Empowering shilling attacks with Katz and Exclusivity-based relatedness

Felice Antonio Merra[1,*,†], Vito Walter Anelli[2,*], Yashar Deldjoo[2], Tommaso Di Noia[2] and Eugenio Di Sciascio[2]

[1] *Amazon Web Services, Berlin, Germany*

[2] *Politecnico di Bari, Bari, Italy*

Abstract

Several domains have widely benefited from the adoption of Knowledge graphs ($\mathcal{KG}s$). For recommender systems (RSs), the adoption of $\mathcal{KG}s$ has resulted in accurate, personalized recommendations of items/products according to users' preferences. Among different recommendation techniques, collaborative filtering (CF) is one the most promising approaches to build RSs. Their success is due to the effective exploitation of similarities/correlations encoded in user interaction patterns. Nonetheless, their strength is also their weakness. A malicious agent can add fake user profiles into the platform, altering the genuine similarity values and the corresponding recommendation lists. While the research community has extensively studied $\mathcal{KG}s$ to solve various recommendation problems, including the empowerment of semantic-aware shilling attacks, limited attention has been paid on exploiting $\mathcal{KG}s$ relatedness measures, i.e., *Katz* and *Exclusivity*-based, computed considering 1-hop of graph exploration. We performed an extensive experimental evaluation with four state-of-the-art recommendation systems and two well-known recommendation datasets to investigate the effectiveness of introducing relatedness information on semantic-aware shilling attacks. Since the semantics of relations has a crucial role in $\mathcal{KG}s$, we have also analyzed the impact of relations' semantics by grouping them in various classes. Experimental results indicate the benefit of embracing $\mathcal{KG}s$ in favor of the attackers' capability in attacking recommendation systems.

Keywords

Shilling Attacks, Collaborative Filtering, Knowledge Graphs

1. Introduction

The advent of Knowledge Graphs ($\mathcal{KG}s$) has changed the way structured information is stored. It has become much more than that developed to make the Semantic Web a concrete idea. The core idea of building a semantic network in which information is represented as directed labeled graphs (RDF graphs) is disarmingly simple. Nevertheless, thanks to the possibilities it paves, it has been welcomed with several promises and expectancies. Complete interoperability, the

Sixth Knowledge-aware and Conversational Recommender Systems (KaRS) Workshop @ RecSys 2024, September 14–18 2024, Bari, Italy.

[*] Corresponding authors: Felice Antonio Merra (felmerra@amazon.de) and Vito Walter Anelli (vitowalter.anelli@poliba.it).

[†] Work done while at Politecnico di Bari before joining Amazon.

felmerra@amazon.de (F. A. Merra); vitowalter.anelli@poliba.it (V. W. Anelli); yashar.deldjoo@poliba.it (Y. Deldjoo); tommaso.dinoia@poliba.it (T. D. Noia); eugenio.disciascio@poliba.it (E. D. Sciascio)

0009-0003-8429-3487 (F. A. Merra)

ability to link knowledge across domains, and the possibility to exploit Logical inference and proofs are just a few of them. In numerous domains, the exploitation of the $\mathcal{KG}$ information has become the norm. Thanks to the appearance of wide-ranging Linked Datasets like DBpedia and Wikidata, we have witnessed the flourishing of novel techniques in several research fields, like Machine Learning, Information Retrieval, and Recommender Systems. To date, Recommender Systems (RSs) are considered the focal solution to assist users' decision-making process. Since the volume of the available products on the Web overwhelms the users, RSs support and ease the decision process. Among them, collaborative filtering (CF) recommendation techniques have shown very high performance in real-world applications (e.g., Amazon [1]). Their rationale is to analyze products experienced by similar users to produce tailored recommendations. Algorithmically speaking, they take advantage of user-user and item-item similarities. Regrettably, malicious users may want to jeopardize the operation of the recommendation platform. For example, they might be a rival company or agents who want to increase (or decrease) the visibility of a particular product. Whatever they are motivated by, the problem is that these similarities are vulnerable to the insertion of fake profiles. This kind of attack is called the *shilling attack* [2], which aims to *push* or *nuke* the probabilities to recommend an item. The malicious agent (or adversary) can rely on an extensive list of techniques to conduct the attack. Researchers and companies have classified them into two broad categories [3]: *low-knowledge* and *informed* attack strategies. In the former attacks, the adversary has poor system-specific knowledge [4, 5]. In the latter, the attacker has an accurate knowledge of the recommendation model and the data distribution [4, 6].

Interestingly, despite the astonishing spread of $\mathcal{KG}s$, little attention has been paid to knowledge-aware strategies to mine RS's security. Since $\mathcal{KG}s$ provide comprehensive information on numerous knowledge domains, a malicious agent can decide to attack RSs making use of the items' semantic descriptions. One work exploiting publicly available information obtained from $\mathcal{KG}$ to generate more influential fake profiles to threaten CF models' performance is named semantics-aware shilling attack *SAShA* [7]. This work extended state-of-the-art shilling attack approaches such as *Random*, *BandWagon*, and *Average* profiting from publicly available semantic information without supposing any additional knowledge about the system. While the previous study modified these attacks considering the cosine vector similarity between the semantic description of items, in this work, we identify that *SAShA* only considers the cosine similarity across the semantic details, which is not particularly suited to bring out semantic relatedness.

In this work, we have overcome this limitation by going beyond the cosine similarity by considering *Katz centrality* and *Exclusivity-based relatedness*. Finally, to provide a more fine-grained analysis, we have grouped the semantic relations into three classes: ontological, categorical, and factual relations.

In detail, this study extends the state-of-the-art approach for the integration of semantics in the shilling attacks [7] in numerous directions:

1. two novel graph topological and semantic approaches to build the set of items from which the adversary can craft the fake profiles;

2. a novel semantic shilling attack strategy based on *BandWagon* strategy;

3. a deeper discussion of the experimental results involving several dimensions: type of

considered relation, recommendation model, amount of injected fake profiles, and dataset.

We have conducted extensive experiments to evaluate the impact of proposed attacks against the recommendation models. To this end, we have exploited two real-world recommender system datasets (`LibraryThing` and `Yahoo!Movies`). Experimental results sharply indicate that $\mathcal{KG}$ information is a valuable source of knowledge that improves attacks' effectiveness. Moreover, adopting semantic relatedness measures can unleash the full potential of the semantics-aware attacks.

The remainder of the paper proceeds as follows. In Section 2, we provide an overview of the state-of-the-art recommendation models and shilling attacks. Section 3 describes the proposed extensions to the *SAShA* by introducing the semantic relatedness measures, and formalizes the semantic attack strategies. Section 4 focuses on the experimental validation of the proposed attack scenarios. We also provide an in-depth discussion of the experimental results analyzing the several dimensions of the study. Finally, in Section 6, we draw some conclusions and introduce the open challenges.

2. Related Work

2.1. Recommender Systems

A recommendation problem can be stated as finding a utility function to automatically predict how much users will like unknown items.

Definition 1 (Recommendation Problem). *Let $\mathcal{U}$ and $\mathcal{I}$ denote a set of users and items in a system, respectively. Each user $u \in \mathcal{U}$ is related to $\mathcal{I}_u^+$, the set of items she has consumed, or her user profile. Given a utility function $g : \mathcal{U} \times \mathcal{I} \rightarrow \mathbb{R}$ a* ***Recommendation Problem*** *is defined as*

$$\forall u \in \mathcal{U},\ i'_u = \underset{i \in \mathcal{I}}{\operatorname{argmax}}\, g(u, i)$$

where i'_u denotes an item not consumed by the user u before. We assume that the preference of user $u \in \mathcal{U}$ on item $i \in \mathcal{I}$ is encoded with a continuous-valued preference score $r_{ui} \in \mathcal{R}$, where $\mathcal{R}$ represent the set of (u, i) pairs for which r_{ui} is known.

The major class of recommendation models includes content-based filtering (CBF), collaborative filtering (CF), and hybrid [8, 9]. CBF models build a profile of user interests based on the content features of the items preferred by that user (liked or consumed), characterizing the nature of her interests. CF models compute recommendations based on similarities in preference patterns of like-minded users. They can be classified according to neighborhood-based and model-based. Neighborhood-based models compute recommendations exclusively based on correlations in interactions across users (user-based CF [10, 11]) or items (item-based CF [12, 11]), while model-based approaches learn a model that can be queried in the production phase to generate recommendations for a given user profile, e.g., MF [13].

2.2. Knowledge-aware RSs

RSs exploit various side information such as metadata (e.g., tags, reviews) [14], social connections [15], image and audio signal features [16], and users-items contextual data [17] to build more in-domain [18] (i.e., domain-dependent), cross-domain [19], or context-aware [20, 21] recommendation models. Among the diverse information sources, Knowledge Graphs ($\mathcal{KG}s$) are one of the most relevant. A $\mathcal{KG}$ is a heterogeneous network that encodes multiple relationships, edges, nodes, and links items at high-level relationships, making them a strong item representation technique. Thanks to the heterogeneous domains that $\mathcal{KG}s$ cover, the design of knowledge-based recommendation systems has arisen as a specific research field of its own in the community of RSs, usually referred to by Knowledge-aware Recommender Systems (KaRS [22, 23]). Knowledge-aware Recommender Systems have been particularly impactful for several research domains:
$mathcalKG$*/graph-embeddings* [24, 25, 26, 27, 28, 29, 30], *hybrid collaborative/content-based recommendation* [25, 31], *knowledge-completion, link-prediction, knowledge-discovery* [32, 33, 34, 30, 35, 36, 37, 38], *knowledge-transfer, cross-domain recommendation* [39, 19, 40], *interpretable/explainable-recommendation* [41, 42, 43, 44, 31], *graph-based recommendation* [45, 46, 47, 48, 49, 50], *content-based recommendation* [51, 52].

