Leveraging Semantic Information to Facilitate the Discovery of Underserved Podcasts

Maryam Aziz, Alice Wang, Aasish Pappu, Hugues Bouchard,
Yu Zhao, Benjamin Carterette and Mounia Lalmas
Spotify

ABSTRACT

Podcasts are a popular medium for rapid dissemination of information, entertainment, and casual conversations. Content aggregators are taking an increased interest in recommending podcasts to listeners to help them build larger audiences. With many podcasts released every day, many podcasts that would be of interest to listeners remain underserved by these recommendation systems. In this paper, we study variables related to podcast appeal to listeners selected at random in a large online study, in a production setting, involving more than five million recommendations. We present the results of two observational studies, which suggests that underserved podcast have the potential to grow their audiences. To mitigate the rich-get-richer effect, we propose leveraging semantic information, via means of knowledge graphs, to recommend underserved podcasts to listeners. Finally, we conduct empirical experiments that show our method is effective at recommending underserved podcasts, in comparison to baseline methods that rely on listening behavior.

CCS CONCEPTS

• **Computing methodologies** → **Semantic networks**.

KEYWORDS

semantic technologies, podcasts, long-tail content, recommendation, content discovery

ACM Reference Format:
Maryam Aziz, Alice Wang, Aasish Pappu, Hugues Bouchard, and Yu Zhao, Benjamin Carterette and Mounia Lalmas. 2021. Leveraging Semantic Information to Facilitate the Discovery of Underserved Podcasts. In *Proceedings of the 30th ACM Int'l Conf. on Information and Knowledge Management (CIKM '21), November 1–5, 2021, Virtual Event, Australia.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3459637.3481934

1 INTRODUCTION

Podcasting as a mass-media form is changing the dynamics of entertainment and information exchange. It has become an important part of our social discourse. Creators of podcasts discuss current affairs and evergreen content in various lengths, represent diverse voices, speak different languages, communicate in a spectrum of casual to formal styles and draw audiences from many communities.

To engage with audiences on audio platforms, new podcasts are often served through RSS feeds and sometimes on the creators' personal blogs or websites. To help accelerate podcast consumption, major audio platforms such as Spotify aggregate RSS feeds from independent creators, thus enabling creators to upload their podcasts to be served from a central location. These audio platforms play an important role in surfacing podcasts to listeners through search and recommendation. However, due to the sheer volume of podcast episodes being created every day, creators have to compete to have their voices heard whereas listeners may struggle to find podcasts that fulfill their listening interests or their desire to discover their community, no matter how small. The onus is on audio platforms to facilitate the matching between this ever-growing pool of creators and audiences, irrespective of how niche the community or how novel the podcast content.

In particular, a major opportunity and challenge is how to surface "underserved" podcasts and help audiences discover relevant but up-and-coming creators. Recommender systems in this fast-evolving podcast ecosystem need to consider not only historical user interaction signals, but also information about the podcast creators and their contents. This would mean to strike a balance between promoting popular content and more niche content, *as well as* between users' historical podcast interest and opportunities to discover new creators.

This work explores the underserved podcast challenge through a large-scale industry study involving more than five million recommendations. We consider a podcast to be "underserved" if it has relatively few listeners and if randomly-selected listeners were less likely than average to stream it. For instance, a high-quality podcast on Formula One Racing is enjoyed by a smaller group of people, so recommending it to a randomly-selected group of users will result in less streams compared to a podcast with a broader appeal. Such podcasts would benefit from more strategic recommendations, to match them to users more likely to be interested in them. Our focus is on good quality podcasts and hence we only consider podcasts professionally produced by established publishers.[1]

At first glance, focusing on podcasts with smaller listener counts seems like an instance of the cold-start problem. However, it is distinct because many podcasts we consider underserved have been on the platform for several months, are of high quality, and have a small but significant community of followers. However, they have not built large audiences yet. This makes it a challenge to match these podcasts with interested users. We do not focus on podcasts that are new to the platform, but rather on podcasts

CIKM '21, November 1–5, 2021, Virtual Event, Australia.

ACM ISBN 978-1-4503-8446-9/21/11…$15.00
https://doi.org/10.1145/3459637.3481934

[1] We leave studying the relationship between quality and being underserved for future work.

that are less likely to build larger audiences without the help of a recommender system. In fact, in one of our studies we find that users are more receptive than average to very recent podcasts, so we do not consider such podcasts to be underserved.

