items		Filler items	Intent	Target item ratings
	Selection method	Rating			
Random	Not used	–	System mean	Push/nuke	$r_{\max}/r_{\min}$
Average	Not used	–	Item mean	Push/nuke	$r_{\max}/r_{\min}$
Bandwagon	Popular items	$r_{\max}$	System mean	Push	$r_{\max}$
Reverse bandwagon	Unpopular items	$r_{\min}$	System mean	Nuke	$r_{\min}$
Love/hate	Not used	–	$r_{\max}$	Nuke	$r_{\min}$

The model-based approach of Adomavicius and Kwon (2007) obtains an aggregation function that relates sub-criteria to an overall liking degree. Such function can be obtained through domain expertise, statistical techniques, or machine learning methods. Once the aggregation function is built, estimations for the sub-criteria are weighted into an overall prediction value. Other than linear regression method proposed by Adomavicius and Kwon (2007) and Jannach et al. (2012) proposes support vector regession to achieve higher accuracy in predictions and overcoming sparsity issues. They also suggest combining weighted user- and item-based regression models.

MCCF systems solely rely on user preferences and hence accessible by public users which make them vulnerable to manipulations by malicious profile injection. Shilling attacks research can be divided into three broad areas (Bilge, Gunes, & Polat, 2014), i.e., robustness analysis of existing schemes, malicious profile detection methods, and developing robust algorithms. This study focuses on analyzing the robustness of state-of-the-art MCCF schemes against most well-known shilling attack strategies.

Although there were studies on fake profile injection into CF systems (Lam & Riedl, 2004; O'Mahony, Hurley, & Silvestre, 2003) is the first to coin the term *shilling attack* where authors examine *random* and *average attack* implementations on the most well-known CF schemes. Random attack simply constructs a shilling profile consisting of random ratings with a normal distribution to arbitrarily selected items, and average attack follows a similar approach by inserting item mean ratings instead of random ones to selected items. Burke, Mobasher, and Bhaumik (2005) and O'Mahony, Hurley, and Silvestre (2005) propose *bandwagon attack* which intends to build a limited knowledge attack scheme with publicly known popular item sets. Such approach introduces the concept of required knowledge to perform an attack. Further research (Burke et al., 2006; Burke, Mobasher, Zabicki, & Bhaumik, 2005) investigates the random, average, and bandwagon attacks. Mobasher, Burke, Bhaumik, and Sandvig (2007) and Zhang (2009) discuss *reverse bandwagon* strategy to disrepute an item by following a contrary approach to bandwagon attack and exploiting interest to unpopular items. The main difference between these two attacks types is that the intent of the attack where bandwagon pushes the target items and reverse bandwagon nukes. Mobasher, Burke, Bhaumik, and Sandvig (2007) discuss another simple yet effective nuking strategy called love/hate attack, provide detailed definitions of several attacking schemes and demonstrate experimental outcomes of applying these attacks. In addition to these attacks *segment attack* (Mobasher, Burke, Bhaumik, & Sandvig, 2007), *probe attack* (Burke, Mobasher et al., 2005; Mobasher, Burke, Bhaumik, & Sandvig, 2007; O'Mahony et al., 2005), *perfect knowledge attack* (Burke, Mobasher et al., 2005), *popular attack* (O'Mahony et al., 2003), and *power user attack* (Wilson & Seminario, 2013) are among other attacking strategies.

Shilling attacks can be categorized as low-knowledge and informed (high-knowledge) attacks according to the amount of required information to mount a successful attack. In low-knowledge attacks such as bandwagon, segment, reverse bandwagon, and love/hate attacks, the attacker has only access to publicly known information such as top-10 lists or specific segment products such as baby diapers. Informed attacks, on the other hand, utilize all kinds of information that can hardly be gathered without inside intervention to the system such as the set of rated items for particular users, probing the recommender system and using its recommendations as a means to select which items to get into neighborhoods of genuine users. Popular, probe, and power user attacks are among high-knowledge attacking strategies (Burke, O'Mahony, & Hurley, 2015; O'Mahony et al., 2005; Wilson & Seminario, 2013).

Research on multi-criteria recommendation generation focuses on solving traditional challenges of CF systems such as accuracy, sparsity, scalability, and cold start problems (Jannach, Zanker, & Fuchs, 2014; Nilashi et al., 2014b) on multi-criteria data or developing new recommendation techniques based on such collections (Fan & Xu, 2013; Maneeroj et al., 2012). As an alternative direction, Yi and Zhang (2016) propose a robust algorithm based on a multidimensional trust model and resistant against shilling attacks for traditional single-criterion preferences. Although there exists a comprehensive research on developing multi-criteria recommenders and multi-criteria preference collection well informs the recommender systems about users and attackers, how to perform shilling attacks on such systems and their robustness against manipulations are not discussed. Table 3 provides a brief overview of the robustness research in the recommender systems field. In summary related the scope of this study, researchers focus on developing novel attacking scenarios and analyzing robustness of single-criterion recommendation methods. However, this work proposes a novel attacking scheme and aims to investigate how shilling attack concepts can be applied to state-of-the-art MCCF recommendation schemes and measure their robustness in recommendation reliability.

4. Designing shilling attacks against multi-criteria collaborative filtering

Shilling attack profiles are constructed based on available statistical information, confronted recommendation algorithm, and strategies that are utilized for avoiding detection (Bilge, Ozdemir, & Polat, 2014). However, due to the altered structure of multi-criteria preference collections, traditional shilling attack methodologies cannot be directly applied to MCCF environment. Design decisions of attack profiles targeting MCCF systems would enable adapting to the multi-criteria domain by putting forward strategies for determining attack-specific parameters and methodologies to imitate genuine users in the best way without being detected.

Although MCCF systems achieve promising accuracy values on recommendation results, they introduce intrusiveness as a more challenging problem due to submission of a higher amount of explicit information (Palanivel & Sivakumar, 2010). Therefore, users might leave some sub-criteria blank or rate randomly, especially when the number of criteria is high. In perspective of an attacker, possibly corrupted sub-criteria ratings and non-public information on the collections should be considered. Therefore, the attacker needs to decide how to determine several attack-specific parameters in existence of several criteria. Implementation choices to

Table 3
Research on robustness in recommender systems.

Research field	Research subject	Publications
Attack schemes	Random attack	Burke, Mobasher, and Bhaumik (2005); Burke et al. (2006); Burke, Mobasher et al. (2005), Lam and Riedl (2004), Mobasher, Burke, Bhaumik, and Sandvig (2007), O'Mahony, Hurley, Kushmerick, and Silvestre (2004)
	Bandwagon attack	Burke, Mobasher et al. (2005), Hurley et al. (2007), O'Mahony et al. (2005)
	Reverse bandwagon attack	Mobasher, Burke, Bhaumik, and Sandvig (2007), Zhang (2009)
	Segment attack	Burke, Mobasher, Bhaumik, and Williams (2005), Mobasher, Burke, Bhaumik, and Sandvig (2007)
	Love/hate attack	Mobasher, Burke, Bhaumik, and Sandvig (2007), Zhang (2010)
	Average attack	Burke, Mobasher, and Bhaumik (2005); Burke, Mobasher et al. (2005), Hurley et al. (2007)
	Popular attack	O'Mahony et al. (2003)
	Probe attack	Hurley et al. (2007), Mobasher, Burke, Bhaumik, and Sandvig (2007), Burke, Mobasher et al. (2005), O'Mahony et al. (2005)
	Power user attack	Wilson and Seminario (2013)
	Perfect knowledge attack	Burke, Mobasher et al. (2005)
Robustness analysis	Memory-based algorithms	Resnick and Sami (2008), Hurley et al. (2007) Burke, Mobasher, and Bhaumik (2005), O'Mahony et al. (2004), Long and Hu (2010)
	Model-based algorithms	Cheng and Hurley (2009a, 2009b, 2010a)
Robustness surveys	Recommender systems	Gunes et al. (2014), Aggarwal (2016)
	Collaborative recommendation	Burke et al. (2015)

consider for multi-criteria attack design are two-fold, (*i*) how to adopt attack-specific parameters described in Table 2 into multi-criteria data and (*ii*) how to choose among the criteria to be utilized in attack profile construction. These choices might lead to differing effects on achieved manipulation intensities in predictions and detectability of the attack.

In this section, we first discuss how to obtain attack-specific parameters in the multi-criteria domain, then propose multi-criteria extended shilling attack strategies considering potential parameter extension methodologies along with determining the size of the exploited number of sub-criteria.

4.1. Acquisition of attack-specific parameters

Realizing shilling attack attempts against a recommender system requires the attacker to construct attack profiles based on some statistical and domain-specific information as can be followed in Table 2. Some of this statistical information would be system mean score, system standard deviation, item mean scores, and item mode scores. The attacker can obtain these parameters via several strategies.

Utilizing publicly available data. User feedbacks are often exposed by the recommender service itself to inform their users about offered products better. As a multi-criteria hotel recommender system, HRS.com[1] exhibits general user experience of hotels by displaying detailed ratings and comments of previous users. Fig. 2 presents a sample of general evaluation about a hotel.

Moreover, individual customer experiences might also be public. Fig. 3, for example, presents highly-detailed evaluation feedback of an anonymous user about a hotel. By crawling such public feedback from the websites of the service provider, the attacker obtains necessary statistical information to perform a profile injection attack (Bhaumik, Burke, & Mobasher, 2007).

The shilling attack literature categorizes system mean, system standard deviation, and sets of popular/unpopular items parameters as low-knowledge information due to their public availability in many e-commerce fields (Burke, Mobasher et al., 2005; Lam & Riedl, 2004). Although item mean score and item mode score parameters are considered high-knowledge, they can still be obtained publicly for some domains.

Estimating parameters. If required statistical parameters are not publicly available, they can be easily estimated by observing people using the recommender system and gathering a sample of their votes (Lam & Riedl, 2004), or by employing a domain expert method (Adomavicius & Kwon, 2007). Lam and Riedl (2004) claim that attack-specific parameters can be estimated relatively effortlessly by observation which helps in generating a normal distribution to approximate the observed user behavior in the system. Alternatively, Adomavicius and Kwon (2007) suggest operating domain expertize to specify sub-criteria weights of a multi-criteria recommender system.

Utilizing other systems' data. Another strategy for acquiring statistical parameters is to adopt known values of other systems public data (Lam & Riedl, 2004). For instance, IMDb, MovieLens[2] and Rotten Tomatoes[3] are specialized at collecting and presenting user reviews on movies. TripAdvisor's[4] item collection varies from hotels, restaurants, sights, and activities. There are also many alternatives in tourism domain such as Trivago[5], Booking.com, and Hotels.com. Online retailers, such as Amazon.com and AliExpress[6], accumulates and serves feedback from customers experiences on products from a broad range of diversity. One example of what kind of information that the attacker can retrieve on the movie domain is given in Fig. 4 which demonstrates the rating distribution of an item retrieved from IMDb. As can be followed from Fig. 4, the attacker might acquire crucial statistical information from one or more related service providers, and put such information into account directly for achieving manipulations.