All the former advances have been shown to enhance the recommendation quality or the overall user experience. Although the algorithms differ on many levels, we can still classify recommendation techniques into two broad approaches: *Path-based* methods [45, 46, 47, 50, 53, 54], which employ paths and meta-paths to estimate the user-item similarities or the nearest items; and *KG embedding-based* techniques [45, 26, 31, 55, 21, 56], which leverage $\mathcal{KG}$ embeddings (usually obtained through matrix factorization or neural network encoding) for items' representation.

2.3. Security of RSs

Malicious users, the *adversaries*, can meticulously craft fake profiles to poison the data and alter the recommendation behavior toward malicious goals [57, 58, 59]. An adversary may execute a **shilling attack** (injection of malicious profiles) to achieve a whole different set of objectives. To name a few, she may want to demote competitor products [4], misuse the underlying recommendation system [2], or increase the recommendability of specific products [60, 61].

The research works on shilling attacks explored two main research perspectives: proposing and investigating attack strategies with their effects on the recommendation performance [4, 62, 63, 64] and exploring defensive mechanisms [59, 65, 66, 67, 68, 69].

A typical characteristic of the first line of research on shilling attacks is that the adversary's knowledge is related only to the recommender system's user-item interaction matrix. Furthermore, Anelli et al. [7] demonstrate that publicly available $\mathcal{KG}$ improves adversary's efficacy, also in the case of *low-informed* attacks (e.g., Random). In this work, we extend the *SAShA* framework to verify the possible improvement of the adversary's efficacy when processing the $\mathcal{KG}$ information with semantic similarity measures.

Note that this work focuses on shilling attacks, that are hand-engineered strategies to study recommender systems' security. This research line is different from machine-learned data poisoning attack [70, 71, 72, 73, 74] and adversarial machine-learned attacks [75, 76, 61, 77, 78, 79] where adversaries adopt optimization techniques to create perturbations.

3. Method

This section introduces the reader to the notations and formalisms that may help understand the design of shilling attacks against targeted items integrating information obtained from a knowledge graph ($\mathcal{KG}$).

3.1. Knowledge Graph Content Extraction

A knowledge graph is a structured repository of knowledge, designed in the form of a graph, that encodes various kinds of information:

- **Factual.** General statements as *Rika Dialina was born in Crete* or *Heraklion is the capital of Crete* that describe an entity by using a controlled vocabulary of predicates that connect the entity to other entities (or literal values).
- **Categorical.** These statements connect the entity to a particular category (i.e., the categories associated with a Wikipedia page). Often, categories are in turn organized as a hierarchy.
- **Ontological.** These are formal statements that describe the entity's nature and its ontological membership to a specific class. Classes are often organized in a hierarchical structure. In contrast to categories, sub-classes and super-classes are connected through IS-A relations.

In a knowledge graph, we can express statements through triplets $\sigma \xrightarrow{\rho} \omega$, with a *subject* ($\sigma$), a *predicate (or relation)* (ρ), and an *object* (ω). There are several ways to transform the knowledge coming from a knowledge graph into a feature. We have chosen to represent each distinct path as an explicit feature [31].

Given a set of items $I = \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_N\}$ in a collection and the corresponding triples $\langle i, \rho, \omega \rangle$ in a knowledge graph, the set of 1-hop features is defined as 1-HOP-$F = \{\langle \rho, \omega \rangle \mid \langle i, \rho, \omega \rangle \in \mathcal{KG}$ with $i \in I\}$.

3.2. Entity Similarity/Relatedness in KGs

The keystone of the Knowledge Graph representation is the semantics enclosed in the resource description and the predicates that connect the different resources. Nevertheless, if the metric to compute similarities between the resources is not carefully chosen, this piece of information is lost irretrievably. Motivated by this awareness, we decided to consider a broad spectrum of diverse similarity/relatedness metrics in addition to the **cosine vector similarity** [80] (used in $SASHA$ [7]: ***Katz*** **centrality** [81] and ***Exclusivity*-based semantic relatedness** [82]. In general, the three metrics cover different aspects of the similarity between the resource a signal of the overlap of the descriptions and a semantics-aware signal that highlights the specificity of the relations between the resources.

Cosine Vector Similarity is a well-known similarity that is very popular in recommendation systems. The idea is to measure how similar the two different representations are. Suppose a numerical vector can represent the resource description, with the number of the predicate-object chains observed in the $\mathcal{KG}$ being the vector's cardinality. Mathematically, it measures the cosine of the angle between two vectors that represent two different resources. The smaller the angle, the higher the cosine similarity is, and thus the similarity. Suppose i and j are two items in the $\mathcal{KG}$, and $F(\cdot)$ is a function that returns the features associated with an entity in the $\mathcal{KG}$. Hence $in(i, f)$ is a function that returns 1 if entity i is associated with feature f, else 0. The Cosine Vector Similarity has been already formulated for $\mathcal{KG}$ as follows [80]:

$$sim(i,j) = \frac{\sum_{f \in F(i) \cup F(j)} in(i,f) \cdot in(j,f)}{\sqrt{\sum_{f \in F(i)} in(i,f)^2} \cdot \sqrt{\sum_{f \in F(j)} in(j,f)^2}} \tag{1}$$

This is the baseline method used in $SAShA$ [7]/

Katz centrality [81] is a famous graph-centrality measure that inspired several semantics-aware metrics [83, 82]. Katz suggests that the probability of the path between two nodes can indicate the effectiveness of the link. Given a constant probability for a single-hop path, called α, the whole path's overall probability is α^y, where y is the number of the nodes involved. Hulpus [82] exploits the rationale to build a relatedness measure. Therefore, he defined the Katz relatedness between two items i and j as the accumulated score over the top-t shortest paths between them.

$$rel^{(t)}_{Katz}(i,j) = \frac{\sum_{p \in SP^{(t)}_{ij}} \alpha^{length(p)}}{t} \tag{2}$$

where $SP^{(t)}_{ij}$ is the set of the top-t shortest paths between items i and j. This is the first novel similarity metric tested in this work. Note that the shortest path has a larger implication in multi-hops experiments; results on these have been reported in .

Exclusivity-based semantic relatedness [82] is a semantic relatedness measure that takes into account the type of relations that connect two nodes. The idea is that two concepts are strongly connected if the type of relations between them is different from the type of relations they have with other concepts. This property of relations, named exclusivity, is defined as follows.

Suppose a predicate ρ of type τ between two items i and j, directed from i to j. The exclusivity of predicate ρ is the probability of selecting, with a uniform random distribution, a predicate ρ' of type τ among the predicates of type τ that exit resource i and enter node j, such that predicate ρ' is exactly the predicate ρ:

$$exclusivity(i \xrightarrow{\mathcal{T}} j) = \frac{1}{|i \xrightarrow{\mathcal{T}} *| + |* \xrightarrow{\mathcal{T}} j| - 1} \tag{3}$$

where $|i \xrightarrow{\mathcal{T}} *|$ denotes the cardinality of relations of type $\tau \in \mathcal{T}$ that exit resource i, and $|* \xrightarrow{\mathcal{T}} j|$ denotes the number of relations of type $\tau \in \mathcal{T}$ that enter resource j. Since the relation $i \xrightarrow{\mathcal{T}} j$ is in $|i \xrightarrow{\mathcal{T}} *|$ and in $|* \xrightarrow{\mathcal{T}} j|$, 1 is subtracted from the denominator. The exclusivity score for a predicate falls inside the $(0, 1]$ interval. The value 1 denotes the extreme case in which the predicate is the only relation of its type for both i and j.

Given a path through $\mathcal{KG}$, $\mathcal{P} = n_1 \xrightarrow{\tau} n_2 \xrightarrow{\tau_2}, \dots, n_k$ with $\tau_i \in \mathcal{T}^{\mp}$, the weight of the path is defined as:

$$weight(\mathcal{P}) = \frac{1}{\sum_i \frac{1}{exclusivity(n_i \xrightarrow{\tau_i} n_{i+1})}} \tag{4}$$

Finally, the relatedness between two resources can be computed as the sum of the path weights of the top-t paths between the resources with the highest weights. To penalize longer paths, a constant length decay factor, $\alpha \in (0, 1]$, can be introduced. The overall exclusivity-based relatedness measure is therefore defined as follows:

$$rel^{(t)}_{Excl}(i, j) = \sum_{\mathcal{P}_n \in P^t_{ij}} \alpha^{lenght(\mathcal{P}_n)} weight(\mathcal{P}_n) \tag{5}$$

This is the second novel similarity metric tested in this work.

3.3. Attacks

Table 1
Overview of shilling attack strategies and their profile composition for adversaries' goal of *pushing* a target item ($\mathcal{I}_T$).