We study the characteristics of the podcasts that are less likely to be streamed at random with the belief that as long as these podcasts are of adequate quality more listeners could be found for them. Salganik et al. [25] showed in the context of music that artificially changing a song's download count significantly affects users' decisions as to whether to download a song. Moreover, they show that only a few songs are always/never popular regardless of social signals, and that social factors play a significant role in determining what music becomes popular for many songs. We suspect that underserved podcasts may suffer from a similar problem: many potential listeners not being aware of such podcasts is likely a main factor in them being underserved.

To mitigate the underserved podcast problem we propose to leverage both collaborative and semantic information. We extract semantic signals from podcast contents as well as data about the creators (e.g. publishers). We combine these semantic signals by building a knowledge graph (KG), which is a common way to unify heterogeneous sources of knowledge via typed relations. Knowledge-aware recommendation paradigms have recently gained much success, as they introduce semantic relationships, alleviate data sparsity problems for cold-start items, and can be used to provide explainability and diversity to recommendations [1, 34]. Here, we create embeddings derived from an in-house podcast KG and apply these semantic representations to proactively recommend underserved podcasts. Our paper makes the following contributions:

(1) We identify underserved podcasts through a large-scale online randomized study on 5M recommendations. We study characteristics of less-streamed podcasts to identify those that would benefit most from strategic recommendation.
(2) We carried out observational studies on the listening behaviour of users on a podcast platform, and show the potential for increasing the audiences of underserved podcasts.
(3) We propose a knowledge based approach that recommends underserved podcasts to mitigate the "rich-get-richer" phenomenon. The approach leverages semantic information about podcasts, via means of knowledge graphs, and improves average precision by up to 19% for underserved podcasts, and 17% overall.
(4) We further show that semantic features (via KGs) outperform collaborative filtering (CF) based features for recommendation of podcasts. Even for podcasts with many user interaction signals, semantic features significantly improved upon CF based features.

2 RELATED WORK

This work is related to social influence studies, reciprocal recommendation systems and knowledge-aware recommendations.

The Influence of Social Messaging on Decision Making. Our randomized study presented in the next section indicates the influence of various variables on podcast listening behaviour. This relates to the literature on the significance of social factors on behaviour and decision making. Chevalier and Mayzlin [10] report that an improvement in online book reviews, even written by strangers, causally leads to an increase in relative sales. Bond et al. [5] report the results of a randomized controlled trial of political ads showing that messages not only influenced political behaviour of millions of people but also users' friends, and friends of friends. Finally, Salganik et al. [25] show how artificially seeding songs with different download counts influences a user's decision whether to listen to the song. Our work studies factors influencing a user's decision on whether to stream a podcast.

Reciprocal Recommendation Systems. Jannach and Adomavicius [14] observe that in practice, recommender systems are designed to satisfy goals for both users and other stakeholders, such as content creators. They discuss the differences between the perspectives of consumers and providers, then outline objectives and metrics for recommendations that satisfy both. Our work aims to balance users' podcast listening preferences and the streaming of underserved podcasts, easing the discovery of upcoming creators.

Podcast creators are outnumbered by podcast listeners, but each listener has only so much time to consider new podcasts. This creates a matching problem, which many have studied. In some recommender systems, e.g., *job search* [20] and *carpooling* [35], content providers and their inventories of items are often outnumbered by consumers but have limited consumer attention to allocate via recommendations. Both Bateni et al. [3] and Chen et al. [9] proposed resource allocation frameworks under an e-commerce setting where sellers optimize for revenue while ensuring buyers' satisfaction. In our work, we aim to surface podcasts with fewer streams while satisfying users' preferences.

Finally, Celma and Cano [7] proposed the *Long Tail* model to help discover niche artists relevant to users' tastes. The model is based on a network of artists (as nodes), edges connecting the co-occurring artists, and artists' popularity measured in terms of their purchases and downloads. This model surfaces the artists from the "torso" of the popularity to help users navigate to the artists in the tail of popularity and put a check on the over-recommendation of homogeneous artists that are either entirely niche or mainstream. In our work, we do not focus on the problem of balancing recommendations based on popularity or any other feature. Rather, we first seek to characterize underserved podcasts, establish that their audiences could potentially grow, and finally develop a model to match podcasts with potential listeners to increase the listener count for such podcasts. Our work thus complements an approach like the Long Tail model, which seeks to balance as well as match.