4.2. Extension methodologies for determining attack-specific parameters

The statistical parameters are calculated by utilizing all available ratings in single-criterion systems. However, multi-criteria data collection enables to compute attack-specific parameters in several ways according to acquired knowledge while extending such parameters. We propose the following approaches to estimate these parameters in the multi-criteria domain.

[1] https://www.hrs.com.

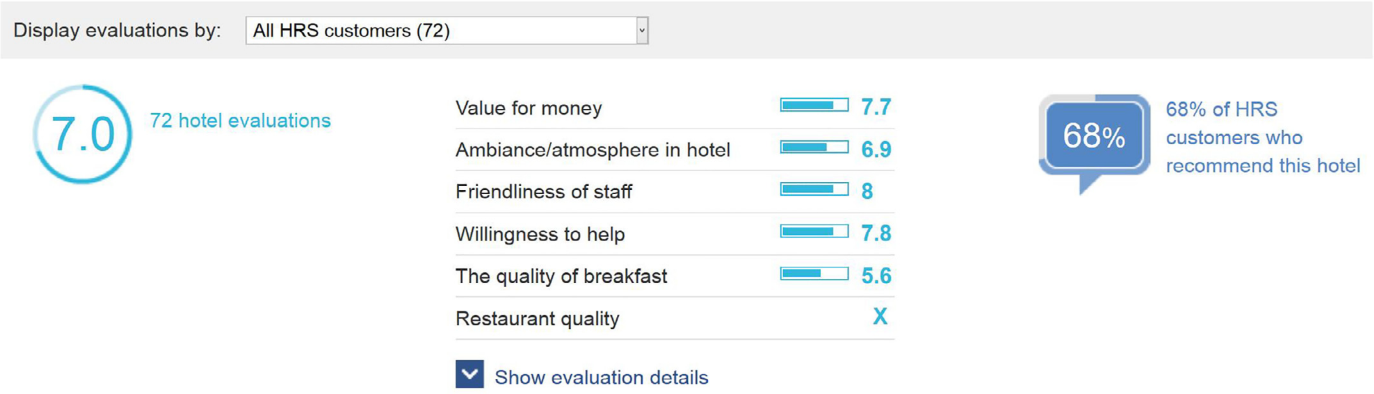
[2] https://movielens.org.

[3] https://www.rottentomatoes.com.

[4] https://www.tripadvisor.com.

[5] https://www.trivago.com.

[6] https://aliexpress.com.



Display evaluations by: All HRS customers (72)

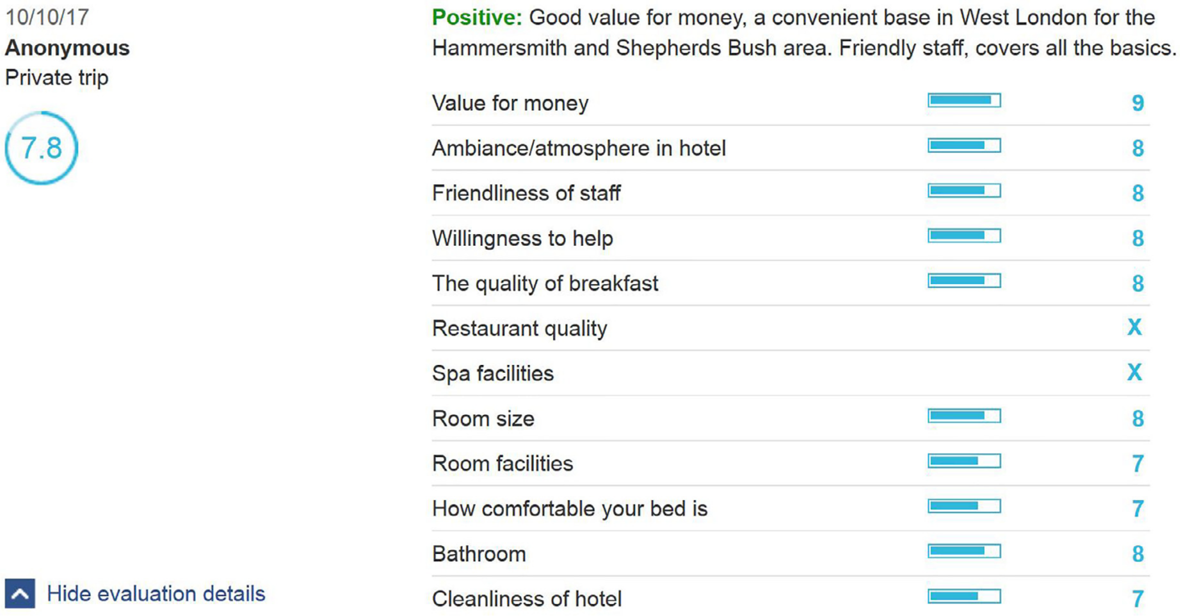
7.0 72 hotel evaluations

Value for money	7.7
Ambiance/atmosphere in hotel	6.9
Friendliness of staff	8
Willingness to help	7.8
The quality of breakfast	5.6
Restaurant quality	X

68% 68% of HRS customers who recommend this hotel

Show evaluation details

Fig. 2. A sample evaluation page for a hotel on HRS.com.



10/10/17
Anonymous
Private trip

7.8

Positive: Good value for money, a convenient base in West London for the Hammersmith and Shepherds Bush area. Friendly staff, covers all the basics.

Value for money	9
Ambiance/atmosphere in hotel	8
Friendliness of staff	8
Willingness to help	8
The quality of breakfast	8
Restaurant quality	X
Spa facilities	X
Room size	8
Room facilities	7
How comfortable your bed is	7
Bathroom	8
Cleanliness of hotel	7

Hide evaluation details

Fig. 3. A sample feedback from a user about a hotel on HRS.com.

Individual criterion extension. Statistical parameters are calculated based on each criterion separately, and these values are used independently in attack profile generation operations. Such approach produces as many parameter values as the number of criteria to be individually used in corresponding part of the attack profile. Although straightforward; this approach is intuitively the best way to utilize all available information with the system in an encapsulated manner as individual values for each sub-criterion have the potential to help fake profiles resemble genuine users. Also, it is more suitable for systems that have much sub-criteria since aggregated results might cause losing the essence of the preference information.

Aggregated criteria extension. Statistical parameters are calculated based on aggregated ratings for all existing criteria, and the obtained value is used in attack profile generation operations. Such approach produces a single aggregated parameter value to be utilized solely in all parts of the attack profile. Intuitively, this approach reduces the likelihood of influencing the recommendation system since the fake profiles are based on an approximation. However, such approximation would be helpful in diminishing intrusiveness effects caused by recklessly rated sub-criteria.

Overall criterion extension. Statistical parameters are calculated solely based on the *Overall* criterion, and such value is utilized in attack profile generation operations. In most cases, the rating for the *Overall* criterion reflects the most dependable information in the multi-criteria domain. Therefore, it would be reasonable to attribute fake profiles on such value. Also, this approach would be useful when the attacker has no access to information about sub-criteria or the systems that are subject to intrusiveness due to having too much sub-criteria.

Besides statistical parameters, determining the set of popular and unpopular items are vital for bandwagon and reverse bandwagon attacks, respectively. Both frequently and highly rated items are considered to be popular items, and exploited by bandwagon attackers as assigning the highest possible rating to such items in order to take part in more genuine users neighborhood. Similarly, frequently and poorly rated items are exploited as assigning the least possible rating for reverse bandwagon attack. However, determining the sets of popular and unpopular items would be achieved

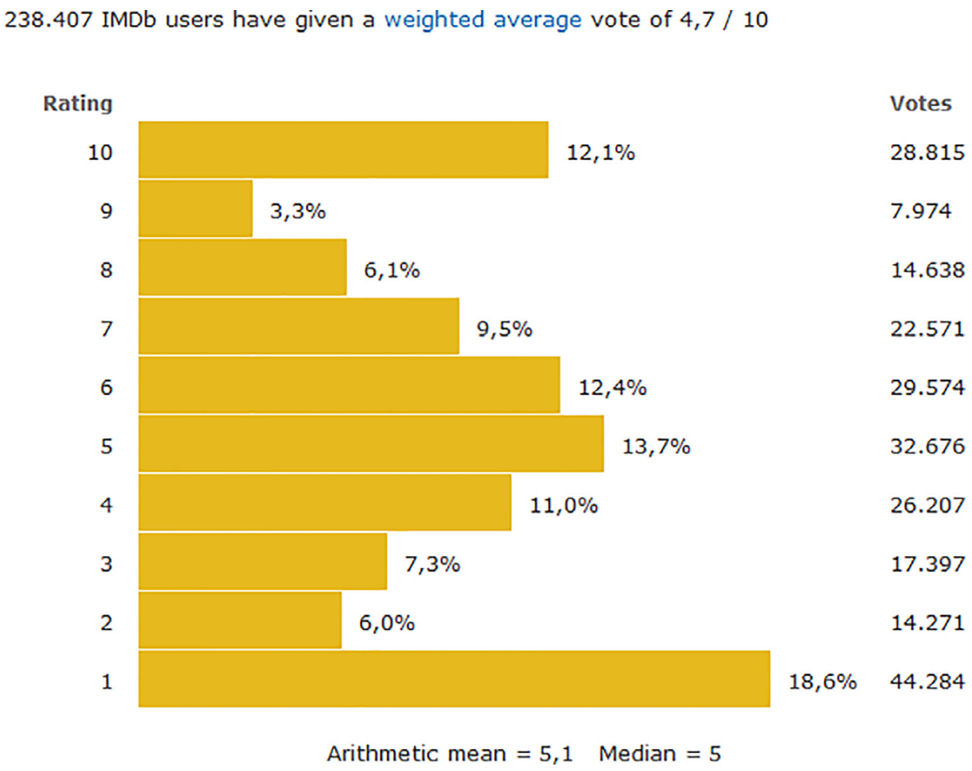


Fig. 4. Rating distribution of a movie publicly shared by IMDb.

differently for MCCF systems. We propose following approaches to estimate these attack-specific parameters in the multi-criteria domain.

Selection over overall criterion. Popularity/unpopularity of an item is determined based on the number of ratings and the mean score solely for the Overall criterion, and top popular/unpopular items are selected to form the corresponding attack. Such approach reflects that the attacker relies on the most dependable criterion for characterizing the attack. Also, this approach would be useful when the attacker has no access to information about sub-criteria or the systems that are subject to intrusiveness due to having too much sub-criteria.

Selection over aggregated criteria. Popularity/unpopularity of an item is determined based on the number of ratings and the mean scores for all available criteria, and top popular/unpopular items are selected to form the corresponding attack. Such approach would be useful in systems with lesser criteria, and it reflects a complete view on the dataset. Although the set of popular/unpopular items will be similar to the one obtained by selecting on *Overall* criterion, there would be slight differences reflecting that an item is exceptionally liked/disliked for some sub-criterion. This approach also would be effective when a particular segment of users is targeted for manipulation such as the people who are keen into visual effects of a movie above all.