	Selected Items ($\mathcal{I}_S$)		Filler Items ($\mathcal{I}_F$)			$\mathcal{I}_\phi$	$\mathcal{I}_T$
Attack Type	**Number Items**	**Rating**	**Selection**	**Number Items**	**Rating**		
Random [4]	$\emptyset$		Random	$\frac{\sum_{u \in \mathcal{U}} \lvert\mathcal{I}_u\rvert}{\lvert\mathcal{U}\rvert} - 1$	$rnd(N(\mu, \sigma^2))$	$\mathcal{I} - \mathcal{I}_F$	max
Love-Hate [84]	$\emptyset$		Random	$\frac{\sum_{u \in \mathcal{U}} \lvert\mathcal{I}_u\rvert}{\lvert\mathcal{U}\rvert} - 1$	min	$\mathcal{I} - \mathcal{I}_F$	max
Popular [85]	$\frac{\sum_{u \in \mathcal{U}} \lvert\mathcal{I}_u\rvert}{\lvert\mathcal{U}\rvert} - 1$	min **if** $\mu_f < \mu$ **else** $min + 1$		$\emptyset$		$\mathcal{I} - \mathcal{I}_S$	max
Average [4]	$\emptyset$		Random	$\frac{\sum_{u \in \mathcal{U}} \lvert\mathcal{I}_u\rvert}{\lvert\mathcal{U}\rvert} - 1$	$rnd(N(\mu_f, \sigma_f^2))$	$\mathcal{I} - \mathcal{I}_F$	max
Bandwagon [62]	$(\frac{\sum_{u \in \mathcal{U}} \lvert\mathcal{I}_u\rvert}{\lvert\mathcal{U}\rvert})/2 - 1$	max	Random	$(\frac{\sum_{u \in \mathcal{U}} \lvert\mathcal{I}_u\rvert}{\lvert\mathcal{U}\rvert})/2$	$rnd(N(\mu, \sigma^2))$	$\mathcal{I} - \mathcal{I}_S - \mathcal{I}_F$	max
P. Knowledge [57]	$\frac{\sum_{u \in \mathcal{U}} \lvert\mathcal{I}_u\rvert}{\lvert\mathcal{U}\rvert} - 1$	max		$\emptyset$		$\mathcal{I} - \mathcal{I}_S$	max
***SAShA* Random**	$\emptyset$		Semantics-aware	$\frac{\sum_{u \in \mathcal{U}} \lvert\mathcal{I}_u\rvert}{\lvert\mathcal{U}\rvert} - 1$	$rnd(N(\mu, \sigma^2))$	$\mathcal{I} - \mathcal{I}_F$	max
***SAShA* Love-Hate**	$\emptyset$		Semantics-aware	$\frac{\sum_{u \in \mathcal{U}} \lvert\mathcal{I}_u\rvert}{\lvert\mathcal{U}\rvert} - 1$	min	$\mathcal{I} - \mathcal{I}_F$	max
***SAShA* Average**	$\emptyset$		Semantics-aware	$\frac{\sum_{u \in \mathcal{U}} \lvert\mathcal{I}_u\rvert}{\lvert\mathcal{U}\rvert} - 1$	$rnd(N(\mu_f, \sigma_f^2))$	$\mathcal{I} - \mathcal{I}_F$	max
***SAShA* Bandwagon**	$(\frac{\sum_{u \in \mathcal{U}} \lvert\mathcal{I}_u\rvert}{\lvert\mathcal{U}\rvert})/2 - 1$	max	Semantics-aware	$(\frac{\sum_{u \in \mathcal{U}} \lvert\mathcal{I}_u\rvert}{\lvert\mathcal{U}\rvert})/2$	$rnd(N(\mu, \sigma^2))$	$\mathcal{I} - \mathcal{I}_S - \mathcal{I}_F$	max

where (μ, σ) are the dataset average rating and rating variance, (μ_f, σ_f) are the filler item $\mathcal{I}_F$ rating average and variance, and min and max are the minimum and maximum rating value. rnd function generates one integer (i.e., rating) from a discrete uniform distribution.

Given a Recommendation Problem, a **Shilling Profile** ($\mathcal{SP}$) is a rating profile partitioned into four sets:

$$\mathcal{SP} = \mathcal{I}_S + \mathcal{I}_F + \mathcal{I}_\phi + \mathcal{I}_T \tag{6}$$

where $\mathcal{I}_S$ denotes the *selected* item set containing items identified by the attacker to maximize the effectiveness of the attack, $\mathcal{I}_F$ is the *filler* item set, containing a set of randomly selected items to which rating scores are assigned to make them imperceptible. $\mathcal{I}_T$ is the target item, for which the recommendation model will make a prediction, aimed to be maximal (for push attack) or minimum (for nuke attack). Finally, $\mathcal{I}_\phi$ is the unrated item set, holding a number of items without any ratings.

Note that $\mathcal{I}_S$ and $\mathcal{I}_F$ are chosen depending on the attack strategy, and the attack size is the number of injected fake user profiles. Throughout this paper, we use $\phi = |\mathcal{I}_F|$ to represent the filler size, $\alpha = |\mathcal{I}_S|$ the selected item set size and $\chi = |\mathcal{I}_\emptyset|$ to show the size of unrated items. Table 1 summarizes the main parameters involved in the implementation of most prominent shilling attacks against rating-based CF models. For instance, it can be seen that $SAShA$ attacks are the extension of state-of-the-art shilling attacks, with the difference that selection of the filler item set ($\mathcal{I}_F$) is chosen semantically, not randomly.

Semantics-aware Random Attack is an extension of the baseline Random Attack [4]. The baseline version is a naive attack, which uses randomly chosen items ($\alpha = 0, \phi =$ *profile-size*) to create a fake user profile. The ratings attributed to $\mathcal{I}_\phi$ are sampled from a uniform distribution (see Table 1). $SAShA$ modify this attack by selecting the items to complete $\mathcal{I}_F$ with the cosine-similarity. In this work, we exploit semantic similarities/relatedness between the items in the catalog e the target item using $\mathcal{KG}$-based features (cf. Section 3.1). Afterward, we identify the most similar items ($\mathcal{I}_T$) by considering the first quartile of most similar items, and we extract ϕ items from this set by adopting a uniform distribution.

Semantics-aware Average Attack is an informed attack strategy that extends the AverageBots attack [5]. The baseline attack leverages the mean and variance of the ratings, which is then used to sample each filer item's rating from a normal distribution built using these values. Similar to the previous semantics-aware attack extension, we extract the filler items by exploiting semantic similarities derived from a $\mathcal{KG}$. Finally, as before, we consider the items in the first quartile of the most semantically similar/related to $\mathcal{I}_T$ as the candidate filler items ($\mathcal{I}_F$).

Semantics-aware BandWagon Attack is a low-knowledge attack that extends the standard BandWagon attack [62]. We leave unchanged the injection of the selected items ($\mathcal{I}_S$), which are the most popular ones and on which we associate the maximum possible rating (see Table 1). However, similarly to the previous two semantic attack extensions, we complete $\mathcal{I}_F$ by taking into account the semantic similarity/relatedness between the target item $\mathcal{I}_T$ and the rest of the catalog.

4. Experimental Setting

4.1. Dataset

We tested the proposed approaches on two datasets. The first is `LibraryThing` [50] and it is a popular dataset whose interactions originate from `librarything.com`, a social cataloging web application. The dataset contains user-item rating scores ranging from a minimum of 1 to a maximum of 10. As presented in [7], we use a reduced version by randomly extracting the 25% of products in the catalog. Furthermore, we apply a 5-core filtering by removing all the users with less than five interactions to focus the study on active users. These users are of adversaries' interest since they could more likely buy the pushed products.

The second is `Yahoo!Movies` which is a recommendation dataset released by `research.yahoo.com` with ratings collected up to November 2003. The dataset also provides mappings to the `MovieLens` and `EachMovie` catalogs. The recorded interactions consist of ratings ranging from 1 to 5.

Both datasets have a mapping between the items in the catalogs and `DBpedia` knowledge-base entities. In particular, we use the mappings publicly available at https://github.com/sisinflab/LinkedDatasets. Table 2 reports the statistics of both datasets.

Table 2
Datasets statistics.

Dataset	#Users	#Items	#Ratings	Sparsity	#F-1Hop
LibraryThing	4,816	2,256	76,421	99.30%	56,019
Yahoo!Movies	4,000	2,526	64,079	99.37%	105,733

Table 3
Selected features.

	Categorical		Ontological		Factual	
Dataset	Total	Selected	Total	Selected	Total	Selected
LibraryThing	3,890	373	2,090	311	50,039	1,972
Yahoo!Movies	5,555	1,192	3,036	722	97,142	7,690

4.1.1. Feature Extraction

Once the items are semantically reconciled with DBpedia entities, we remove the noisy features whose triples contain one of the following predicates: owl:sameAs, dbo:thumbnail, foaf:depiction, prov:wasDerivedFrom, foaf:isPrimaryTopicOf. The feature denoising procedure follows the methodology proposed in [42, 31].

4.1.2. Feature Selection

To perform the analysis of the groups (or types) of semantic features, we implement our proposed semantics-aware attacks by considering three different types of features, i.e., categorical (CS), ontological (OS), and factual (FS), a feature taxonomy commonly adopted in the Semantic Web community [31]. We apply the following policies **Categorical-1H**, we use the features with the property dcterms:subject, **Ontological-1H**, we select the features containing the property rdf:type, and **Factual-1H**, we consider all the features except ontological and categorical features.

4.1.3. Feature Filtering

This work aims to study the attack performance differences up to the first hop. Addressing this aim, we obtain thousands of features for both LibraryThing and Yahoo!Movies as reported in the last two columns of Table 2. Measuring semantic similarities across the item catalog would quickly become unfeasible. However, some features only occur once and provide no useful informative or collaborative information. Therefore, we decided to drop off irrelevant features following the filtering technique proposed by Di Noia et al. [50], Paulheim and Fürnkranz [86]. In detail, we removed all the features with more than 99.74% of missing values and distinct values. Table 3 shows the remaining features' statistics after applying the extraction, selection, and filtering.

4.2. Recommender Models

In this work, we test our attack proposal (see Section 3.3) against four baseline collaborative RSs. Two neighborhood-based: **User-*k*NN**, on which we [10, 11] set the size of the neighborhood k to 40, and **Item-*k*NN** [12, 11], where $k = 40$ too. Two model-based: **Matrix Factorization (MF)** [13], where we set the size of latent vectors to 100, and **Neural Matrix Factorization (NeuMF)** [87], on which we used a deep neural network composed by 4 fully connected dense layers with {64, 32, 16, 8} hidden units.

4.3. Evaluation Metrics

We evaluate the attack performance using $HR@N$. The metric describes the average presence of target items in the top-N recommendation lists generated for all the users after the attack. Since we will experiment with the case of push attacks, it follows that the adversary's goal is to increase/maximize the HR of the attacked/targeted items.

4.4. Evaluation Protocol

We perform 180 experiments for each dataset, totaling 720 experiments. Following the evaluation procedure used in [60, 4], we generate the list of recommendations for each recommendation model before executing the attack. After having measured the position and predicted score for each target item-user pair, we simulated the attack. Each attack is performed against 50 randomly selected items in each dataset. Furthermore, we perform each attack using three different amounts of injected shilling profiles: 1%, 2.5%, and 5% of the total number of users, as adopted in [7, 63, 5]. Regarding the relatedness measures, we set the $\alpha = 0.25$ and the t-path length to 10 for both metrics. Datasets and code will be made publicly available.