Knowledge-aware Recommendations. We investigate whether adding semantic information helps improve recommendations for underserved content. Here, we leverage knowledge graphs as a way to add in semantic information. A knowledge graph (KG) is a multi-relational, directed heterogeneous graph, composed of entities (nodes) and relations of different types (edges) [15]. KGs have been applied to question answering [11], semantic search [26], and more recently, to recommender systems [23, 28, 31] and user intent modeling [32]. KG-aware recommender systems often apply KG embedding algorithms on a KG to learn latent vectors of entities and relations. Such embeddings can be incorporated into existing recommender models. Past work has used various knowledge-aware

Table 1: Randomized trial results. We report how often listeners streamed in each segment of our dataset, relative to the mean streaming rate. Differences marked in bold are significant according to a two-sided test with $p < 0.05$.

Partition	Data Fraction	Rel. Streaming
All Rows	100.0%	+0.0%
Podcast Age in Days		
0 - 14	6.7%	**+49.8%**
15 - 28	10.9%	**+13.9%**
29 - inf	82.3%	**-5.9%**
Listener Count		
LQ1	20.9%	**-16.3%**
LQ2	20.0%	**-29.6%**
LQ3	26.0%	-2.5%
LQ4	33.1%	**+30.2%**
Episode Count		
EQ1	22.8%	+1.2%
EQ2	24.8%	**-24.3%**
EQ3	24.2%	**-15.6%**
EQ4	28.1%	**+34.0%**
Podcast Listener		
No	44.5%	**-47.0%**
Yes	55.5%	**+37.7%**

embeddings for recommending e.g. news, e-commerce, movies, and books [23, 29, 30]. Similar to previous work, we train and infer KG embeddings for recommending underserved podcasts.

There are many commonly used embedding algorithms, for example TransE [6], TransR [17], RESCAL [21], DistMult [33], ComplEx [27] and ANALOGY [18]. Here we embed our knowledge graph using DistMult [33]. DistMult is more expressive than the popular TransE algorithm, but still has linear time complexity. While other semantic signals or KG embedding methods exist that could be potentially tested, our goal here is not to specifically benchmark the utility of KGs or different KG embedding methods. Rather, our goal is to examine the utility of adding semantic information to the problem of recommending underserved podcasts.

3 UNDERSERVED PODCASTS

We consider a podcast to be "underserved" if (1) it has relatively few listeners, (2) randomly-selected listeners have streamed it less than average, and (3) it is of sufficient quality to gain more listeners. We first, in Section 3.1, present the results of a randomized study, which was part of a larger unbiased data collection effort to train recommendation models for production use, allowing us to flesh out part (2) of this definition. We do not attempt to define podcast quality for part (3), instead opting to filter out obviously low-quality podcasts from our study. We then, in Section 3.2, report the results of two observational studies of users' listening behaviors on a podcast consumption platform to investigate whether underserved podcasts have the potential to grow their audiences.

3.1 Defining Underserved Podcasts

Our study includes 5.2M recommendations of 90 podcasts to a sample of 3.8M live listeners in an online production setting at Spotify, a global audio streaming platform. In this work, we want to understand how to best recommend underserved podcasts independently of quality issues, which are likely to have some effect. Therefore we only included podcasts professionally produced by established publishers,[2] which we use as our proxy for good quality podcasts.

The online data collection for this study ran over 10 weeks from December 2019 to February 2020. For each podcast, we uniformly sampled listeners ahead of time to receive recommendations. If a targeted listener engaged with the system within a few days of an episode launch, the corresponding podcast was recommended and the interaction (if any) was recorded. When a podcast recommendation was made, a "card" containing the podcast artwork, name, description, and number of episodes was shown to the user. If the user clicked on the card, they could further see the episode list with episode titles, descriptions, release dates and duration. The podcast listener count was not available to the user. If they listened to at least one episode, the interaction was recorded as positive.