Selection over mixed criteria. Auxiliary popularity/unpopularity values of an item are calculated for each individual sub-criterion and the Overall criterion based on the number of ratings and the mean scores for the corresponding criterion. Then, if N globally popular/unpopular items are to be selected, N/k top popular/unpopular items are chosen starting from the Overall criterion and continuing with the most influential sub-criterion on the Overall criterion, where k is the number of criteria. If an item occurs to be previously selected to the global popular/unpopular item set due to a more significant criterion, then the next one in the list is taken by the priority. This way, the set of popular/unpopular items are constructed in a weighted manner by the criterion weights on Overall liking of an item. Also, some of the low-weight sub-criteria might be neglected in this approach to fine tune effects of the selected item list. Note that, such approach requires high-knowledge on the dataset.

4.3. Methodologies to determine exploited criteria size

MCCF systems tend to include many sub-criteria to understand user inclinations better; however, such phenomena might lead users to rate sub-criteria recklessly or even leave them blank. Therefore, there is a trade-off between acquiring better-expressed preferences and satisfying customers. Attackers need to create fake profiles that best imitate real users, so the decision of which criteria to be utilized in attack profile generation is a challenging problem in terms of both effectiveness and detectability of an attack in the multi-criteria domain. Therefore, which sub-criteria to be utilized in attack profile generation is dependent on the number of criteria in the system, the sparsity level of data set, and the preference domain, and it is a substantial question. We propose following approaches to determine the number of sub-criteria to be exploited by the attacker in the multi-criteria domain.

Exploit-all. Such scenario assumes equal weights for each sub-criterion on overall liking degree of an item and strikes to exploit all of them without differentiation. This case would be considered as the straightforward approach to be applied when the attacker does not have substantial information on the system, or particularly the system has a relatively less number of criteria. Also, such approach is expected to be more efficient in systems that are based on similarity aggregation as proposed by Adomavicius and Kwon (2007) where imitating real users in all available criteria is a necessity.

Exploit-overall-only. Such scenario exploits just the Overall criterion where only the *Overall* criterion is filled based on attack-specific parameters and sub-criteria are left blank or filled randomly. Since the Overall criterion is the most significant aspect for evaluating an item and reflects a broader idea of the user opinion, exploiting only the Overall criterion and ignoring sub-criteria would result effective, yet hardly detectable in MCCF systems. Besides, it requires low-level of knowledge about the system which allows attackers to mount effective attacks without comprehensive information.

Exploit-k′. Such scenario exploits a subset of sub-criteria that is determined either randomly or based on their significance on the Overall criterion for all ratings in the system. Such significance of sub-criteria can be estimated using statistical techniques such as least squares regression, machine learning techniques, or domain expertise. In attack profiles, selected sub-criteria can be filled following attack-specific parameters, and the remaining ones are filled randomly or left blank. Such approach would be considered when the number of criteria is high, or especially if confronted recommendation algorithm focuses on the significant criteria in the prediction estimation process. In such case, it would not be wise to attack an insignificant criterion that has little or no effect on the prediction generation process. Intuitively, limiting the size of exploited criteria might disguise attack profile better, hence decreases the likelihood of detection. Furthermore, exploiting randomly selected sub-criteria would contribute more to avoidance of detection since each attack profile would follow a different pattern in choosing sub-criteria to exploit.

4.4. Designing multi-criteria attack profiles

In this section, we describe proposed multi-criteria attacking schemes in detail based on explained parameter determination and criteria exploitation schemes.

4.4.1. Multi-criteria random attack model

Random attacks are based on the idea that a fake profile complying with the distribution of the whole preference collection has the potential to bias the system. Required knowledge to mount random attacks is the system mean and standard deviation values, and uniformly randomly selected filler items are packed based on them. Random attack sets a basis to identify effects of shilling attempts with very less information. The intent of such attack could be both push and nuke regarding the assigned vote to the target item. We characterize proposed multi-criteria random attack model as follows:

1. The set of selected items is null, i.e., $I_S = \emptyset$.
2. The set of filler items, I_F, is constructed by uniformly randomly selected items among the set of all items but the target item, i.e., $I - i_t$, with respect to filler size parameter.
3. The set of sub-criteria to be exploited, C_e, is determined with respect to chosen exploitation scenario.
4. System mean (μ_s) and standard deviation (σ_s) parameter values are determined for all sub-criteria with respect to chosen extension methodology.
5. A set of random numbers, R, is generated from a normal distribution where $R \sim N(\mu_s, \sigma_s^2)$ and these numbers are rounded and packed into the I_F of *Overall* criterion of the fake profile.
6. Step 5 is repeated $|C_e|$ times for all exploited sub-criteria.
7. Finally, the target item, i_t is packed with the highest or lowest possible vote for push and nuke intents, respectively.

4.4.2. Multi-criteria average attack model

Average attacks are designed similarly to random attacks. However, randomly selected filler items are packed with their mean scores. Employing items mean score increases the potential to reside in more users neighborhood. The multi-criteria average attack might be mounted for both push and nuke intents concerning the rating assigned to i_t. We characterize proposed multi-criteria average attack model as follows:

1. The set of selected items is null, i.e., $I_S = \emptyset$.
2. The set of filler items, I_F, is constructed by uniformly randomly selected items among the set of all items but the target item, i.e., $I - i_t$, with respect to filler size parameter.
3. The set of sub-criteria to be exploited, C_e, is determined with respect to chosen exploitation scenario.
4. Item mean (μ_i) and (optionally) standard deviation (σ_i) parameter value(s) are determined for all sub-criteria with respect to chosen extension methodology.
   (a) (optional for avoiding detection) A set of random numbers, R, is generated from a normal distribution where $R \sim N(\mu_i, \sigma_i^2)$ and rounded.
5. Multi-criteria shilling profiles are created by filling I_F of *Overall* criterion by corresponding item mean scores or (optionally) R.
6. Step 5 is repeated $|C_e|$ times for all exploited sub-criteria.
7. Finally, the target item, i_t, is packed with the highest or lowest possible vote for push and nuke intents, respectively.

4.4.3. Multi-criteria bandwagon attack model

Bandwagon attacks are designed to exploit interest of users towards popular items. Employing these items in an attack profile increases the likelihood of appearance in more genuine users neighborhood. Additionally, randomly selected filler items are packed with system mean to help to avoid detection. Such attacks are used to push a target items predictions. Such attacks are mounted to push predictions for i_t. We characterize proposed multi-criteria bandwagon attack model as follows:

1. The set of selected items is determined with respect to chosen extension methodology, i.e., $I_S = \{p_1, p_2, \ldots, p_s\}$ where s is the size of popular products to be selected and $p_i, i = 1, \ldots, s$ is a popular item.
2. The set of filler items, I_F, is constructed by uniformly randomly selected items among the set of all items but the target item and I_S, i.e., $(I - \{i_t \cup I_S\})$, with respect to filler size parameter.
3. The set of sub-criteria to be exploited, C_e, is determined with respect to chosen exploitation scenario.
4. System mean (μ_s) and standard deviation (σ_s) parameter values are determined for all sub-criteria with respect to chosen extension methodology.
5. A set of random numbers, R, is generated from a normal distribution where $R \sim N(\mu_s, \sigma_s^2)$ and these numbers are rounded and packed into the I_F of *Overall* criterion of the fake profile.
6. I_S of *Overall* criterion of the fake profile is packed with the highest possible vote.
7. Step 5 and 6 are repeated $|C_e|$ times for all exploited sub-criteria.
8. Finally, the target item, i_t is packed with the highest possible vote for push intent.

4.4.4. Multi-criteria reverse bandwagon attack model

Similar to bandwagon, reverse bandwagon attack employs unpopular items to perform a nuke attack. We characterize proposed multi-criteria reverse bandwagon attack model as follows:

1. The set of selected items is determined with respect to chosen extension methodology, i.e., $I_S = \{p_1, p_2, \ldots, p_s\}$ where s is the size of unpopular products to be selected and $p_i, i = 1, \ldots, s$ is an unpopular item.
2. The set of filler items, I_F, is constructed by uniformly randomly selected items among the set of all items but the target item and I_S, i.e., $(I - \{i_t \cup I_S\})$, with respect to filler size parameter.
3. The set of sub-criteria to be exploited, C_e, is determined with respect to chosen exploitation scenario.

4. System mean (μ_s) and standard deviation (σ_s) parameter values are determined for all sub-criteria with respect to chosen extension methodology.
5. A set of random numbers, R, is generated from a normal distribution where $R \sim N(\mu_s, \sigma_s^2)$ and these numbers are rounded and packed into the I_F of *Overall* criterion of the fake profile.
6. I_S of *Overall* criterion of the fake profile is packed with the lowest possible vote.
7. Step 5 and 6 are repeated $|C_e|$ times for all exploited sub-criteria.
8. Finally, the target item, i_t, is packed with the lowest possible vote for nuke intent.

4.4.5. Multi-criteria love/hate attack model

Love/hate attacks are designed as assigning the highest possible rating to randomly chosen filler items and the lowest vote for the target item. Such attacking scheme is a simple yet very effective nuke attack. Love/hate attack does not employ any statistical parameters, therefore, extending it to the multi-criteria domain is straightforward except for criteria exploiting method. We characterize proposed multi-criteria love/hate attack model as follows:

1. The set of selected items is null, i.e., $I_S = \emptyset$.
2. The set of filler items, I_F, is constructed by uniformly randomly selected items among the set of all items but the target item, i.e., $I - i_t$, with respect to filler size parameter.
3. The set of sub-criteria to be exploited, C_e, is determined with respect to chosen exploitation scenario.
4. Multi-criteria shilling profiles are created by filling I_F of *Overall* criterion by corresponding highest possible vote.
5. Step 4 is repeated $|C_e|$ times for all exploited sub-criteria.
6. Finally, the target item, i_t, is packed with the highest or lowest possible vote for push and nuke intents, respectively.

4.4.6. A novel multi-criteria attack model – mode attack

Researchers (Burke, Mobasher et al., 2005; Mobasher, Burke, Bhaumik, & Sandvig, 2007) state that average attack is a high-knowledge yet very successful in biasing recommender systems due to establishing high similarity bonds based on each specific item. However, such similarity is expected to be high since the fake profiles are packed with an average vote for these items. Because some items have both lovers and haters, such approach might fail in datasets with a wide rating range and for items that have a varying range of votes on both sides of the scale. In such cases, averaging these votes would represent neither side of the distribution, but a middle range between them, which results in the low similarity between attack profiles and genuine profiles reducing the effect of the atack.

Instead of assigning an average vote for the filler items, we propose inserting the most repeated vote, i.e., mode of available ratings to filler items. Since the average attack also packs attack profiles with the rounded value of the mean of given votes for the corresponding item, such approach is expected to be more robust against extreme rating schemes residing at the boundaries, as well as regular items. Effects of creating fake profiles with such scheme are expected to be superior in the multi-criteria domain since neighborhood formation is based on several criteria instead of a single one.