5. Experimental Results

Table 4 and Table 5 report the results. Across the next sections, we identify an attack combination using the format <dataset, recommender, attack strategy, feature type, similarity measures, granularity>, e.g., <`Yahoo!Movies`, User-*k*NN, Average, Categorical, Katz, 1%>.

First, we observe that the results obtained on the `Yahoo!Movies` dataset (Table 5) are more indicative of attacks' effectiveness independently of the attack dimensions, confirming the findings in the previous work by Anelli et al. [7]. Furthermore, Table 4 confirms the semantics-aware strategy's efficacy over the baseline, either for the average or random attacks. For instance, the semantic strategies outperformed all the <`LibraryThing`, Random> and <`LibraryThing`, Average> baseline attacks independently of the recommender model and the size of attacks. However, differently from the results on `Yahoo!Movies`, on <`LibraryThing`, BandWagon>, the baseline attack's effectiveness did not improve. This behavior might be justified by the fact that a bandwagon attack builds profiles by filling the 50% of the profile with the most *popular* items, it might make the semantic strategy that identifies the informative filler items ineffective.

In addition to the general results, we provide a more in-depth discussion answering three research questions.

Table 4
Hit Ratio (HR) result values evaluated on top-10 recommendation lists for the `LibraryThing` dataset. We report in bold the highest value of each column given an adversary budget and knowledge. The usage of *Cosine* similarity is the baseline approach proposed in *SAShA*.

			User-kNN			**Item-kNN**			**MF**			**NeuMF**		
Attack	**Feature Type**	**Similarity**	1	2.5	5	1	2.5	5	1	2.5	5	1	2.5	5
Random	No Attack		*.0736*	*.1570*	*.2301*	*.2885*	*.4588*	*.5590*	*.7660*	*.8987*	*.9419*	*.0612*	*.1130*	*.2216*
	Categorical	*Cosine*	.0745	.1576	.2311	.2804	.4575	.5687	.7837	.9014	**.9439**	.0802	.1324	.1653
		Katz	.0808	.1698	.2441	.2862	.4610	.5691	.7885	.9021	.9418	.0808	.1105	.1812
		Exclusivity	**.0816**	**.1703**	**.2456**	**.2915**	.4635	.5707	.7897	.8993	.9427	.0886	.1479	**.2417**
	Ontological	*Cosine*	.0709	.1503	.2252	.2748	.4483	.5634	.7720	.8979	.9423	.0561	**.1493**	.1926
		Katz	.0774	.1622	.2355	.2837	.4592	.5670	.7845	.9021	.9416	.0751	.1392	.1857
		Exclusivity	.0766	.1619	.2349	.2848	.4602	.5686	.7846	.9010	.9433	**.1091**	.0999	.2240
	Factual	*Cosine*	.0740	.1558	.2280	.2786	.4528	.5642	.7835	.9023	.9419	.0676	.1009	.1285
		Katz	.0760	.1591	.2319	.2823	.4570	.5662	.7839	.9015	.9417	.0685	.1366	.1823
		Exclusivity	.0793	.1672	.2425	.2890	**.4646**	**.5722**	**.7888**	**.9029**	.9434	.0921	.1034	.2143
Average	No Attack		*.0857*	*.1994*	*.2863*	*.3170*	*.5085*	*.6070*	*.8043*	*.9140*	*.9500*	*.0416*	*.0670*	*.1362*
	Categorical	*Cosine*	.0864	.1967	.2823	.3060	.5115	.6202	.8128	.9127	.9502	.0634	.0950	.1316
		Katz	.0940	**.2094**	**.2922**	.3136	.5133	.6136	.8149	.9132	.9486	.0630	.1031	.1119
		Exclusivity	**.0941**	.2074	.2888	**.3185**	**.5142**	.6142	.8165	.9128	.9502	.0482	.0586	.1548
	Ontological	*Cosine*	.0849	.1954	.2805	.3073	.5126	**.6207**	.8114	.9163	**.9509**	**.0906**	**.1248**	**.1569**
		Katz	.0898	.2021	.2845	.3096	.5107	.6143	.8168	.9135	.9491	.0816	.1171	.1108
		Exclusivity	.0890	.2020	.2842	.3119	.5119	.6165	.8121	.9145	.9489	.0285	.0599	.0947
	Factual	*Cosine*	.0868	.1989	.2806	.3073	.5112	.6185	.8163	**.9166**	.9471	.0362	.0851	.1222
		Katz	.0892	.2016	.2844	.3098	.5110	.6158	**.8189**	.9139	.9473	.0588	.0849	.1040
		Exclusivity	.0912	.2049	.2872	.3152	.5131	.6131	.8166	.9138	.9482	.0502	.0746	.0882
BandWagon	No Attack		*.0817*	*.1319*	*.1881*	*.2640*	*.3834*	*.4694*	*.6000*	*.7656*	*.8435*	*.0100*	*.0105*	*.0061*
	Categorical	*Cosine*	.0763	.1234	.1752	.2641	.3801	.4632	.5918	**.7661**	.8429	**.0107**	.0077	.0074
		Katz	.0794	.1266	.1800	**.2647**	.3821	.4648	.5896	.7596	.8422	.0103	.0080	**.0094**
		Exclusivity	.0758	.1227	.1745	.2640	.3818	.4646	.5835	.7590	.8435	.0067	.0054	.0068
	Ontological	*Cosine*	.0758	.1227	.1745	.2626	.3798	.4637	.5904	.7619	.8433	.0064	.0056	.0049
		Katz	.0792	.1257	.1779	.2636	.3802	.4637	.5820	.7642	**.8447**	.0051	.0027	.0077
		Exclusivity	.0776	.1249	.1770	.2633	.3815	.4643	.5979	.7611	.8413	.0057	.0047	.0052
	Factual	*Cosine*	.0738	.1190	.1714	.2632	.3784	.4623	**.6001**	.7634	.8408	.0057	.0044	.0063
		Katz	.0776	.1239	.1771	.2641	.3801	.4630	.5833	.7602	.8415	.0026	.0083	.0036
		Exclusivity	.0792	.1272	.1796	.2638	.3813	.4642	.5948	.7590	.8405	.0051	.0054	.0227

We underline the results with a p-value greater than 0.05 using a paired-t-test statistical significance test.

RQ1: What is the impact of relatedness-based measures and publicly available semantic information? Let us consider the experiments on `LibraryThing`. We can observe that the adoption of graph-based relatedness generally leads to an attack efficacy improvement over the baseline, which adopts the cosine similarity metric. For instance, the random attack largely benefits from the topological information. The same happens for `Yahoo!Movies` too in Table 5. Indeed, HR for < User-kNN, Random, Categorical, Katz> is 10% better than the baseline, i.e., 0.3725 vs. 0.3512. Beyond random attacks, we can observe some general trends also for informed attacks. In detail, Table 4 (`LibraryThing`), we note that categorical information improves both User-kNN and Item-kNN. It is worth noticing that the same consideration does not hold for latent factor-based models. MF and NeuMF suit better cosine vector similarity. This phenomenon is probably due to the significant difference in how the two recommendation families exploit the additional information.

Finally, we can focus on the *BandWagon* attack. In that case, the attack already exploits the most influential knowledge source for collaborative filtering algorithms: popularity. It follows that the integration with other knowledge sources, e.g., $\mathcal{KG}s$, does not provide any significant improvement. However, the influence of popularity is so high in this attack that the final recommendation lists are subject to a strong popularity bias [88]. Indeed, adding fake profiles with the maximum ratings, e.g., 5 in `Yahoo!Movies` and 10 in `LibraryThing`,

Table 5

Hit Ratio (HR) result values evaluated on top-10 recommendation lists for the `Yahoo!Movies` dataset. The usage of *Cosine* similarity is the baseline approach proposed in *SAShA*.