Table 1 shows the results of this study with the recommendations further broken down by podcast age, listener count, episode count, and whether the user was an existing podcast listener. Age was defined as the number of days passed between the first podcast episode release and the time of recommendation. We report on listener count and episode count, respectively, in terms of four quantiles (LQ1 – LQ4 and EQ1 – EQ4, respectively) containing equal numbers of podcasts whose boundaries were chosen based on the maximum numbers of listeners/episodes taken by each of the 90 podcasts. Recommendations were then assigned to these quantiles based on the listener/episode count at the time of the recommendation. For instance, it is possible for a podcast to gain (or lose) listeners over the test period so that its recommendations may lie in different quantiles. Recommendations were triggered by episode releases, so we see more data within higher quantiles. Due to confidentiality, we only report relative statistics.

We study the variables that make a podcast less likely to be streamed by a random listener to identify underserved podcasts.

3.1.1 Podcast age. Users respond more positively to recommendations of podcasts that are at most 14 days old, and increases in age lead to less streaming. We refer to this as a "recency factor." It is surprising that this effect fades so quickly, given that many podcasts are continually updated over time. Users could not observe direct indicators of podcast age, but there are often indirect indicators such as episode count or topic that may explain this.

3.1.2 Listener count. Users stream more than average only for recommendations of podcasts from LQ4, which have more listeners than 75% of the podcasts in the study. The listener count was not directly observable to users during the experiment. Its effect serves as a proxy for the podcast's general appeal: a podcast that is more appealing on average will accumulate more listeners and also presumably be streamed more during our experiment. Finding that listener count quantile can be estimated based on whether podcasts are streamed more than average or not indicates that the study is effective to detect underserved podcasts.

3.1.3 Episode count. In general, recommendations are streamed more in each successive quantile. The exception, EQ1, is related to

[2]For our purpose, a (non-exhaustive) list of established publishers was compiled by in-house editors.

Table 2: Relative streaming rate for podcast age versus listener count. Differences marked in bold are significant according to a two-sided test with $p < 0.05$.

		Listener Count			
		LQ1	LQ2	LQ3	LQ4
Podcast Age	0 - 14	**+40.1%**	**+211.1%**	**+55.5%**	**+19.2%**
	15 - 28	**-63.2%**	-7.2%	**+30.1%**	+12.0%
	29 - inf	**-24.0%**	**-38.6%**	**-19.1%**	**+31.3%**

Table 3: Relative streaming rate for podcast age versus episode count. Differences marked in bold are significant according to a two-sided test with $p < 0.05$.

		Episode Count			
		EQ1	EQ2	EQ3	EQ4
Podcast Age	0 - 14	**+65.9%**	**+20.5%**	+38.1%	—
	15 - 28	**+13.9%**	+14.4%	—	—
	29 - inf	**-44.3%**	**-31.5%**	**-15.8%**	**+34.0%**

podcast age and is explored further below. Episode count is correlated with both age and number of listeners, but episode release schedules and consistency of releases vary widely across podcasts. The episode list was available to listeners at recommendation time, so the episode count may have directly impacted listening decisions.

3.1.4 Listener types. Unsurprisingly, users who have not recently streamed any podcasts are less likely than average to stream a recommendation. However, these non-listeners are still somewhat receptive: the ratio of their (absolute) streaming rates is roughly 0.39, so they streamed about 39% as often as podcast listeners.

3.1.5 Podcast age versus listener count. Table 2 shows the relative streaming rate for each combination of age and listener count. In the first two weeks (first row) the "recency factor" applies, and each quantile performs better than average. Already after 2 weeks (second row), listener count seems strongly associated with whether random users would try the podcast. After 4 weeks (third row), only the most popular podcasts are popular among random users. This result helps to motivate the current work: podcasts with smaller audiences are less popular with random users, so we have to make well-targeted recommendations to help them build audiences.

3.1.6 Podcast age versus episode count. Table 3 presents the relative streaming rate for each combination of podcast age and episode count. Listeners are most likely to stream a podcast if it is 0-14 days old and does not have many episodes (EQ1). After that, the largest (significant) increase in streaming comes from the oldest podcasts with the most episodes. Newer podcasts with more episodes tend to have fewer streams. Among podcasts that are at least 29 days old, the more episodes a podcast has, the higher its stream rate, i.e., $EQ4 > EQ3 > EQ2 > EQ1$ in terms of stream rate. Podcasts with longer cadences of new episodes will move upward in episode count more slowly and be in underserved quantiles for longer. Such podcasts need good recommendation targeting to build audiences.