We characterize proposed multi-criteria mode attack model as follows:

1. The set of selected items is null, i.e., $I_S = \emptyset$.
2. The set of filler items, I_F, is constructed by uniformly randomly selected items among the set of all items but the target item, i.e., $I - i_t$, with respect to filler size parameter.
3. The set of sub-criteria to be exploited, C_e, is determined with respect to chosen exploitation scenario.
4. Item mode ($Mode_i$) parameter value is determined for all sub-criteria with respect to chosen extension methodology.
5. Multi-criteria shilling profiles are created by filling I_F of *Overall* criterion by corresponding $Mode_i$ value.
6. Step 5 is repeated $|C_e|$ times for all exploited sub-criteria.
7. Finally, the target item, i_t, is packed with the highest or lowest possible vote for push and nuke intents, respectively.

5. Experimental evaluation

In order to analyze the robustness of popular multi-criteria recommendation algorithms and effectiveness of the proposed attacks, we have conducted several experiments on real-world datasets.

5.1. Dataset and evaluation criteria

We used two versions of a multi-criteria preference dataset crawled by Jannach et al. (2012) from Yahoo!Movies platform. In the dataset, users have multi-criteria ratings on four sub-aspects of the movie domain, i.e., acting, directing, visuals, and story, along with an overall rating. The original ratings are in a 13-level scale (A+ to F), and we converted lexical representations into numerical ratings where [A+, A, A−] corresponds to [13, 12, 11]-stars, respectively and F to 1-star. We utilize two subsets of the collection, named YM5 and YM20 datasets. The YM5 dataset consists of 63,027 multi-criteria ratings from 4377 users on 2565 movies where each user and movie have least five ratings. Likewise, the YM20 dataset contains 8157 multi-criteria ratings from 202 users on 247 movies where each user and movie have least 20 ratings. All multi-criteria ratings in the sets comprise of the *Overall* preference vote and sub-criteria votes. Due to the high average of collected ratings within the datasets (9.88/13 for YM5 and 9.96/13 for YM20), it is more challenging to mount a push attack rather than a nuke attack. Therefore, we investigate the robustness of the algorithms primarily against nuke attacks. The effects of profile injection attacks are observed by measuring the shift of prediction scores after applying the attack for the targeted items (Burke, Mobasher, & Bhaumik, 2005).

5.2. Experimentation methodology

We followed an all-but-one experimentation methodology where for each iteration one of the users was evaluated as the test user, and the rest of the users construct the train set. Attacking scenarios for each attack type was implemented individually for each item in the pre-selected target item sets with varying filler size parameter and resulting prediction shift values were measured. Attack size parameter was kept constant at 0.5% and 1% for the YM5 dataset which implies injection of 20 and 40 fake profiles, respectively. Due to the small number of users in the YM20 dataset, we evaluated attack size parameters of 5% and 10% which implies injection of 10 and 20 shilling profiles, respectively.

Since it is not practical to push prediction results of an item that already has high scores or to nuke a poorly rated item, two small subsets of 10 target items resembling the overall characteristics of the dataset were selected based on ratings count and averages of items. Table 4 demonstrates statistics of selected target item sets for push and nuke attack evaluations, where pushed items set consists of the items with rating averages from 3 to 9, nuked items set from 10 to 13, and numbers indicating how many of the selected items fall into the corresponding group.

5.3. Experimental results

We demonstrate results of several experimental assessments to investigate various aspects of realizing multi-criteria profile

Table 4
Statistics of targeted items.

Data set	Ratings count	Pushed items			Nuked items		
		[3–7)	[7–8)	[8–9]	[10–11)	[11–12)	[12–13]
YM20	20–30	3	2	1	2	1	1
	31–40	1	1	–	1	2	–
	41–65	1	1	–	1	1	–
	66 and up	–	–	–	–	1	–
YM5	5–10	2	2	2	2	2	1
	10–20	1	1	–	1	1	–
	20–40	1	1	–	1	1	–
	40 and up	–	–	–	1	–	–

Table 5
Effects of extension methodologies for random, average, and mode attacks (YM5).

Attack type	Attack size	Individual criterion extended			Aggregated criteria extended			Overall criterion extended		
		AvgSim	Manhattan	AggFcn	AvgSim	Manhattan	AggFcn	AvgSim	Manhattan	AggFcn
Random$_P$	0.5%	0.11	1.17	1.42	0.12	1.39	1.47	0.43	1.00	1.49
Random$_N$		−0.28	−2.08	−2.69	−0.71	−2.66	−2.80	−0.36	−2.26	**−2.99**
Average$_P$		1.94	1.24	1.27	1.60	1.33	1.63	1.63	1.37	1.44
Average$_N$		**−3.52**	−2.80	**−3.58**	**−3.59**	−2.66	−2.58	**−3.65**	−2.86	**−2.99**
Mode$_P$		**2.18**	1.71	1.64	**2.20**	1.55	1.45	**2.29**	1.63	1.57
Mode$_N$		**−3.34**	**−3.33**	−2.71	**−3.04**	**−3.15**	−2.66	**−3.45**	−3.05	**−2.99**
Random$_P$	1%	0.32	1.55	**2.37**	0.24	1.83	**2.11**	0.46	1.06	**2.14**
Random$_N$		−0.65	−2.56	**−3.04**	−0.76	−2.90	**−3.01**	−0.73	−2.50	**−3.06**
Average$_P$		**2.73**	**2.06**	**2.28**	**2.62**	**2.10**	**2.19**	**2.36**	1.90	1.95
Average$_N$		**−4.61**	**−3.56**	**−3.97**	**−4.53**	**−3.28**	**−3.56**	**−4.54**	**−3.81**	**−3.48**
Mode$_P$		**2.88**	**2.33**	1.93	**2.87**	**2.13**	**2.01**	**3.01**	**2.14**	**2.12**
Mode$_N$		**−4.51**	**−4.39**	**−3.62**	**−4.33**	**−4.19**	**−3.62**	**−4.46**	**−4.07**	**−3.62**

Table 6
Effects of extension methodologies for random, average, and mode attacks (YM20).

Attack type	Attack size	Individual criterion extended			Aggregated criteria extended			Overall criterion extended		
		AvgSim	Manhattan	AggFcn	AvgSim	Manhattan	AggFcn	AvgSim	Manhattan	AggFcn
Random$_P$	5%	0.34	1.48	1.96	0.31	1.83	**2.13**	0.38	1.24	**2.08**
Random$_N$		−0.85	**−3.68**	**−4.38**	−0.99	**−4.19**	**−4.38**	−0.93	**−3.60**	**−4.38**
Average$_P$		**2.91**	**2.32**	**2.88**	**2.97**	**2.23**	1.99	**3.19**	**2.26**	**2.18**
Average$_N$		**−5.88**	**−4.47**	**−4.61**	**−4.37**	**−5.75**	**−3.92**	**−5.72**	**−4.92**	**−4.47**
Mode$_P$		**3.01**	**2.46**	**2.18**	**2.95**	**2.00**	1.70	**2.84**	**2.18**	**2.25**
Mode$_N$		**−6.78**	**−6.35**	**−5.04**	**−6.41**	**−5.98**	**−5.21**	**−6.57**	**−5.82**	**−5.45**
Random$_P$	10%	0.49	1.74	**2.45**	0.43	**2.03**	**2.45**	0.47	1.38	**2.45**
Random$_N$		−0.99	**−3.92**	**−4.70**	−0.99	**−4.19**	**−4.70**	−0.93	**−3.60**	**−4.70**
Average$_P$		**3.86**	**3.02**	**2.92**	**3.80**	**2.98**	**2.92**	**3.79**	**2.96**	**2.92**
Average$_N$		**−7.30**	**−5.87**	**−5.65**	**−7.27**	**−5.75**	**−5.65**	**−7.27**	**−5.87**	**−5.65**
Mode$_P$		**3.77**	**3.05**	**2.70**	**3.70**	**2.81**	**2.70**	**3.70**	**2.99**	**2.70**
Mode$_N$		**−7.47**	**−7.26**	**−5.99**	**−7.17**	**6.94**	**−5.99**	**−7.39**	**6.74**	**−5.99**

injections. Presented results are the average of 100 experimental trials due to randomization procedure in selecting and assigning ratings to filler items within attack profiles.

5.3.1. *Effects of extension methodologies*

In this section, we compare proposed extension methodologies for multi-criteria attack profile generation procedure for varying MCCF algorithms. We investigate effects of interpreting individual criteria ratings for internal attack parameter calculation such as system mean value for the random attack, item mean/mode values for the average and mode attacks, and popular/unpopular item sets for the bandwagon and reverse bandwagon attacks. These attack parameters are calculated based on proposed three extension methods, i.e., (*i*) individual criterion extended, (*ii*) aggregated criteria extended, and (*iii*) *Overall* criterion extended for random and average attacks, and selecting popular/unpopular items based over (*i*) *Overall* criterion, (*ii*) aggregated criteria, and (*iii*) mixed criteria. We employed average similarity-based (AvgSim), Manhattan distance-based, and aggregation function-based (AggFcn) MCCF algorithms with fixed filler size values of 7% and 10% for YM5 and YM20 datasets, respectively. Tables 5 and 6 summarize prediction shift values for random and average attacks where the extension methodologies are employed for estimating system- and item-mean values, respectively. Both push and nuke intents are applied and represented as subscripts of the attack names.

For clarity, in Tables 5 and 6, we highlighted prediction shift values greater than 2 for push attacks and greater than 3 for nuke attacks to emphasize significance of the attacks. As can be followed in Table 5, none of the algorithms seem resilient to any manipulation attack where a prediction shift value of greater than one can be achieved in almost any condition. Also, attack size has a linear correlation with the success of the attack as the more fake profiles in the database, the more the possibility of attack leaking into the neighborhoods of genuine users. The random attack has not very significant effects for both push and nuke intents except for rare conditions. The average attack, on the other hand, achieves effective prediction shift values, especially for nuke intent where shifting the resulting prediction category is almost always possible. The mode attack performs similar to the average attack, except it

Table 7
Effects of extension methodologies for random, average, and mode attacks (YM5).

Attack type	Attack size	Selection over overall criterion			Selection over aggregated criteria			Selection over mixed criteria		
		AvgSim	Manhattan	AggFcn	AvgSim	Manhattan	AggFcn	AvgSim	Manhattan	AggFcn
BW	0.5%	1.40	1.10	1.26	0.99	0.57	1.61	0.79	0.29	0.68
RBW		−2.23	−1.81	−2.54	−2.17	−1.96	−2.43	−1.83	−1.78	−2.38
BW	1%	1.64	1.33	1.74	1.30	1.00	1.78	1.06	0.60	1.09
RBW		−2.90	−2.55	**−3.30**	−2.73	−2.42	**−3.05**	−2.28	−2.11	**−2.99**

Table 8
Effects of extension methodologies for random, average, and mode attacks (YM20).