			User-kNN			**Item-kNN**			**MF**			**NeuMF**		
Attack	**Feature Type**	**Similarity**	1	2.5	5	1	2.5	5	1	2.5	5	1	2.5	5
Random	No Attack		*.1927*	*.3624*	*.4461*	*.3260*	*.5099*	*.6011*	*.4108*	*.5857*	*.7043*	*.0247*	*.0221*	*.0700*
	Categorical	*Cosine*	.1869	.3512	.4277	.3163	.4980	.5886	.4084	.5720	.6648	.0018	.0127	.0464
		Katz	.1912	.3725	.4559	.3429	.5270	.6098	.4244	.6029	.7049	.0223	.0317	**.0891**
		Exclusivity	.1968	.3712	.4533	.3394	.5233	.6072	.4272	.6011	.7023	.0171	.0516	.0544
	Ontological	*Cosine*	.1730	.3353	.4163	.2994	.4793	.5726	.3916	.5513	.6407	.0030	.0051	.0118
		Katz	.1766	.3547	.4337	.3224	.5046	.5904	.4029	.5698	.6638	.0106	.0191	.0386
		Exclusivity	**.2101**	**.3898**	**.4706**	.3532	**.5442**	**.6243**	**.4450**	**.6328**	**.7376**	.0242	**.0567**	.0515
	Factual	*Cosine*	.1881	.3501	.4289	.3149	.4933	.5840	.4087	.5665	.6590	.0188	.0115	.0365
		Katz	.2094	.3869	.4703	**.3545**	.5398	.6213	.4442	.6272	.7371	**.0368**	.0507	.0269
		Exclusivity	.2055	.3799	.4632	.3479	.5317	.6178	.4361	.6142	.7187	.0176	.0402	.0430
Average	No Attack		*.2293*	*.4117*	*.4918*	*.3758*	*.5759*	*.6564*	*.4900*	*.6824*	*.7849*	*.0033*	*.0044*	*.0236*
	Categorical	*Cosine*	.2581	**.4296**	.4972	.3955	**.5953**	.6689	.5326	.7255	.8076	.0017	**.0383**	.0029
		Katz	.2319	.4142	.4917	.3882	.5773	.6542	.4889	.6777	.7716	.0015	.0064	**.0272**
		Exclusivity	.2277	.4026	.4845	.3752	.5698	.6493	.4813	.6658	.7624	.0064	.0014	.0087
	Ontological	*Cosine*	**.2584**	.4264	.4953	**.4019**	.5952	**.6704**	**.5457**	**.7315**	**.8128**	.0043	.0018	.0111
		Katz	.2406	.4209	.4964	.3940	.5877	.6615	.5131	.7093	.7950	.0040	.0022	.0098
		Exclusivity	.2196	.3965	.4771	.3623	.5531	.6337	.4552	.6401	.7347	**.0099**	.0348	.0205
	Factual	*Cosine*	.2573	.4290	**.4960**	.3882	.5884	.6634	.5353	.7256	.8009	.0026	.0055	.0054
		Katz	.2293	.4101	.4910	.3736	.5608	.6414	.4746	.6559	.7511	.0073	.0047	.0231
		Exclusivity	.2311	.4075	.4894	.3706	.5661	.6467	.4809	.6661	.7602	.0042	.0070	.0194
BandWagon	No Attack		*.0996*	*.2418*	*.3556*	*.2427*	*.3764*	*.4691*	*.2357*	*.3606*	*.4320*	*.0010*	*.0026*	*.0025*
	Categorical	*Cosine*	.1020	.2544	**.3634**	.2453	.3831	.4748	.2536	.3909	.4662	.0010	.0208	.0010
		Katz	.0981	.2412	.3495	.2383	.3676	.4546	.2300	.3540	.4248	.0017	.0022	.0077
		Exclusivity	.0926	.2357	.3476	.2378	.3670	.4562	.2248	.3472	.4150	.0009	**.0094**	.0026
	Ontological	*Cosine*	.1039	**.2632**	.3606	.2460	**.3853**	**.4786**	**.2726**	**.4080**	**.4798**	.0045	.0060	.0009
		Katz	.0958	.2476	.3528	.2412	.3754	.4652	.2253	.3602	.4376	.0009	.0023	.0012
		Exclusivity	.0941	.2227	.3346	.2289	.3528	.4402	.2092	.3191	.3885	.0030	.0022	.0054
	Factual	*Cosine*	**.1050**	.2562	.3614	**.2476**	.3814	.4734	.2506	.3890	.4625	.0133	.0043	.0004
		Katz	.0930	.2302	.3460	.2295	.3569	.4461	.2178	.3399	.4064	**.0255**	.0028	**.0115**
		Exclusivity	.0926	.2360	.3515	.2345	.3616	.4504	.2309	.3446	.4137	.0023	.0012	.0014

We underline the results with a p-value greater than 0.05 *using a paired-t-test statistical significance test.*

placed on the most popular/rated items that will form the $\mathcal{I}_S$ (see Table 1) will amplify, even more, the probability that these items will be recommended in the highest positions of top-N recommendation lists making ineffective the adversaries' pushing goal toward the target items. As a consequence, it even prevents the attacked recommendation system from suggesting the target item. All the experimental datasets and all the recommendation models clearly show this effect.

Another aspect that we want to underline is that increasing the number of fake profiles injected into the systems unleashes the potential of different semantic knowledge types. Let us take as an example the <`LibraryThing`, Average, MF>. With 1% injected fake profiles, we observe the best results with Factual knowledge and *Katz* centrality. With 2%, the best results are with Factual knowledge and cosine similarity. Finally, with 5%, the best results come with Ontological knowledge and cosine similarity. This behavior suggests that the graph-based similarities have a big impact even in a very sparse scenario. In contrast, with the increase of fake profiles, the cosine similarity starts leveraging interesting correlations. On the other dimension, the factual information is massive by nature, and it is crucial in sparse scenarios. However, when the number of fake profiles increases, the knowledge at a higher level of abstraction (Categorical and Ontological) finds its way to improve the attack efficacy further.

RQ2: What is the most impactful type of semantic information? We start focusing on Categorical knowledge. The experiments on `LibraryThing` show that Exclusivity is probably the relatedness that best suits this information type. However, the results are not that clear for the Yahoo!Movies dataset. This behavior suggests that semantic information type and relatedness are not the only members of the equation. Indeed, the extension and the quality of the item descriptions seem to have a role. Afterward, we focus on Ontological information. Here, we can draw a general consideration since, for both datasets, it is the cosine similarity metric that leads to the best results. Lastly, Factual information respects all the general remarks we have drawn before showing that relatedness is a better source of adversaries' knowledge to perform more effective attacks.

In detail, we found that with low-knowledge attacks, the best relatedness is *Exclusivity* for LibraryThing and *Katz* for Yahoo!Movies. With informed attacks, the best relatedness metric is the cosine similarity. However, for the sake of electing a similarity that better suits Factual information, we can note that *Exclusivity* generally leads to better results with LibraryThing.

RQ3: What are the most vulnerable recommendation models> Since the neighborhood-based models directly exploit a similarity to compute the recommendation lists, they are the privileged victim models to effectively alter the recommendation performance. Slightly more robust are latent factor models on which the new semantic attacks will still produce an improvement in the attacker's performance. Finally, the most robust model is NeuMF. This result is probably due to the nonlinearity of NeuMF that helps the model avoid learning from the pretended profiles. In detail, the neural network may learn more sophisticated correlations that the other models do not capture. We believe that this ability deserves specific further investigation since it may lead to developing a new line of research on Deep Learning-based semantics-aware attacks that might exploit non-linear item-item similarities to build more impactful attack methods.

6. Conclusions

In this work, we verified how the adoption of stronger semantic similarity measures of structured and freely accessible knowledge can further improve malicious agents' ability to attack a recommendation platform. Starting from the state-of-the-art semantics-aware shilling attacks (*SAShA*), this work investigated the impact of graph-based metrics (*Katz* centrality and *Exclusivity*-based relatedness) and semantic information type. As a result, we verified that: (1) the adoption of structured knowledge generally improves by a large margin the attacker's performance, (2) the graph-based metrics can efficiently deal with very sparse scenarios capturing similarities that are otherwise imperceptible, (3) the type of semantic information to feed the attacking system with has a significant function in enhancing the adversaries' effectiveness, (4) neural models are the sole recommendation techniques to be more robust to semantic attacks. The latter finding suggests that there is still room for improvements in the semantics-aware attacks investigating Deep Learning-based semantic attacks.

References

[1] G. Linden, B. Smith, J. York, Amazon.com recommendations: Item-to-item collaborative filtering, IEEE Internet Computing 7 (2003) 76–80. URL: https://doi.org/10.1109/MIC.2003.1167344. doi:10.1109/MIC.2003.1167344.

[2] I. Gunes, C. Kaleli, A. Bilge, H. Polat, Shilling attacks against recommender systems: a comprehensive survey, Artif. Intell. Rev. 42 (2014) 767–799.

[3] R. Burke, M. P. O'Mahony, N. J. Hurley, Robust collaborative recommendation, in: Recommender Systems Handbook, Springer, 2015, pp. 961–995.

[4] S. K. Lam, J. Riedl, Shilling recommender systems for fun and profit, in: WWW, ACM, 2004, pp. 393–402.

[5] B. Mobasher, R. D. Burke, R. Bhaumik, C. Williams, Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness, ACM Trans. Internet Techn. 7 (2007) 23.

[6] K. Chen, P. P. K. Chan, F. Zhang, Q. Li, Shilling attack based on item popularity and rated item correlation against collaborative filtering, Int. J. Machine Learning & Cybernetics 10 (2019) 1833–1845.

[7] V. W. Anelli, Y. Deldjoo, T. D. Noia, E. D. Sciascio, F. A. Merra, Sasha: Semantic-aware shilling attacks on recommender systems exploiting knowledge graphs, in: The Semantic Web - 17th International Conference, ESWC 2020, Heraklion, Crete, Greece, May 31-June 4, 2020, Proceedings, volume 12123 of *Lecture Notes in Computer Science*, Springer, 2020, pp. 307–323. URL: https://doi.org/10.1007/978-3-030-49461-2_18. doi:10.1007/978-3-030-49461-2_18.

[8] F. Ricci, L. Rokach, B. Shapira, Introduction to recommender systems handbook, in: F. Ricci, L. Rokach, B. Shapira, P. B. Kantor (Eds.), Recommender Systems Handbook, Springer, 2011, pp. 1–35. URL: https://doi.org/10.1007/978-0-387-85820-3_1. doi:10.1007/978-0-387-85820-3_1.

[9] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions, IEEE Trans. Knowl. Data Eng. 17 (2005) 734–749. URL: https://doi.org/10.1109/TKDE.2005.99. doi:10.1109/TKDE.2005.99.

[10] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, J. Riedl, Grouplens: An open architecture for collaborative filtering of netnews, in: CSCW, ACM, 1994, pp. 175–186.

[11] Y. Koren, Factor in the neighbors: Scalable and accurate collaborative filtering, TKDD 4 (2010) 1:1–1:24.

[12] B. M. Sarwar, G. Karypis, J. A. Konstan, J. Riedl, Item-based collaborative filtering recommendation algorithms, in: V. Y. Shen, N. Saito, M. R. Lyu, M. E. Zurko (Eds.), Proceedings of the Tenth International World Wide Web Conference, WWW 10, Hong Kong, China, May 1-5, 2001, ACM, 2001, pp. 285–295. URL: https://doi.org/10.1145/371920.372071. doi:10.1145/371920.372071.

[13] Y. Koren, R. M. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, IEEE Computer 42 (2009) 30–37.

[14] X. Ning, G. Karypis, Sparse linear methods with side information for top-n recommendations, in: P. Cunningham, N. J. Hurley, I. Guy, S. S. Anand (Eds.), Sixth ACM Conference on Recommender Systems, RecSys '12, Dublin, Ireland, September 9-13, 2012, ACM, 2012, pp. 155–162. URL: https://doi.org/10.1145/2365952.2365983. doi:10.1145/2365952.2365983.