3.1.7 Podcast episode count versus listener count. Table 4 reports the relative streaming rates for listener count versus episode count. Listeners are more likely to stream podcasts in the intersection of LQ4 and EQ4: the most popular podcasts with the highest numbers of episodes. Podcasts with listeners in LQ1 and LQ2 are streamed less than average regardless of episode count except for episode counts in EQ3, which is not significantly different from average.

Table 4: Relative streaming rate for podcast episode versus listener count. Differences marked in bold are significant according to a two-sided test with $p < 0.05$.

		Listener Count			
		LQ1	LQ2	LQ3	LQ4
Episode Count	EQ1	**-12.3%**	**-22.1%**	**+32.7%**	-34.9%
	EQ2	**-51.3%**	**-36.5%**	**-37.8%**	**+24.6%**
	EQ3	+6.4%	**-24.8%**	**+13.0%**	**-40.8%**
	EQ4	**-27.1%**	**-46.7%**	**-22.0%**	**+68.2%**

We identified several characteristics of podcasts that indicate whether random users are more or less likely to stream those podcasts than average. Young podcasts, less than 28 days old and especially less than 14 days old, are much more likely to be streamed. Among podcasts more than 28 days old, the larger the number of listeners or episodes the more likely a user was to stream the podcast. We can now formally identify underserved podcasts as

> *"being more than 28 days old and having either listener counts below LQ4 or episode counts below EQ4."*

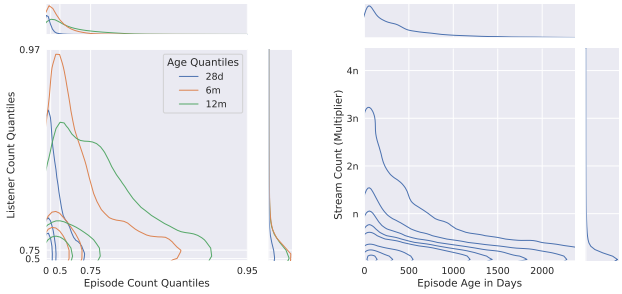
3.2 Underserved Podcasts Growth Potential

Now that we have a definition of underserved podcasts, the next step is to study if such podcasts can expand their audiences. To this end, we carry out two observational studies, one at the podcast level and one at the episode level, conducted on two large datasets, different from the one used in the previous section.

3.2.1 Podcast age and episode count versus audience growth. We explore how podcast age and episode count affect audience growth. For this observational study, we sampled uniformly at random $40K$ podcasts consumed during April 2021, excluding podcasts that were either very popular or older than one year. The sampled dataset had $2M$ listeners, $2.2M$ episodes and $9.5M$ streams. We divided the podcasts into three age groups: up to 28 days old, 29 to 180 days old ("6m"), and 181 to 365 days old ("12m"). For each age group, we report contour lines of the podcast count density estimate over episode count and listener count in Figure 1a; the lines are plotted at 50%, 75% and 95% of the estimated podcast count density.

We see two trends. First, there appears to be some optimal number of episodes per podcast for each podcast age group that drives consumption, except that some of the most popular among the youngest podcasts have fewer episodes. This suggests that podcasts need to produce enough episodes to keep their audience engaged, but that releasing too many episodes too quickly may deter potential listeners, a phenomenon also reported in Section 3.1.6.

The second trend, complicating the first, is that among older podcasts with many episodes at a given episode count those with the most listeners tend to be older. An example of this is podcasts more than 28 days with episode counts above 75% quantile at 95% podcast count density. This trend runs somewhat counter to the previous trend, suggesting that over time listeners who are not deterred by the relatively high release rate may discover the podcast.



(a) For each age group, contour lines of podcast count density estimate, over episode count versus listener count at 50, 75, 95%.

(b) Contour lines of episode count density estimate over episode age versus stream count.

Figure 1: Contour (3 dimensions) plots reporting the results of the two observational studies.

To summarize, podcasts with the most listeners tend to be relatively new, with neither too many nor too few episodes. Podcasts' audiences seem to grow over time, suggesting that well-targeted recommendations may help these podcasts find more listeners faster.