Attack type	Attack size	Selection over overall criterion			Selection over aggregated criteria			Selection over mixed criteria		
		AvgSim	Manhattan	AggFcn	AvgSim	Manhattan	AggFcn	AvgSim	Manhattan	AggFcn
BW	5%	1.67	1.27	**2.01**	1.4	0.99	1.78	1.08	0.4	1.21
RBW		**−3.34**	−2.96	**−3.93**	**−3.44**	−2.91	**−3.64**	−2.51	−2.5	**−3.54**
BW	10%	**2.19**	1.8	**2.47**	1.82	1.52	**2.33**	1.5	0.95	1.62
RBW		**−3.87**	**−3.44**	**−4.37**	**−3.79**	**−3.36**	**−4.08**	−2.99	−2.88	**−3.97**

achieves higher prediction shift values in the distance-based algorithm.

Similarly for the YM20 dataset, as can be followed in Table 6, almost all attack types achieve a significant amount of prediction shift for all extension methodologies except random attacks for AvgSim algorithm. The reason for the attacks to be more effective in YM20 dataset than YM5 dataset is the difference between their sparsity levels. It is reasonable to expect that shilling profiles have a more significant effect as the dataset becomes denser due to the increase in the likelihood of appearing in genuine users neighborhood. Although individual criterion extension method utilizes more information on preferences than other approaches, there is not a substantial difference in achieved manipulation levels which conclude that separately exploiting available criteria would render futile. In addition, utilizing a lesser number of criteria helps the attacker to disguise themselves better. It can be concluded that even though the attacker has low-level knowledge about the collection, they still can design an effective attack. Average and mode attacks achieve more than half the range shift values for nuke attacks and one-third of the range for push attacks.

Also, Tables 7 and 8 demonstrate prediction shift values for bandwagon and reverse bandwagon attacks where the extension methodologies are employed for calculating the sets of popular and unpopular items, respectively.

As understood from Tables 7 and 8, *RBW* attack is more effective than *BW* attack due to the characteristics of the dataset. Due to high averages of ratings in both YM5 and YM20 datasets, it is challenging to determine which movies are popular exactly. For both datasets, selecting the set of popular/unpopular items based on *Overall* criterion has superior effects compared to selection over aggregated and mixed criteria which indicates that users are rating *Overall* criterion more carefully. AggFcn is the most vulnerable MCCF algorithm to such kind of attacks for which more than one-third of the range prediction shift values are obtained in the YM20 dataset. Although the most robust algorithm is distance-based MCCF, it is still exposed to significant amount of prediction shift values even with the lowest configurations in the YM5 dataset.

5.3.2. Effects of exploited criteria size

In this section, we evaluate effects of generating multi-criteria attack profiles by employing scenarios that exploit varying number of criteria. For the sake of clarity, we only demonstrate results for individual criterion extended average nuke attack with varying criteria exploitation scenarios and MCCF algorithms. Experimented scenarios can be explained as follows:

Serendipitous. This scenario measures serendipity effects by merely creating multi-criteria attack profiles having random ratings for randomly chosen filler items.

Exploit-all. This scenario assumes equal weights for each sub-criterion and strikes to exploit all criteria. Such scenario can be considered as the straightforward approach.

Overall-only. This scenario exploits only the *Overall* criterion where only the *Overall* criterion is filled with item's mean score, and sub-criteria are filled with randomly generated ratings.

Exploit-k. This scenario exploits *k* criteria among all. Such *k* sub-criteria can either be chosen randomly (Exploit-random-*k*) or selected among the most significant criteria (Exploit-significant-*k*) based on linear least squares regression coefficients. Selected criteria are filled with item mean scores, and the remaining criteria are filled with randomly generated ratings.

Tables 9 and 10 present prediction shift values for varying criteria exploitation scenarios and filler sizes for MCCF algorithms. We employ filler size values of 1–10% for YM5 dataset and 3–25% for YM20 dataset.

According to presented results in Tables 10 and 9, all proposed exploitation methods achieve a significantly more prediction shift value than the serendipitous approach which concludes that MCCF systems are highly vulnerable to profile injection attacks. For YM5 dataset, significant manipulations are obtained at more than 7% filler size parallel to the density of the dataset. For YM20 dataset, even exploiting only one criterion, i.e., *Overall* criterion produces about four times more prediction shift compared to the serendipitous approach. On the other hand, exploiting more criteria results in higher prediction shift values as can be anticipated, and Exploit-all scenario achieves the highest prediction shift values, as about half the range. Although exploiting more significant sub-criterion has a positive effect, exploiting more sub-criteria has a superior effect as Exploit-rand-4 achieves higher prediction shift than Exploit-sign-3 for both datasets. AggFcn seems like the most vulnerable algorithm among all since its recommendation approach is directly based on auxiliary predictions to sub-criteria. Therefore, the more the number of exploited sub-criteria, the more the levels of manipulations in this algorithm. Finally, based on the characteristics of the dataset, 10% filler size achieves an optimal result for manipulation levels in both datasets.

5.3.3. Comparison of multi-criteria attack types

In this section, we provide a comprehensive evaluation regarding each attack type for all variations of MCCF algorithms. We uti-

Table 9
Prediction shift values for varying scenarios (YM5).

Scenario	Attack size	AvgSim					Manhattan					AggFcn				
		1%	3%	5%	7%	10%	1%	3%	5%	7%	10%	1%	3%	5%	7%	10%
Serendipitous	0.5%	−0.01	−0.06	−0.25	−0.26	−0.30	0.00	−0.04	−0.19	−0.18	−0.21	−0.01	−0.01	−0.35	−0.90	−1.06
Exploit-all		−0.15	−1.11	−1.38	**−3.52**	**−4.74**	−0.12	−0.80	−0.97	**−2.80**	**−3.38**	−1.34	−1.69	−1.91	**−3.58**	**−5.38**
Overall-only		−0.25	−0.23	−0.20	−0.65	−0.98	−0.19	−0.17	−0.13	−0.47	−0.70	−0.45	−0.45	−0.75	−1.34	−1.45
Exploit-sign-2		−0.07	−0.09	−0.55	−1.50	**−2.34**	−0.07	−0.07	−0.39	−1.07	−1.67	−0.21	−0.45	−0.70	−1.37	−1.68
Exploit-sign-3		−0.08	−0.62	−0.74	**−1.92**	**−2.69**	−0.07	−0.44	−0.52	−1.37	**−1.92**	−0.29	−0.62	−0.95	−1.87	**−2.29**
Exploit-rand-3		−0.07	−0.27	−0.24	−1.85	**−2.87**	−0.06	−0.20	−0.17	−1.32	**−2.04**	−0.21	−0.77	−0.87	**−2.09**	**−2.68**
Exploit-rand-4		−0.13	−1.00	−1.20	**−3.03**	**−4.36**	−0.10	−0.71	−0.85	**−2.17**	**−3.10**	−0.16	−1.35	**−1.69**	**−3.03**	**−3.65**
Serendipitous	1%	−0.01	−0.01	−0.18	−0.47	−0.55	−0.03	−0.02	−0.10	−0.40	−0.46	−0.01	−0.01	−0.42	−1.09	−1.29
Exploit-all		−0.22	−1.53	**−1.96**	**−4.61**	**−6.40**	−0.19	−1.20	−1.53	**−3.56**	**−4.93**	−1.63	**−2.05**	**−2.32**	**−3.97**	**−6.54**
Overall-only		−0.35	−0.37	−0.41	−0.99	−1.45	−0.29	−0.31	−0.35	−0.80	−1.15	−0.55	−0.55	−0.91	−1.63	**−1.76**
Exploit-sign-2		−0.12	−0.19	−0.88	**−2.11**	**−3.24**	−0.12	−0.17	−0.71	−1.65	**−2.52**	−0.26	−0.55	−0.85	−1.67	**−2.04**
Exploit-sign-3		−0.13	−0.88	−1.13	**−2.66**	**−3.70**	−0.13	−0.70	−0.90	**−2.07**	**−2.87**	−0.35	−0.75	−1.16	**−2.27**	**−2.78**
Exploit-rand-3		−0.11	−0.42	−0.47	**−2.56**	**−3.93**	−0.11	−0.35	−0.40	**−2.00**	**−3.04**	−0.26	−0.94	−1.06	**−2.54**	**−3.26**
Exploit-rand-4		−0.19	−1.38	−1.73	**−4.12**	**−5.89**	−0.17	−1.08	−1.36	**−3.19**	**−4.54**	−0.19	−1.64	**−2.06**	**−3.68**	**−4.44**

Table 10
Prediction shift values for varying scenarios (YM20).

Scenario	Attack size	AvgSim					Manhattan					AggFcn				
		3%	5%	10%	15%	25%	3%	5%	10%	15%	25%	3%	5%	10%	15%	25%
Serendipitous	5%	−0.33	−0.35	−0.36	−0.23	−0.08	−0.08	−0.04	−0.02	−0.07	−0.08	**−2.32**	**−2.70**	**−2.29**	−1.45	−0.37
Exploit-all		**−4.68**	**−5.57**	**−5.88**	**−5.55**	**−4.45**	**−5.74**	**−5.26**	**−5.59**	**−4.35**	**−2.95**	**−3.79**	**−4.53**	**−4.61**	**−3.71**	−1.71
Overall-only		−1.29	−1.39	−1.34	−1.14	−0.93	−0.24	−0.14	−0.03	−0.06	−0.08	**−2.20**	**−2.54**	**−2.24**	−1.46	−0.40
Exploit-sign-2		**−2.60**	**−2.76**	**−2.75**	**−2.12**	−1.30	**−4.54**	**−4.59**	**−3.58**	**−2.35**	−1.01	**−3.75**	**−4.38**	**−4.37**	**−3.49**	−1.56
Exploit-sign-3		**−2.66**	**−2.97**	**−2.89**	**−2.22**	−1.39	**−3.44**	**−3.28**	**−2.14**	−1.11	−0.26	−3.62	−4.22	−4.09	−3.13	−1.31
Exploit-rand-3		**−2.58**	**−2.54**	**−2.46**	−1.74	−0.90	−1.47	−1.19	−0.55	−0.15	−0.05	**−3.28**	**−3.77**	**−3.74**	**−2.79**	−1.17
Exploit-rand-4		**−3.72**	**−4.53**	**−4.57**	**−3.79**	**−2.30**	**−3.29**	**−3.05**	−1.87	−0.88	−0.12	**−3.50**	**−4.13**	**−4.15**	**−3.27**	−1,44
Serendipitous	10%	−0.41	−0.44	−0.45	−0.29	−0.10	−0.09	−0.06	−0.03	−0.01	−0.001	−2.84	**−3.31**	−2.81	−1.78	−0.45
Exploit−all		**−5.81**	**−6.92**	**−7.30**	**−6.89**	**−5.52**	**−5.75**	**−5.28**	**−5.64**	**−4.43**	**−3.03**	**−4.65**	**−5.55**	**−5.65**	**−4.55**	−2.09
Overall−only		−1.60	−1.73	−1.66	−1.41	−1.16	−0.25	−0.16	−0.08	−0.02	−0.002	−2.70	**−3.11**	−2.75	−1.79	−0.49
Exploit-sign−2		−3.23	−3.80	−3.42	−2.63	−1.62	−4.55	**−4.61**	**−3.63**	−2.43	−1.09	**−4.60**	**−5.37**	**−5.36**	**−4.28**	−1.91
Exploit-sign−3		**−3.30**	**−3.87**	**−3.59**	−2.75	−1.72	**−3.45**	**−3.30**	−2.19	−1.19	−0.34	**−4.44**	**−5.17**	**−5.01**	**−3.84**	−1.61
Exploit-rand−3		**−3.20**	**−3.15**	**−3.06**	−2.16	−1.12	−1.48	−1.21	−0.60	−0.23	−0.03	**−4.02**	**−4.62**	**−4.58**	**−3.42**	−1.44
Exploit-rand−4		**−4.62**	**−5.63**	**−5.67**	**−4.70**	−2.85	**−3.30**	**−3.07**	−1.92	−0.96	−0.20	**−4.29**	**−5.06**	**−5.09**	**−4.01**	−1.77