[15] L. Backstrom, J. Leskovec, Supervised random walks: predicting and recommending links in social networks, in: Proceedings of the Forth International Conference on Web Search

and Web Data Mining, WSDM 2011, Hong Kong, China, February 9-12, 2011, 2011, pp. 635–644.

[16] Y. Deldjoo, M. F. Dacrema, M. G. Constantin, H. Eghbal-zadeh, S. Cereda, M. Schedl, B. Ionescu, P. Cremonesi, Movie genome: alleviating new item cold start in movie recommendation, User Model. User-Adapt. Interact. 29 (2019) 291–343.

[17] V. W. Anelli, V. Bellini, T. Di Noia, W. L. Bruna, P. Tomeo, E. Di Sciascio, An analysis on time- and session-aware diversification in recommender systems, in: UMAP, ACM, 2017, pp. 270–274.

[18] Q. Guo, F. Zhuang, C. Qin, H. Zhu, X. Xie, H. Xiong, Q. He, A survey on knowledge graph-based recommender systems, IEEE Transactions on Knowledge & Data Engineering (5555) 1–1. doi:10.1109/TKDE.2020.3028705.

[19] I. Fernández-Tobías, I. Cantador, P. Tomeo, V. W. Anelli, T. D. Noia, Addressing the user cold start with cross-domain collaborative filtering: exploiting item metadata in matrix factorization, User Model. User Adapt. Interact. 29 (2019) 443–486. URL: https://doi.org/10.1007/s11257-018-9217-6. doi:10.1007/s11257-018-9217-6.

[20] Y. Huo, D. F. Wong, L. M. Ni, L. S. Chao, J. Zhang, Knowledge modeling via contextualized representations for lstm-based personalized exercise recommendation, Inf. Sci. 523 (2020) 266–278. URL: https://doi.org/10.1016/j.ins.2020.03.014. doi:10.1016/j.ins.2020.03.014.

[21] M. Hildebrandt, S. S. Sunder, S. Mogoreanu, M. Joblin, A. Mehta, I. Thon, V. Tresp, A recommender system for complex real-world applications with nonlinear dependencies and knowledge graph context, in: ESWC, volume 11503 of *Lecture Notes in Computer Science*, Springer, 2019, pp. 179–193.

[22] V. W. Anelli, T. Di Noia, 2nd workshop on knowledge-aware and conversational recommender systems - kars, in: CIKM, ACM, 2019, pp. 3001–3002.

[23] V. W. Anelli, P. Basile, D. G. Bridge, T. D. Noia, P. Lops, C. Musto, F. Narducci, M. Zanker, Knowledge-aware and conversational recommender systems, in: S. Pera, M. D. Ekstrand, X. Amatriain, J. O'Donovan (Eds.), Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018, Vancouver, BC, Canada, October 2-7, 2018, ACM, 2018, pp. 521–522. URL: https://doi.org/10.1145/3240323.3240338. doi:10.1145/3240323.3240338.

[24] E. Palumbo, D. Monti, G. Rizzo, R. Troncy, E. Baralis, entity2rec: Property-specific knowledge graph embeddings for item recommendation, Expert Syst. Appl. 151 (2020) 113235. URL: https://doi.org/10.1016/j.eswa.2020.113235. doi:10.1016/j.eswa.2020.113235.

[25] J. Li, Z. Xu, Y. Tang, B. Zhao, H. Tian, Deep hybrid knowledge graph embedding for top-n recommendation, in: G. Wang, X. Lin, J. A. Hendler, W. Song, Z. Xu, G. Liu (Eds.), Web Information Systems and Applications - 17th International Conference, WISA 2020, Guangzhou, China, September 23-25, 2020, Proceedings, volume 12432 of *Lecture Notes in Computer Science*, Springer, 2020, pp. 59–70. URL: https://doi.org/10.1007/978-3-030-60029-7_6. doi:10.1007/978-3-030-60029-7_6.

[26] M. Nayyeri, S. Vahdati, X. Zhou, H. S. Yazdi, J. Lehmann, Embedding-based recommendations on scholarly knowledge graphs, in: A. Harth, S. Kirrane, A. N. Ngomo, H. Paulheim, A. Rula, A. L. Gentile, P. Haase, M. Cochez (Eds.), The Semantic Web - 17th International Conference, ESWC 2020, Heraklion, Crete, Greece, May 31-June 4, 2020, Proceedings,

volume 12123 of *Lecture Notes in Computer Science*, Springer, 2020, pp. 255–270. URL: https://doi.org/10.1007/978-3-030-49461-2_15. doi:10.1007/978-3-030-49461-2_15.

[27] Y. Zhang, X. Xu, H. Zhou, Y. Zhang, Distilling structured knowledge into embeddings for explainable and accurate recommendation, in: J. Caverlee, X. B. Hu, M. Lalmas, W. Wang (Eds.), WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020, ACM, 2020, pp. 735–743. URL: https://doi.org/10.1145/3336191.3371790. doi:10.1145/3336191.3371790.

[28] C. Ni, K. S. Liu, N. Torzec, Layered graph embedding for entity recommendation using wikipedia in the yahoo! knowledge graph, in: A. E. F. Seghrouchni, G. Sukthankar, T. Liu, M. van Steen (Eds.), Companion of The 2020 Web Conference 2020, Taipei, Taiwan, April 20-24, 2020, ACM / IW3C2, 2020, pp. 811–818. URL: https://doi.org/10.1145/3366424.3383570. doi:10.1145/3366424.3383570.

[29] P. Ristoski, J. Rosati, T. D. Noia, R. D. Leone, H. Paulheim, Rdf2vec: RDF graph embeddings and their applications, Semantic Web 10 (2019) 721–752. URL: https://doi.org/10.3233/SW-180317. doi:10.3233/SW-180317.

[30] T. Dettmers, P. Minervini, P. Stenetorp, S. Riedel, Convolutional 2d knowledge graph embeddings, in: S. A. McIlraith, K. Q. Weinberger (Eds.), Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, AAAI Press, 2018, pp. 1811–1818. URL: https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17366.

[31] V. W. Anelli, T. Di Noia, E. Di Sciascio, A. Ragone, J. Trotta, How to make latent factors interpretable by feeding factorization machines with knowledge graphs, in: C. Ghidini, O. Hartig, M. Maleshkova, V. Svátek, I. F. Cruz, A. Hogan, J. Song, M. Lefrançois, F. Gandon (Eds.), The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part I, volume 11778 of *Lecture Notes in Computer Science*, Springer, 2019, pp. 38–56. URL: https://doi.org/10.1007/978-3-030-30793-6_3. doi:10.1007/978-3-030-30793-6_3.

[32] G. He, J. Li, W. X. Zhao, P. Liu, J. Wen, Mining implicit entity preference from user-item interaction data for knowledge graph completion via adversarial learning, in: Y. Huang, I. King, T. Liu, M. van Steen (Eds.), WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020, ACM / IW3C2, 2020, pp. 740–751. URL: https://doi.org/10.1145/3366423.3380155. doi:10.1145/3366423.3380155.

[33] A. Bordes, N. Usunier, A. García-Durán, J. Weston, O. Yakhnenko, Translating embeddings for modeling multi-relational data, in: C. J. C. Burges, L. Bottou, Z. Ghahramani, K. Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States, 2013, pp. 2787–2795. URL: https://proceedings.neurips.cc/paper/2013/hash/1cecc7a77928ca8133fa24680a88d2f9-Abstract.html.

[34] Y. Cao, X. Wang, X. He, Z. Hu, T. Chua, Unifying knowledge graph learning and recommendation: Towards a better understanding of user preferences, in: L. Liu, R. W. White, A. Mantrach, F. Silvestri, J. J. McAuley, R. Baeza-Yates, L. Zia (Eds.), The World Wide Web

Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019, ACM, 2019, pp. 151–161. URL: https://doi.org/10.1145/3308558.3313705. doi:10.1145/3308558.3313705.

[35] G. Piao, J. G. Breslin, Transfer learning for item recommendations and knowledge graph completion in item related domains via a co-factorization model, in: A. Gangemi, R. Navigli, M. Vidal, P. Hitzler, R. Troncy, L. Hollink, A. Tordai, M. Alam (Eds.), The Semantic Web - 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3-7, 2018, Proceedings, volume 10843 of *Lecture Notes in Computer Science*, Springer, 2018, pp. 496–511. URL: https://doi.org/10.1007/978-3-319-93417-4_32. doi:10.1007/978-3-319-93417-4_32.

[36] M. S. Schlichtkrull, T. N. Kipf, P. Bloem, R. van den Berg, I. Titov, M. Welling, Modeling relational data with graph convolutional networks, in: A. Gangemi, R. Navigli, M. Vidal, P. Hitzler, R. Troncy, L. Hollink, A. Tordai, M. Alam (Eds.), The Semantic Web - 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3-7, 2018, Proceedings, volume 10843 of *Lecture Notes in Computer Science*, Springer, 2018, pp. 593–607. URL: https://doi.org/10.1007/978-3-319-93417-4_38. doi:10.1007/978-3-319-93417-4_38.

[37] C. Shang, Y. Tang, J. Huang, J. Bi, X. He, B. Zhou, End-to-end structure-aware convolutional networks for knowledge base completion, in: The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019, AAAI Press, 2019, pp. 3060–3067. URL: https://doi.org/10.1609/aaai.v33i01.33013060. doi:10.1609/aaai.v33i01.33013060.

[38] X. Wang, X. He, Y. Cao, M. Liu, T. Chua, KGAT: knowledge graph attention network for recommendation, in: A. Teredesai, V. Kumar, Y. Li, R. Rosales, E. Terzi, G. Karypis (Eds.), Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019, ACM, 2019, pp. 950–958. URL: https://doi.org/10.1145/3292500.3330989. doi:10.1145/3292500.3330989.

[39] Q. Zhang, P. Hao, J. Lu, G. Zhang, Cross-domain recommendation with semantic correlation in tagging systems, in: International Joint Conference on Neural Networks, IJCNN 2019 Budapest, Hungary, July 14-19, 2019, IEEE, 2019, pp. 1–8. URL: https://doi.org/10.1109/IJCNN.2019.8852049. doi:10.1109/IJCNN.2019.8852049.