3.2.2 Episode age versus stream count. We next look at how episode age affects listening interest. We sampled uniformly at random $98K$ episodes consumed in April 2021, from $74K$ podcasts and with a total of approximately $3M$ streams. We plot in Figure 1b the contours of episode count density estimate over the episodes' age and the number of times the episodes were streamed in April 2021.

The main takeaways are that (1) the episodes that get streamed the most are relatively new, but (2) even episodes that are several years old still get streamed. This shows that older podcasts are still of interest to some listeners, and that these podcasts may benefit from more strategic recommendations.

These two studies suggest that pairing audiences and podcasts appropriately is key for driving underserved podcast consumption.

4 PODCAST KNOWLEDGE GRAPHS

To facilitate finding audiences for underserved podcasts, we propose to leverage semantic information. To do this, we extract semantic information about the content, licensor, publisher, hosts and guests. We combine this data by means of a knowledge graph. While there may be many ways to inject knowledge signals into recommender systems, our focus here is not to test different KGs or different KG embedding methods, but rather to examine the effects of adding such semantic signals for recommendation.

We build a knowledge graph containing entity types PODCAST, TOPIC, LICENSOR, PUBLISHER, and PERSON. Relationships between entities are typed (e.g. HasTopic, HasLicensor). TOPIC entities are extracted from the episode metadata text field using Google Cloud's Natural Language service.[3] We also include relations between topics (e.g. RelatedTo). Relations between TOPIC entities are extracted from topic co-occurrences from an external corpus like Wikipedia. We extracted PERSON entities using a bi-directional LSTM-CRF model.

[3]https://cloud.google.com/natural-language

Table 5: Knowledge graph statistics.

Entity Type	Count
Person	1.74M
Topic	1.11M
Podcast	706K
Publisher	573K
Licensor	344

Person entities include hosts and guests, for example, for interview style podcasts.

The knowledge graph includes underserved podcasts, as well as a larger collection of other podcasts that are relatively well-consumed. We train the embeddings using all the KG entities and their relations. Statistics for the KG entities are listed in Table 5. As explained in Section 2, we employ the *DistMult* algorithm to embed entities in a 25-dimensional vector space. In particular, we experimented with two knowledge graphs to study the utility of certain entities on the downstream recommendation task and contrast the computational overhead due to a larger set of entities in the graph.

The two knowledge graphs are:

(1) $\texttt{KG}_{\texttt{sm}}$: PODCAST, TOPIC, and LICENSOR.
(2) $\texttt{KG}_{\texttt{lg}}$: PODCAST, TOPIC, LICENSOR, PUBLISHER, and PERSON.

KG embeddings trained on the above two variants are employed to compute cosine similarity between embedding vectors of a podcast we are considering for recommendation and a user's top-k podcasts. Thus, the k cosine similarity scores are treated as features to train the recommendation model. The intention is to capture the semantic relatedness between underserved podcasts and a user's favorite podcasts. In Section 6, we empirically show that a strong signal of user preference such as this matters more when a podcast is less likely to be streamed than an average podcast.

5 EXPERIMENTAL SETUP

In Section 6, we perform an offline evaluation of our knowledge based approach to mitigate the underserved podcast recommendation problem. Here we discuss the datasets, features, models and metrics used in the offline evaluation.

5.1 Datasets

We used the dataset described in Section 3.1, which consists of over 5M recommendations of 90 unique podcasts. The 90 podcasts varied by topic, format (e.g. interview or not), and length. Recommendations are considered as positive examples if they were streamed by the user and negative otherwise. The data collected from December 2019 to January 2020 was used for training, and consists of 3.5M examples. For computational efficiency and to address label imbalance [8], we subsample the negative examples for training. For each training repetition, all the positives were included and for each date, 32%[4] of the negatives were sampled from the data. Finally, the data collected in February 2020 consisting of 1.7M recommendations was used for evaluation.

[4]We doubled the sampling rate until we reached computational capacity.

5.2 Features

We describe the features used as input to the recommendation models. Each recommendation example corresponds to a user-podcast pair and is represented using three types of features.

5.2.1 Basic features. These features are mostly related to the user's podcast usage history (e.g., average time spent listening to podcasts in the past month) and demographics (e.g., age, gender and country). This set is called "Basic" in our experiments. These features indicate the user's likelihood to listen to podcasts in general, but contain little information about which podcasts they might enjoy.