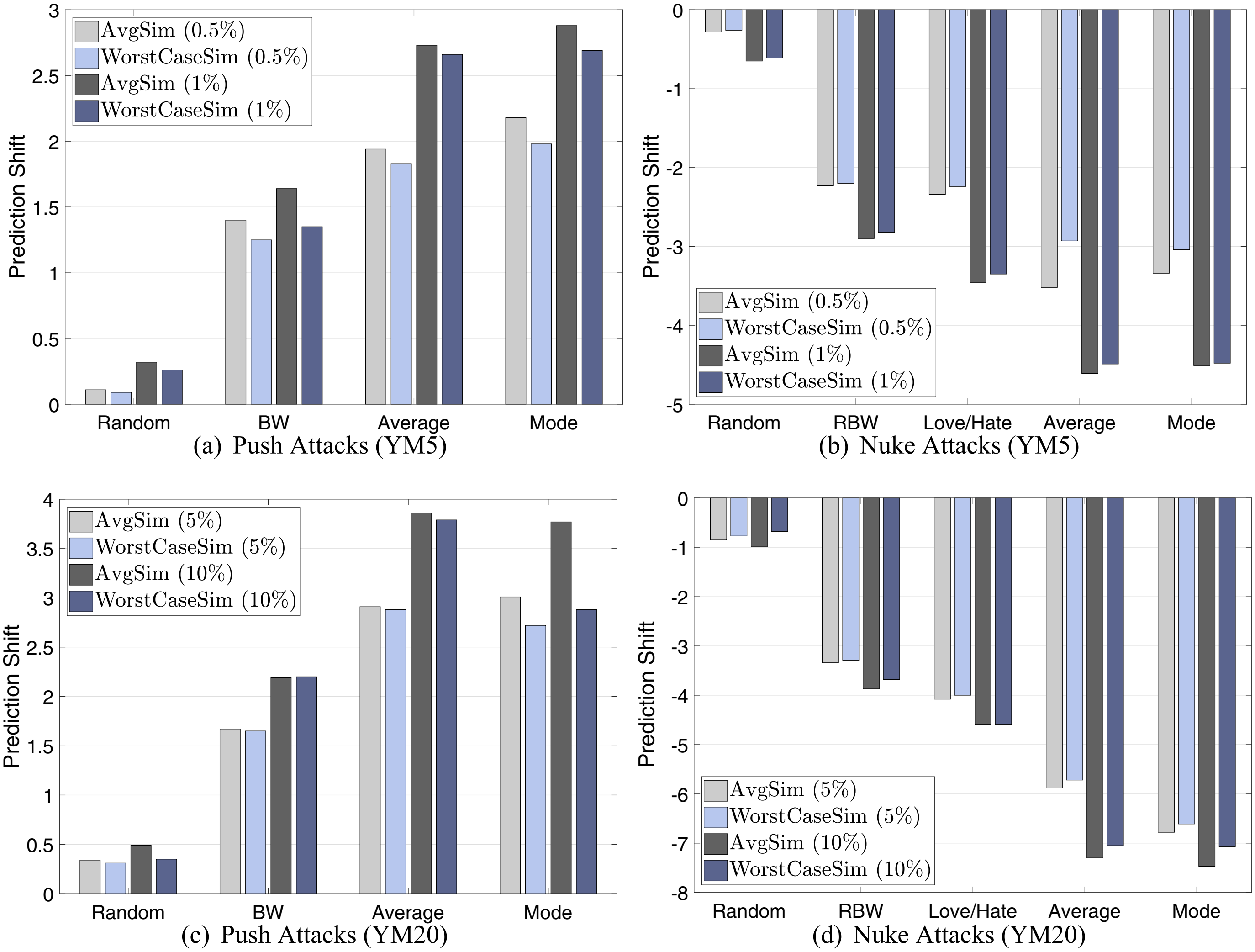
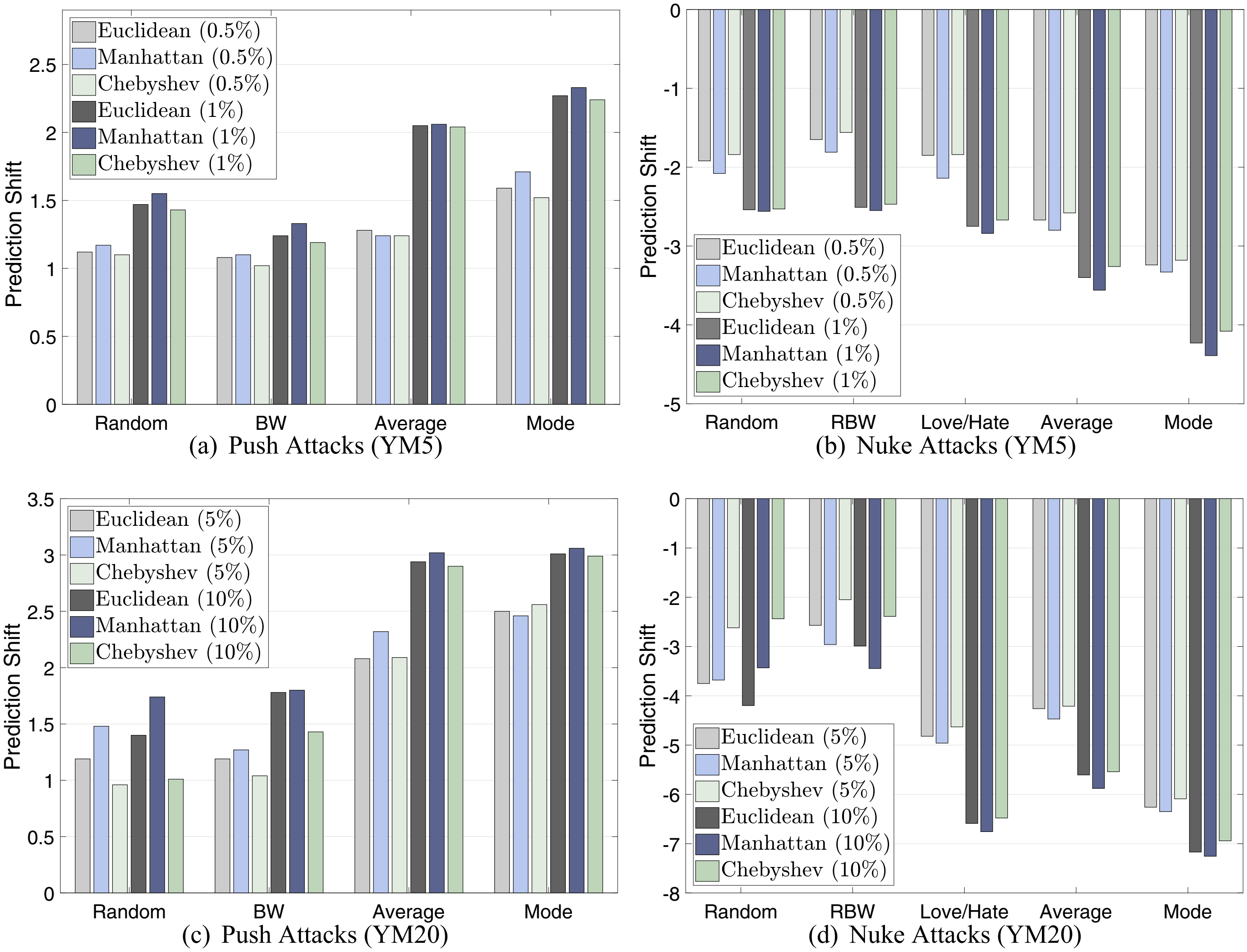


Fig. 5. Comparison of multi-criteria attack types on similarity-based MCCF algorithms.

lize individual criterion extension method along with Exploit-all strategy for fixed filler size values of 7% and 10% for YM5 and YM20 datasets, respectively. Proposed extended attack types are compared based on (*i*) similarity-based, (*ii*) distance-based, and (*iii*) aggregation function-based MCCF algorithms. Fig. 5 presents prediction shift values for varying similarity merging methods with two different attack size configurations depicted in parentheses.

Between two similarity merging-based MCCF algorithms in Fig. 5, the WorstCaseSim-based approach is more robust than AvgSim-based approach against all attacks. The inherent protection mechanism against attacks causes such robustness by accepting the minimum similarity value as ultimate. The random attack is the less effective than other attack types achieving prediction shift values less than 1. Although simple to mount, love/hate attack is more effective than Random and *RBW* attacks. Among all attacking schemes, average and mode attacks appear as the most influential attacks where they can manipulate prediction results by significant amounts that would change victims mind. However, it can be concluded that all attack scenarios achieve a significant amount of prediction shift values except for random attacks. Although attack size has a positive effect on the success of the attacks, obtained augmentations on the prediction shift values do not linearly increase with the amount of injected profiles, which concludes that the algorithms are highly vulnerable to manipulations regardless of the magnitude of the attack. Sparsity levels of datasets act as another parameter on the success of the attacks where the similarity-based MCCF algorithms are weaker in denser datasets due to their neighborhood formation logic. In a nutshell, obtained results demonstrate that similarity merging-based MCCF algorithms are vulnerable to profile injection attacks.

Fig. 6 presents prediction shift values for varying distance-based MCCF algorithms. Although they perform similarly, with slight differences Chebyshev distance-based MCCF algorithm is the most robust and Manhattan distance-based is the weakest among three distance-based MCCF algorithms in Fig. 6. While average and mode attacks perform similarly and achieve the highest prediction shift values for push attacks, mode attack attains even higher shift values for nuke attacks. Love/hate attack, on the other hand, is more effective than the average nuke attack. Such result arises due to the very high average of ratings in the dataset which aligns with the intent of love/hate attack. Random attacks are also as effective as *BW* and *RBW* attacks due to their profile configuration based on random numbers drawn from some distribution which has a positive effect on inverse distance-based neighborhood formation. Size of mounted attacks has a merely positive effect on obtained prediction shifts not compliant with its growth, and especially *BW* and *RBW* attacks achieve similar manipulations even with the size of injected profiles doubled due to the difficulty of locating popular/unpopular items in the datasets. As a concluding remark, all attack scenarios cause a considerably high amount of prediction



(a) Push Attacks (YM5)

(b) Nuke Attacks (YM5)

(c) Push Attacks (YM20)

(d) Nuke Attacks (YM20)

Fig. 6. Comparison of multi-criteria attack types on distance-based MCCF algorithms.

shift value which demonstrates the vulnerability of distance-based MCCF algorithms against profile injection attacks, too.