[40] T. Köllmer, E. Berndl, T. Weißgerber, P. Aichroth, H. Kosch, A workflow for cross media recommendations based on linked data analysis, in: R. Troncy, R. Verborgh, L. J. B. Nixon, T. Kurz, K. Schlegel, M. V. Sande (Eds.), Joint Proceedings of the 4th International Workshop on Linked Media and the 3rd Developers Hackshop co-located with the 13th Extended Semantic Web Conference ESWC 2016, Heraklion, Crete, Greece, May 30, 2016, volume 1615 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2016. URL: http://ceur-ws.org/Vol-1615/limePaper1.pdf.

[41] V. W. Anelli, V. Bellini, T. D. Noia, E. D. Sciascio, Knowledge-aware interpretable recommender systems, in: I. Tiddi, F. Lécué, P. Hitzler (Eds.), Knowledge Graphs for eXplainable Artificial Intelligence: Foundations, Applications and Challenges, volume 47 of *Studies on the Semantic Web*, IOS Press, 2020, pp. 101–124. URL: https://doi.org/10.3233/SSW200014. doi:10.3233/SSW200014.

[42] V. W. Anelli, T. Di Noia, E. Di Sciascio, A. Ragone, J. Trotta, Semantic interpretation of

top-n recommendations, IEEE Transactions on Knowledge and Data Engineering (2020) 1–1. doi:10.1109/TKDE.2020.3010215.

[43] Z. Yang, S. Dong, Hagerec: Hierarchical attention graph convolutional network incorporating knowledge graph for explainable recommendation, Knowl. Based Syst. 204 (2020) 106194. URL: https://doi.org/10.1016/j.knosys.2020.106194. doi:10.1016/j.knosys.2020.106194.

[44] X. Wang, D. Wang, C. Xu, X. He, Y. Cao, T.-S. Chua, Explainable reasoning over knowledge graphs for recommendation, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, 2019, pp. 5329–5336.

[45] L. Sang, M. Xu, S. Qian, X. Wu, Knowledge graph enhanced neural collaborative recommendation, Expert Syst. Appl. 164 (2021) 113992. URL: https://doi.org/10.1016/j.eswa.2020.113992. doi:10.1016/j.eswa.2020.113992.

[46] T. Wang, D. Shi, Z. Wang, S. Xu, H. Xu, Mrp2rec: Exploring multiple-step relation path semantics for knowledge graph-based recommendations, IEEE Access 8 (2020) 134817–134825. URL: https://doi.org/10.1109/ACCESS.2020.3011279. doi:10.1109/ACCESS.2020.3011279.

[47] D. Shi, T. Wang, H. Xing, H. Xu, A learning path recommendation model based on a multidimensional knowledge graph framework for e-learning, Knowl. Based Syst. 195 (2020) 105618. URL: https://doi.org/10.1016/j.knosys.2020.105618. doi:10.1016/j.knosys.2020.105618.

[48] H. Wang, M. Zhao, X. Xie, W. Li, M. Guo, Knowledge graph convolutional networks for recommender systems, in: L. Liu, R. W. White, A. Mantrach, F. Silvestri, J. J. McAuley, R. Baeza-Yates, L. Zia (Eds.), The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019, ACM, 2019, pp. 3307–3313. URL: https://doi.org/10.1145/3308558.3313417. doi:10.1145/3308558.3313417.

[49] H. Wang, F. Zhang, J. Wang, M. Zhao, W. Li, X. Xie, M. Guo, Ripplenet: Propagating user preferences on the knowledge graph for recommender systems, in: A. Cuzzocrea, J. Allan, N. W. Paton, D. Srivastava, R. Agrawal, A. Z. Broder, M. J. Zaki, K. S. Candan, A. Labrinidis, A. Schuster, H. Wang (Eds.), Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018, ACM, 2018, pp. 417–426. URL: https://doi.org/10.1145/3269206.3271739. doi:10.1145/3269206.3271739.

[50] T. Di Noia, V. C. Ostuni, P. Tomeo, E. Di Sciascio, Sprank: Semantic path-based ranking for top-*N* recommendations using linked open data, ACM TIST 8 (2016) 9:1–9:34. URL: https://doi.org/10.1145/2899005. doi:10.1145/2899005.

[51] V. W. Anelli, T. D. Noia, P. Lops, E. D. Sciascio, Feature factorization for top-n recommendation: From item rating to features relevance, in: Y. Zheng, W. Pan, S. S. Sahebi, I. Fernández (Eds.), Proceedings of the 1st Workshop on Intelligent Recommender Systems by Knowledge Transfer & Learning co-located with ACM Conference on Recommender Systems (RecSys 2017), Como, Italy, August 27, 2017, volume 1887 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2017, pp. 16–21. URL: http://ceur-ws.org/Vol-1887/paper3.pdf.

[52] T. D. Noia, R. Mirizzi, V. C. Ostuni, D. Romito, M. Zanker, Linked open data to support content-based recommender systems, in: V. Presutti, H. S. Pinto (Eds.), I-SEMANTICS 2012 - 8th International Conference on Semantic Systems, I-SEMANTICS '12, Graz, Austria,

September 5-7, 2012, ACM, 2012, pp. 1–8. URL: https://doi.org/10.1145/2362499.2362501. doi:10.1145/2362499.2362501.

[53] X. Yu, X. Ren, Y. Sun, Q. Gu, B. Sturt, U. Khandelwal, B. Norick, J. Han, Personalized entity recommendation: a heterogeneous information network approach, in: WSDM, ACM, 2014, pp. 283–292.

[54] L. Gao, H. Yang, J. Wu, C. Zhou, W. Lu, Y. Hu, Recommendation with multi-source heterogeneous information, in: IJCAI, ijcai.org, 2018, pp. 3378–3384.

[55] H. Wang, F. Zhang, X. Xie, M. Guo, DKN: deep knowledge-aware network for news recommendation, in: WWW, ACM, 2018, pp. 1835–1844.

[56] T. Di Noia, C. Magarelli, A. Maurino, M. Palmonari, A. Rula, Using ontology-based data summarization to develop semantics-aware recommender systems, in: ESWC, volume 10843 of *Lecture Notes in Computer Science*, Springer, 2018, pp. 128–144.

[57] M. P. O'Mahony, N. J. Hurley, N. Kushmerick, G. C. M. Silvestre, Collaborative recommendation: A robustness analysis, ACM Trans. Internet Techn. 4 (2004) 344–377.

[58] C. C. Aggarwal, Attack-resistant recommender systems, in: Recommender Systems, Springer, 2016, pp. 385–410.

[59] R. Bhaumik, C. Williams, B. Mobasher, R. Burke, Securing collaborative filtering against malicious attacks through anomaly detection, in: Proceedings of the 4th Workshop on Intelligent Techniques for Web Personalization (ITWP'06), Boston, volume 6, 2006, p. 10.

[60] B. Mobasher, R. Burke, R. Bhaumik, C. Williams, Effective attack models for shilling item-based collaborative filtering systems, in: Proceedings of the WebKDD Workshop, Citeseer, 2005, pp. 13–23.

[61] T. D. Noia, D. Malitesta, F. A. Merra, Taamr: Targeted adversarial attack against multimedia recommender systems, in: DSN Workshops, IEEE, 2020, pp. 1–8.

[62] M. P. O'Mahony, N. J. Hurley, G. C. M. Silvestre, Recommender systems: Attack types and strategies, in: AAAI, AAAI Press / The MIT Press, 2005, pp. 334–339.

[63] Y. Deldjoo, T. D. Noia, F. A. Merra, Assessing the impact of a user-item collaborative attack on class of users, in: ImpactRS@RecSys, volume 2462 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2019.

[64] Y. Deldjoo, T. D. Noia, E. D. Sciascio, F. A. Merra, How dataset characteristics affect the robustness of collaborative recommendation models, in: Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020, ACM, 2020, pp. 951–960. URL: https://doi.org/10.1145/3397271.3401046. doi:10.1145/3397271.3401046.

[65] J. Cao, Z. Wu, B. Mao, Y. Zhang, Shilling attack detection utilizing semi-supervised learning method for collaborative recommender system, World Wide Web 16 (2013) 729–748.

[66] W. Zhou, J. Wen, Q. Xiong, M. Gao, J. Zeng, Svm-tia a shilling attack detection method based on svm and target item analysis in recommender systems, Neurocomputing 210 (2016) 197–205.

[67] W. Zhou, J. Wen, Q. Qu, J. Zeng, T. Cheng, Shilling attack detection for recommender systems based on credibility of group users and rating time series, PloS one 13 (2018) e0196533.

[68] M. Aktukmak, Y. Yilmaz, I. Uysal, Quick and accurate attack detection in recommender systems through user attributes, in: RecSys, ACM, 2019, pp. 348–352.

[69] Y. Cai, D. Zhu, Trustworthy and profit: A new value-based neighbor selection method in recommender systems under shilling attacks, Decision Support Systems 124 (2019) 113112.

[70] B. Li, Y. Wang, A. Singh, Y. Vorobeychik, Data poisoning attacks on factorization-based collaborative filtering, in: NIPS, 2016, pp. 1885–1893.

[71] K. Christakopoulou, A. Banerjee, Adversarial attacks on an oblivious recommender, in: RecSys, ACM, 2019, pp. 322–330.

[72] M. Fang, N. Z. Gong, J. Liu, Influence function based data poisoning attacks to top-n recommender systems, in: WWW, ACM / IW3C2, 2020, pp. 3019–3025.

[73] Y. Liu, X. Xia, L. Chen, X. He, C. Yang, Z. Zheng, Certifiable robustness to discrete adversarial perturbations for factorization machines, in: SIGIR, ACM, 2020, pp. 419–428.

[74] J. Tang, H. Wen, K. Wang, Revisiting adversarially learned injection attacks against recommender systems, in: Fourteenth ACM Conference on Recommender Systems, 2020, pp. 318–327.