5.2.2 KG features. We added the KG based features $\texttt{KG}_{\texttt{sm}}$ and $\texttt{KG}_{\texttt{lg}}$ (see Section 4). We train `DistMult` based embeddings for a large collection of podcasts including the recommended podcast and each user's top-k podcasts in the KG space, as described in Section 4. We used the *OpenKE* framework [12] out-of-the-box to train the embeddings with default hyperparameters except that we set $epochs = 200$. We then calculated the cosine similarity between the target podcast's vector and each of the user's top k podcast vectors. We choose $k = 5$ as a good trade-off between capturing users' preferences and computational efficiency. This resulted in five KG features, sorted from most similar to least similar. The first feature contains the highest similarity score from the target podcast to any of the user's top five podcasts, the second feature contains the second-highest score, and so on. For missing values, i.e., if the user did not listen to five podcasts, we used -1 (least similar).

5.2.3 CF features. We also experimented with two sets of features based on collaborative filtering (CF). We learned CF vectors for each podcast and user through matrix factorization [16] from user-podcast pairs, setting the item weights for each user according to the cumulative streaming time. The $\texttt{CF}_{\texttt{i}}$ ("item-to-item") features take the cosine similarity between the recommended podcast and each of the user's top five podcast vectors, exactly as for the $\texttt{KG}_{\texttt{sm}}$ and $\texttt{KG}_{\texttt{lg}}$ features. The $\texttt{CF}_{\texttt{u}}$ ("user-to-item") features are the cosine similarity from the recommended podcast to the user's vector.

Due to the cold start problem for collaborative filtering, we do not have CF vectors for new podcasts. This results in more sparsity than for the KG features. We discuss this further in Section 6.3.

5.3 Models

We present results using two statistical methods — Logistic Regression and Generalized Additive Models (GAM) [13] — to model the likelihood of a listener streaming a podcast. Under Logistic Regression each feature makes a linear contribution to the log odds of the dependent variable. Under GAM, the contribution of each feature to the log odds is modeled by a gradient boosted decision tree. Consequently, GAM can account for more complex features and target interactions than Logistic Regression can. We include both to compare results under a simple and a more complex models. For Logistic Regression we used `Scikit-Learn` [24]. For GAM we used the `InterpretML` library [22]. We used the default library parameters. We contrast the performance of these models in Section 6.

5.4 Metrics

We report average precision (AP) and the area under the ROC curve (AUROC) to evaluate the models on various partitions of the test data. AUROC gives a sense of the classifiers' overall performance, and AP, which approximates the area under the precision-recall curve [2], indicates how well sorting by model prediction scores can surface recommendations that were successfully streamed i.e., positive examples. We compare and contrast the effects of the two variants of KG features presented in Section 4. Last but not least, we further use GAM, an interpretable model, to analyze and discuss KG features versus CF features in Section 6.3.

6 RESULTS

We evaluate the performance of the statistical methods using AP and AUROC, and report results for Logistic Regression in Table 6 and for GAM in Table 11. As a reminder and as stated in Section 3.1.7, we are interested in podcasts that are more than one month old and so far have had fewer listeners and/or fewer episodes.

In summary, both KG feature types provide statistically significant improvements in AUROC both overall and for underserved podcasts for both models. For Logistic Regression, both feature types also provide such improvements for AP. However, for GAM, only $\texttt{KG}_{\texttt{sm}}$ improves AP and while the overall improvement is better than for CF features with stastical significance, underserved podcasts are improved more by $\texttt{CF}_{\texttt{u}}$ than by $\texttt{KG}_{\texttt{sm}}$.

As we will see in Section 6.1, although $\texttt{KG}_{\texttt{sm}}$ is better overall, $\texttt{KG}_{\texttt{lg}}$ gave promising gains for certain subsets. Neither KG type dominates the other for all subsets, so it is not clear that the additional entities added for $\texttt{KG}_{\texttt{lg}}$ provided a more useful embedding for this task. The training time for $\texttt{KG}_{\texttt{lg}}$ and $\texttt{KG}_{\texttt{sm}}$ embeddings are 4.8 days (200 `epochs` x 2100 `secs/epoch`) and 3.7 days (200 `epochs` x 1600 `secs/epoch`). The smaller $\texttt{KG}_{\texttt{sm}}$ is efficient and effective for the problem at hand.