Fig. 7 presents prediction shift values for aggregation function-based MCCF algorithm.

As can be concluded by Fig. 7, aggregation function-based MCCF algorithm is extremely defenseless against all attack types for both push and nuke intents. The reason behind such vulnerability is that in aggregation function-based approach all components of resulting prediction can be affected by the attack. Among all, average and mode attacks are the most effective ones regardless of the size of attacks. However, distributions of item ratings in the datasets determine which one is better suitable to be mounted.

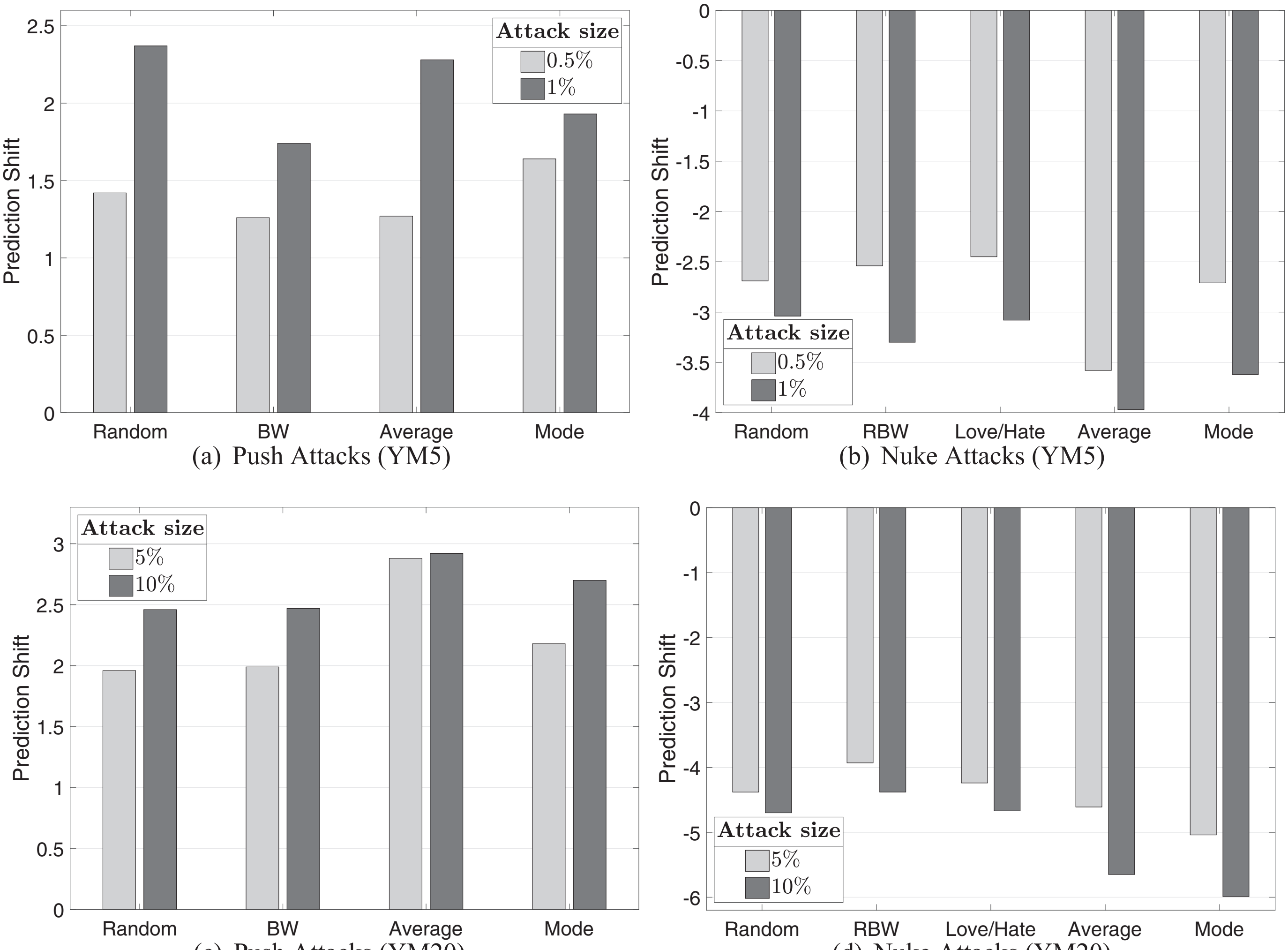
Regarding the level of achieved manipulations in all three MCCF algorithms, it can be concluded that the multi-criteria domain is extremely susceptible against shilling attempts. As a future research direction, mechanisms to detect such malicious profiles should be investigated and robust MCCF algorithms to avoid or alleviate such manipulations should be developed.

5.4. Insights and discussions

In this study, we intend to scrutinize the robustness of the state-of-the-art MCCF techniques and propose a novel shilling attack methodology, called mode attack, exploiting product ratings having a skewed distribution. Attacking MCCF systems differ from traditional single-criterion CF systems since there are various options to apply popular attack types due to multi-dimensional preference data, which also complicates both configuration and detection of these attacks. Therefore, investigating the robustness of MCCF systems against shilling attacks is vital in the proliferation of these systems in real-world applications of recommender systems.

Generating shilling profiles relies on some statistical attack-specific parameters that can be determined in various ways in MCCF environment. Considering all sub-criteria separately to determine such parameters achieves the best performance regarding obtained manipulations. However, it requires the highest amount of information to mount the attacks and utilizing only the ratings of *Overall* criterion to create shilling profiles is much smoother which also realizes similar levels of manipulations.

Another consideration in multi-criteria attack profile generation is the decision of which sub-criteria to exploit and fill the rest sub-criteria randomly or leave blank. Such decision includes a trade-off between the success of the attack and possibility of detection. We propose several exploitation scenarios which would help attackers in avoiding detection of the attack profiles by the service provider. Although intuitively exploiting all available criteria achieves the highest manipulations, the attacker might choose abusing the *Overall* criterion only, significantly essential criteria only, or even a random subset of available criteria to avoid detection. Obtained empirical results demonstrate that even exploiting only the *Overall*



(a) Push Attacks (YM5)

(b) Nuke Attacks (YM5)

(c) Push Attacks (YM20)

(d) Nuke Attacks (YM20)

Fig. 7. Comparison of multi-criteria attack types on aggregation based MCCF algorithms.

criterion achieves non-negligible manipulations and a well-designed attack strategy employing key criteria, possibly with the support of expert opinions, would result in critical damage with the recommendation output.

All MCCF recommendation algorithms analyzed within this study demonstrates severe vulnerability against proposed multi-criteria attacking strategies in almost all scenarios, which concludes there is the need of producing efficient detection methodologies of multi-criteria attacks or developing new robust MCCF algorithms.

6. Conclusions and future work

Multi-criteria preference collection is an intuitive method that helps to understand particularly what a consumer likes about a product or service. Multi-criteria collaborative filtering systems utilize such rating collections to reason about deeper personal inclinations of their users and provide more contextually meaningful recommendations. However, these methods might more substantially be subjected to profile injection attacks than traditional single-criterion systems due to their potential of delivering more dependable suggestions. Such systems have not been evaluated for their robustness against shilling attempts.

In this study, we mainly investigated how to extend well-known shilling attack types to manipulate major multi-criteria recommendation algorithms. We discuss design choices for determining attack-specific parameters based on prevalent multi-criteria data collection schemes and propose a novel attack type, named mode attack. Then, we extensively scrutinize the robustness of three major multi-criteria collaborative filtering algorithms based on similarity aggregation, multidimensional distance-based similarity calculation, and aggregation-function based approaches. Empirical outcomes demonstrate that independent from the recommendation method they utilize, analyzed multi-criteria collaborative filtering methods are severely vulnerable to profile injection manipulations similar to traditional single criterion-based systems. Especially, extended average, reverse bandwagon, and love/hate attacks along with newly proposed mode attack achieve significantly high prediction shift values in produced recommendations. However, bandwagon attack is not as effective as other push attack models due to the difficulty in locating popular items within the experimented dataset. Furthermore, aggregation function-based recommendation method reveals a higher level of exposure against shilling attacks due to its weighting approach among criteria.

Multi-criteria preference elicitation is a promising technique for improved personalization. Therefore, main contribution of the study is to investigate whether multi-criteria recommendation systems can be manipulated through profile injection attacks. Key significance of experimental results is that they approve applicability of shilling intentions on principal multi-criteria collaborative filtering systems, which leads to the question of how to avoid such effects by developing more robust multi-criteria recommendation schemes. Consequently, detecting malicious multi-criteria attack profiles and improving multi-criteria recommendation algorithms warrants future research directions.

Acknowledgments

This work is supported by the Grant No. 215E335 from The Scientific and Technological Research Council of Turkey (TUBITAK). The authors would like to thank Prof. Dietmar Jannach for providing multi-criteria data set collection.

References

Adomavicius, G., & Kwon, Y. (2007). New recommendation techniques for multicriteria rating systems. *IEEE Intelligent Systems, 22*, 48–55. doi:10.1109/MIS.2007.58.

Adomavicius, G., Manouselis, N., & Kwon, Y. (2011). Multi-criteria recommender systems. In F. Ricci, L. Rokach, B. Shapira, & P. Kantor (Eds.), *Recommender Systems Handbook* (pp. 769–803). Boston, MA, USA: Springer. doi:10.1007/978-1-4899-7637-6_25.

Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering, 17*(6), 734–749. doi:10.1109/TKDE.2005.99.

Aggarwal, C. C. (2016). Attack-resistant recommender systems. In *Recommender Systems* (pp. 385–410). Springer. Chapter 12. doi:10.1007/978-3-319-29659-3_12.

Bhaumik, R., Burke, R. D., & Mobasher, B. (2007). Crawling attacks against web-based recommender systems. In *Proceedings of the 2007 International Conference on Data Mining* (pp. 183–189). USA: Las Vegas, Nevada.

Bilge, A., Gunes, I., & Polat, H. (2014). Robustness analysis of privacy-preserving model-based recommendation schemes. *Expert Systems with Applications, 41*(8), 3671–3681. doi:10.1016/j.eswa.2013.11.039.

Bilge, A., & Kaleli, C. (2014). A multi-criteria item-based collaborative filtering framework. In *Proceedings of the 2014 Eleventh International Joint Conference on Computer Science and Software Engineering* (pp. 18–22). Thailand: IEEE, Chon Buri. doi:10.1109/JCSSE.2014.6841835.

Bilge, A., Ozdemir, Z., & Polat, H. (2014). A novel shilling attack detection method. *Procedia Computer Science, 31*, 165–174. doi:10.1016/j.procs.2014.05.257.

Burke, R., Mobasher, B., & Bhaumik, R. (2005). Limited knowledge shilling attacks in collaborative filtering systems. In *Proceedings of the Third International Workshop on Intelligent Techniques for Web Personalization, Edinburgh, Scotland* (pp. 17–24).