[75] X. He, Z. He, X. Du, T. Chua, Adversarial personalized ranking for recommendation, in: SIGIR, ACM, 2018, pp. 355–364.

[76] F. Yuan, L. Yao, B. Benatallah, Adversarial collaborative neural network for robust recommendation, in: SIGIR, ACM, 2019, pp. 1065–1068.

[77] V. W. Anelli, T. D. Noia, D. Malitesta, F. A. Merra, Assessing perceptual and recommendation mutation of adversarially-poisoned visual recommenders (short paper), in: DP@AI*IA, volume 2776 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2020, pp. 49–56.

[78] V. W. Anelli, Y. Deldjoo, T. DiNoia, F. A. Merra, Adversarial Recommender Systems: Attack, Defense, and Advances, Springer US, New York, NY, 2022, pp. 335–379. URL: https://doi.org/10.1007/978-1-0716-2197-4_9. doi:10.1007/978-1-0716-2197-4_9.

[79] F. A. Merra, V. W. Anelli, T. D. Noia, D. Malitesta, A. C. M. Mancino, Denoise to protect: A method to robustify visual recommenders from adversaries, in: SIGIR, ACM, 2023, pp. 1924–1928.

[80] T. Di Noia, R. Mirizzi, V. C. Ostuni, D. Romito, M. Zanker, Linked open data to support content-based recommender systems, in: Proc. of the 8th Int. Conf. on Semantic Systems, ACM, 2012, pp. 1–8.

[81] L. Katz, A new status index derived from sociometric analysis, Psychometrika 18 (1953) 39–43. URL: https://doi.org/10.1007/BF02289026. doi:10.1007/BF02289026.

[82] I. Hulpus, N. Prangnawarat, C. Hayes, Path-based semantic relatedness on linked data and its use to word and entity disambiguation, in: International Semantic Web Conference (1), volume 9366 of *Lecture Notes in Computer Science*, Springer, 2015, pp. 442–457.

[83] B. P. Nunes, S. Dietze, M. A. Casanova, R. Kawase, B. Fetahu, W. Nejdl, Combining a co-occurrence-based and a semantic measure for entity linking, in: P. Cimiano, Ó. Corcho, V. Presutti, L. Hollink, S. Rudolph (Eds.), The Semantic Web: Semantics and Big Data, 10th International Conference, ESWC 2013, Montpellier, France, May 26-30, 2013. Proceedings, volume 7882 of *Lecture Notes in Computer Science*, Springer, 2013, pp. 548–562. URL: https://doi.org/10.1007/978-3-642-38288-8_37. doi:10.1007/978-3-642-38288-8_37.

[84] B. Mobasher, R. Burke, R. Bhaumik, C. Williams, Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness, ACM Transactions on Internet Technology (TOIT) 7 (2007).

[85] M. P. O'Mahony, N. J. Hurley, G. C. Silvestre, An evaluation of the performance of

collaborative filtering, in: 14th Irish Artificial Intelligence and Cognitive Science (AICS 2003) Conference, Citeseer, 2003.

[86] H. Paulheim, J. Fürnkranz, Unsupervised generation of data mining features from linked open data, in: WIMS, ACM, 2012, pp. 31:1–31:12.

[87] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, T. Chua, Neural collaborative filtering, in: WWW, ACM, 2017, pp. 173–182.

[88] H. Abdollahpouri, R. Burke, B. Mobasher, Controlling popularity bias in learning-to-rank recommendation, in: RecSys, ACM, 2017, pp. 42–46.

7. Appendices

7.1. Multiple hop v.s. single-hop attacks.

The subsequent analysis focuses on the impact of the 1-hop and 2-hops of the $\mathcal{KG}$ exploration.

Analogously to 1-hop definition insection 3.1, we built 2nd-hop features. By continuing the exploration of $\mathcal{KG}$ we retrieve the triples $\omega \xrightarrow{\rho'} \omega'$, where ω is the *object* of a 1st-hop triple and the *subject* of the next triple. The double-hop *predicate* is denoted by ρ' and the *object* is referred as (ω'). Therefore, the overall feature set is defined as $2\text{-}HOP\text{-}F = \{\langle \rho, \omega, \rho', \omega' \rangle \mid \langle i, \rho, \omega, \rho', \omega' \rangle \in \mathcal{KG}$ with $i \in I\}$. Given the current definition, 2nd-hop features also contain heterogeneous predicates (see the previous classification of different kinds of statements). To make it possible to analyze the impact of the kind of semantic information, we consider a 2nd-hop feature as Factual *if and only if* both relations (ρ, and ρ') are Factual. The same holds for the other types of encoded information. In the attacks employing double-hop (2H) features, the strategies evolve as described below:

- **Categorical-2H**, we pick up the features with either `dcterms:subject` or `skos:broader` properties;
- **Ontological-2H**, we select the features containing either `rdf-schema:subClassOf` or `owl:equivalentClass` properties;
- **Factual-2H**, we use the features not selected in the previous two classes.

Note that we did not place any domain-specific categorical/ontological features in the respective lists. To provide a domain-agnostic evaluation, we have treated them as factual features. Table 6 shows the average variation of attack efficacy passing from the adoption of single-hop extracted features to the double-hop extraction for `LibraryThing` and `Yahoo!Movies`.

Analyzing the results of attacks on `Yahoo!Movies` in Table 6, the first and foremost consideration we can draw is that graph-based relatedness measures seem to have no positive impact when exploiting a double-hop exploration. However, it can be observed that those relatedness metrics already achieved impressive results with the first-hop exploration. Hence, further improving the performance is somehow challenging. Indeed, in most cases, we can observe a minimal variation in the double-hop performance. However, in some cases, the attacks witness a more significant decrease, probably due to the injection of some noisy and

Table 6
Variation of Hit Ratio (HR) when using the features extracted from the second hop concerning the first hop for both the `LibraryThing` and `Yahoo!Movies` datasets.

			`LibraryThing`				`Yahoo!Movies`			
Attack	**Feature Type**	**Similarity**	**U-kNN**	**I-kNN**	**MF**	**NeuMF**	**U-kNN**	**I-kNN**	**MF**	**NeuMF**
Random	Categorical	Cosine	-1.28	-1.63	-0.70	-20.07	-0.03	-0.01	-0.01	1.57
		Katz	-0.77	2.05	-0.20	-6.05	-0.11	-0.10	-0.06	-0.47
		Exclusivity	-2.12	0.14	-0.26	-21.09	-0.05	-0.04	-0.02	0.08
	Ontological	Cosine	1.97	0.64	0.35	13.45	0.16	0.12	0.10	1.31
		Katz	-3.00	-0.24	0.10	-38.28	-0.07	-0.07	-0.04	-0.29
		Exclusivity	-4.57	-1.92	-0.47	-46.85	-0.13	-0.09	-0.07	-0.66
	Factual	Cosine	-0.64	-0.62	-0.11	46.94	-0.01	0.02	0.01	-0.62
		Katz	0.93	2.60	0.07	56.47	-0.12	-0.09	-0.07	-0.73
		Exclusivity	-0.33	0.25	-0.39	-29.80	-0.16	-0.11	-0.08	-0.21
Average	Categorical	Cosine	-0.87	-0.86	-0.21	-17.66	-0.03	0.00	-0.01	0.67
		Katz	0.07	2.13	0.02	36.36	0.03	-0.03	0.05	3.81
		Exclusivity	-1.82	-0.09	-0.22	52.37	0.02	-0.02	0.03	-0.69
	Ontological	Cosine	0.47	-0.05	0.22	-8.44	-0.14	-0.12	-0.17	-0.19
		Katz	-3.92	-0.82	-0.52	-70.51	0.07	0.00	0.06	2.94
		Exclusivity	-4.49	-2.26	0.32	152.52	0.07	0.02	0.06	-0.77
	Factual	Cosine	-0.19	0.29	0.06	123.56	-0.04	0.00	-0.04	0.22
		Katz	0.64	1.73	-0.28	13.12	0.01	-0.02	0.04	-0.75
		Exclusivity	0.53	0.87	-0.33	-2.11	0.06	0.03	0.09	-0.17
BandWagon	Categorical	Cosine	-0.02	-0.55	-0.42	-51.24	-0.03	0.00	0.02	-0.01
		Katz	-1.93	-1.01	-0.04	-68.96	-0.06	0.02	0.00	8.87
		Exclusivity	3.25	-0.32	0.07	36.58	0.02	-0.02	0.05	0.07
	Ontological	Cosine	-1.37	-0.10	0.16	49.05	-0.14	-0.08	-0.20	-0.62
		Katz	-5.69	-0.18	2.05	-9.28	0.01	-0.01	0.10	0.78
		Exclusivity	-2.37	-0.45	-0.55	-35.24	-0.02	0.02	0.10	0.61
	Factual	Cosine	1.80	-0.14	-0.32	5.18	-0.07	-0.02	-0.02	-0.91
		Katz	1.57	-0.45	1.00	190.44	0.02	0.05	0.07	-0.90
		Exclusivity	-1.57	-0.61	-1.52	140.00	0.07	0.03	0.08	-0.17

loosely related second-hop features. In general, given the high performance achieved with a single-hop exploration, it seems that it is not worth exploring the second-hop, and thus increasing the computational complexity and introducing the new challenge of loosely-related second-hop features. Beyond graph-based relatedness, we observe that cosine vector similarity almost always shows an improvement when considering second-hop features (particularly with Ontological and Factual information). Finally, we have to observe that, even here, the NeuMF model does not benefit from this new information.

Table 6 also shows the average attack efficacy variation for `LibraryThing`. Here, some of the previously described behaviors are even more evident. In detail, we note that the cosine similarity takes advantage of the second-hop information. In this case, we can also observe *Katz*'s improvement, suggesting that this metric did not have unleashed its full potential with only the first-hop features. Finally, in some cases, the second-hop information also improves informed attacks (reaching a peak of 53% improvement for <Average, Factual, *Exclusivity*>), confirming a less evident trend we found with `Yahoo!Movies`.