Burke, R., Mobasher, B., Bhaumik, R., & Williams, C. (2005). Segment-based injection attacks against collaborative filtering recommender systems. In *Proceedings of the Fifth IEEE International Conference on Data Mining. IEEE, Houston, Texas, USA* (pp. 577–580). doi:10.1109/ICDM.2005.127.

Burke, R., Mobasher, B., Williams, C., & Bhaumik, R. (2006). Detecting profile injection attacks in collaborative recommender systems. In *Proceedings of the Eighth IEEE International Conference on E-commerce Technology and the Third IEEE International Conference on Enterprise Computing, E-commerce, and E-services*. San Francisco, CA, USA: IEEE. doi:10.1109/CEC-EEE.2006.34.

Burke, R., Mobasher, B., Zabicki, R., & Bhaumik, R. (2005). Identifying attack models for secure recommendation. In *Proceedings of the Beyond Personalization Workshop, San Diego, California, USA* (pp. 19–25).

Burke, R., O'Mahony, M. P., & Hurley, N. J. (2015). Robust collaborative recommendation. In F. Ricci, L. Rokach, & B. Shapira (Eds.), *Recommender Systems Handbook* (pp. 961–995). Boston, MA,USA: Springer. doi:10.1007/978-1-4899-7637-6_28.

Cheng, Z., & Hurley, N. (2009a). Effective diverse and obfuscated attacks on model-based recommender systems. In *Proceedings of the Third ACM Conference on Recommender Systems* (pp. 141–148). New York, USA: ACM, New York. doi:10.1145/1639714.1639739.

Cheng, Z., & Hurley, N. (2009b). Robustness analysis of model-based collaborative filtering systems. In *Proceedings of the Artificial Intelligence and Cognitive Science* (pp. 3–15). Dublin, Ireland: Springer. doi:10.1007/978-3-642-17080-5_3.

Cheng, Z., & Hurley, N. (2010a). Analysis of robustness in trust-based recommender systems. In *Proceedings of the Adaptivity, Personalization and Fusion of Heterogeneous Information* (pp. 114–121). Paris, France: Le Centre De Hautes Etudes Internationales D'informatique Documentaire.

Cheng, Z., & Hurley, N. (2010b). Robust collaborative recommendation by least trimmed squares matrix factorization. In *Proceedings of the 2010 Twenty-Second IEEE International Conference on Tools with Artificial Intelligence* (pp. 105–112). Arras, France: IEEE. doi:10.1109/ICTAI.2010.90.

Fan, J., & Xu, L. (2013). A robust multi-criteria recommendation approach with preference-based similarity and support vector machine. In *Advances in Neural Networks* (pp. 385–394). Dalian, China: Springer. doi:10.1007/978-3-642-39068-5_47.

Gunes, I., Kaleli, C., Bilge, A., & Polat, H. (2014). Shilling attacks against recommender systems: a comprehensive survey. *Artificial Intelligence Review, 42*(4), 767–799. doi:10.1007/s10462-012-9364-9.

Herlocker, J. L., Konstan, J. A., Borchers, A., & Riedl, J. (1999). An algorithmic framework for performing collaborative filtering. In *Proceedings of the Twenty-Second Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* (pp. (pp. 230–237). ACM, Berkeley, California, USA. doi:10.1145/312624.312682.

Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems, 22*(1), 5–53. doi:10.1145/963770.963772.

Hu, Y.-C., Chiu, Y.-J., Liao, Y.-L., & Li, Q. (2015). A fuzzy similarity measure for collaborative filtering using non-additive grey relational analysis. *Journal of Grey System, 27*(2), 93–103.

Hurley, N. J., O'Mahony, M. P., & Silvestre, G. C. (2007). Attacking recommender systems: a cost-benefit analysis. *IEEE Intelligent Systems, 22*(3). doi:10.1109/MIS.2007.44.

Jannach, D., Karakaya, Z., & Gedikli, F. (2012). Accuracy improvements for multi-criteria recommender systems. In *Proceedings of the Thirteenth ACM Conference on Electronic Commerce* (pp. 674–689). ACM, New York, New York, USA. doi:10.1145/2229012.2229065.

Jannach, D., Zanker, M., & Fuchs, M. (2014). Leveraging multi-criteria customer feedback for satisfaction analysis and improved recommendations. *Information Technology & Tourism, 14*(2), 119–149. doi:10.1007/s40558-014-0010-z.

Lakiotaki, K., Tsafarakis, S., & Matsatsinis, N. (2008). UTA-Rec: a recommender system based on multiple criteria analysis. In *Proceedings of the 2008 ACM Conference on Recommender Systems* (pp. 219–226). Lausanne, Switzerland: ACM. doi:10.1145/1454008.1454043.

Lam, S. K., & Riedl, J. (2004). Shilling recommender systems for fun and profit. In *Proceedings of the Thirteenth International Conference on World Wide Web* (pp. 393–402). New York, USA: ACM, New York. doi:10.1145/988672.988726.

Lee, H.-H., & Teng, W.-G. (2007). Incorporating multi-criteria ratings in recommendation systems. In *Proceedings of the IEEE International Conference on Information Reuse and Integration* (pp. 273–278). Las Vegas, IL, USA: IEEE. doi:10.1109/IRI.2007.4296633.

Long, Q., & Hu, Q. (2010). Robust evaluation of binary collaborative recommendation under profile injection attack. In *Proceedings of the IEEE International Conference on Progress in Informatics and Computing* (pp. 1246–1250). Shangai, China: IEEE. doi:10.1109/PIC.2010.5687920.

Maneeroj, S., Samatthiyadikun, P., Chalermpornpong, W., Panthuwadeethorn, S., & Takasu, A. (2012). Ranked criteria profile for multi-criteria rating recommender. In *Proceedings of the International Conference on Information Systems, Technology and Management* (pp. 40–51). Grenoble, France: Springer. doi:10.1007/978-3-642-29166-1_4.

Manouselis, N., & Costopoulou, C. (2007a). Analysis and classification of multi-criteria recommender systems. *World Wide Web, 10*(4), 415–441. doi:10.1007/s11280-007-0019-8.

Manouselis, N., & Costopoulou, C. (2007b). Experimental analysis of design choices in multiattribute utility collaborative filtering. *International Journal of Pattern Recognition and Artificial Intelligence, 21*(02), 311–331. doi:10.1142/S021800140700548X.

Mobasher, B., Burke, R., Bhaumik, R., & Sandvig, J. J. (2007). Attacks and remedies in collaborative recommendation. *IEEE Intelligent Systems, 22*(3), 56–63. doi:10.1109/MIS.2007.45.

Mobasher, B., Burke, R., Bhaumik, R., & Williams, C. (2007). Toward trustworthy recommender systems: an analysis of attack models and algorithm robustness. *ACM Transactions on Internet Technology, 7*(4), 23. doi:10.1145/1278366.1278372.

Nguyen, P., Kang, M., Kim, J.-M., Ahn, B.-H., Ha, J.-M., & Choi, B.-K. (2015). Robust condition monitoring of rolling element bearings using de-noising and envelope analysis with signal decomposition techniques. *Expert Systems with Applications, 42*(22), 9024–9032. doi:10.1016/j.eswa.2015.07.064.

Nilashi, M., bin Ibrahim, O., & Ithnin, N. (2014a). Hybrid recommendation approaches for multi-criteria collaborative filtering. *Expert Systems with Applications, 41*(8), 3879–3900. doi:10.1016/j.eswa.2013.12.023.

Nilashi, M., bin Ibrahim, O., & Ithnin, N. (2014b). Multi-criteria collaborative filtering with high accuracy using higher order singular value decomposition and neuro-fuzzy system. *Knowledge-Based Systems, 60*, 82–101. doi:10.1016/j.knosys.2014.01.006.

Nilashi, M., bin Ibrahim, O., Ithnin, N., & Sarmin, N. H. (2015). A multi-criteria collaborative filtering recommender system for the tourism domain using Expectation Maximization (EM) and PCA-ANFIS. *Electronic Commerce Research and Applications, 14*(6), 542–562. doi:10.1016/j.elerap.2015.08.004.

O'Mahony, M., Hurley, N., Kushmerick, N., & Silvestre, G. (2004). Collaborative recommendation: a robustness analysis. *ACM Transactions on Internet Technology, 4*(4), 344–377. doi:10.1145/1031114.1031116.

O'Mahony, M. P., Hurley, N. J., & Silvestre, G. C. C. (2005). Recommender systems: attack types and strategies. In *Proceedings of the Twentieth National Conference on Artificial Intelligence* (pp. 334–339). Pittsburgh, Pennsylvania, USA: AAAI.

O'Mahony, M. P., Hurley, N. J., & Silvestre, G. (2003). An evaluation of the performance of collaborative filtering. In *Proceedings of the Fourteenth Irish Artificial Intelligence and Cognitive Science Conference. Dublin, Ireland* (pp. 164–168).

Palanivel, K., & Sivakumar, R. (2010). A study on implicit feedback in multicriteria e–commerce recommender system. *Journal of Electronic Commerce Research, 11*(2), 140–156.

Resnick, P., & Sami, R. (2008). The information cost of manipulation-resistance in recommender systems. In *Proceedings of the 2008 ACM conference on Recommender Rystems* (pp. 47–154). Lausanne, Switzerland: ACM. doi:10.1145/1454008.1454033.

Risnumawan, A., Shivakumara, P., Chan, C. S., & Tan, C. L. (2014). A robust arbitrary text detection system for natural scene images. *Expert Systems with Applications, 41*(18), 8027–8048. doi:10.1016/j.eswa.2014.07.008.

Sanchez-Vilas, F., Ismoilov, J., Lousame, F. P., Sanchez, E., & Lama, M. (2011). Applying multicriteria algorithms to restaurant recommendation. In *Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology.* (pp. 87–91). Lyon, France: IEEE. doi:10.1109/WI-IAT.2011.124.

Wilson, D. C., & Seminario, C. E. (2013). When power users attack: assessing impacts in collaborative recommender systems. In *Proceedings of the Seventh ACM conference on Recommender Systems* (pp. 427–430). Hong Kong, China: ACM. doi:10.1145/2507157.2507220.

Yi, H., & Zhang, F. (2016). Robust recommendation method based on suspicious users measurement and multidimensional trust. *Journal of Intelligent Information Systems, 46*(2), 349–367. doi:10.1007/s10844-015-0375-2.

Yurekli Yilmazel, B., & Kaleli, C. (2016). Robustness analysis of arbitrarily distributed data-based recommendation methods. *Expert Systems with Applications, 44*, 217–229. doi:10.1016/j.eswa.2015.09.012.

Zhang, F. (2009). Reverse bandwagon profile inject attack against recommender systems. In *Proceedings of the Second International Symposium on Computational Intelligence and Design* (pp. 15–18). Changsha, China: IEEE. doi:10.1109/ISCID.2009.11.

Zhang, F. (2010). Analysis of love-hate shilling attack against e-commerce recommender system. In *Proceedings of the 2010 International Conference of Information Science and Management Engineering* (pp. 318–321). Xi'an, China: IEEE. doi:10.1109/ISME.2010.116.