Note that in all cases, the metric scores for underserved podcasts are much smaller than those for the full population. This underscores the difficulty of recommending these podcasts, and the need for work such as this to focus on better matching between underserved podcasts and potential listeners.

We discuss results for the Logistic model next and then discuss the GAM and compare model performance in Section 6.2.

6.1 Logistic Regression Results

We explore the effect of KG features on various partitions of the test set with a particular focus on those we identified in Section 3.1.7 as the underserved podcast groups. We limit our discussion to the Logistic model (Table 6), and discuss the impact of choosing a more complex model in Section 6.2. Both types of KG features improved AP and AUROC of the Logistic model for each of the considered partitions, so our discussion will focus on how the features perform relative to each other.

6.1.1 Effects by podcast age group. The CF features have a cold-start problem in general, so it is not surprising that they do not provide much performance improvement for the youngest podcasts. More surprising is the fact that this is one of just a few groups (along with the smallest listener quantile, LQ1, which is correlated with this group) where $\texttt{KG}_{\texttt{lg}}$ displays better performance than $\texttt{KG}_{\texttt{sm}}$. Recall from Section 4 that $\texttt{KG}_{\texttt{lg}}$ incorporates two additional entities, PUBLISHER and PERSON. Perhaps podcast persons (e.g., host or guest) a user is interested in help amplify the "recency factor" enthusiasm

Table 6: Logistic Regression. Improvements over Basic are denoted by + and deteriorations by −. KG results in bold are significantly greater than Basic, $\texttt{CF}_\texttt{i}$, and $\texttt{CF}_\texttt{u}$ according to a one-sided test with combined p-value< 0.05.

		Average Precision					AUROC				
		Basic	+$\texttt{KG}_\texttt{sm}$	+$\texttt{KG}_\texttt{lg}$	+$\texttt{CF}_\texttt{i}$	+$\texttt{CF}_\texttt{u}$	Basic	+$\texttt{KG}_\texttt{sm}$	+$\texttt{KG}_\texttt{lg}$	+$\texttt{CF}_\texttt{i}$	+$\texttt{CF}_\texttt{u}$
All Rows		0.0073	**+17.7%**	**+16.9%**	+1.7%	+2.2%	0.7332	**+2.5%**	**+2.4%**	-0.1%	+0.3%
Underserved		0.0031	**+19.4%**	**+14.9%**	+9.4%	+11.0%	0.6818	**+3.6%**	**+3.2%**	+0.3%	+0.5%
Podcast Age	0 – 14	0.0086	**+11.8%**	**+26.7%**	-39.5%	-18.2%	0.7402	+0.1%	**+3.7%**	-2.2%	-1.0%
	15 – 28	0.0044	**+19.7%**	+7.4%	+9.6%	+14.2%	0.6775	**+3.5%**	**+2.3%**	+0.4%	+1.0%
	29 – inf	0.0082	**+18.9%**	**+18.8%**	+3.4%	+2.3%	0.7501	**+2.6%**	**+2.5%**	+0.0%	+0.3%
Listener Count	LQ1	0.0056	**+9.5%**	**+20.8%**	-13.8%	-7.8%	0.7111	**+3.2%**	**+5.1%**	-1.0%	-0.1%
	LQ2	0.0026	**+13.2%**	**+13.9%**	-1.5%	+5.8%	0.6505	**+2.5%**	**+2.6%**	+0.2%	+0.6%
	LQ3	0.0039	**+13.9%**	+3.6%	+11.8%	+11.5%	0.6713	**+3.4%**	**+2.0%**	+0.6%	+1.0%
	LQ4	0.0103	**+19.6%**	**+19.6%**	+3.1%	+2.8%	0.7733	**+2.2%**	**+2.2%**	+0.0%	+0.2%
Episode Count	EQ1	0.0051	**+18.5%**	**+11.7%**	-1.9%	+7.0%	0.6928	**+2.6%**	**+2.8%**	-0.1%	+0.6%
	EQ2	0.0029	+24.6%	+19.3%	+10.4%	+24.8%	0.6736	**+3.9%**	**+3.3%**	+0.8%	+0.7%
	EQ3	0.0036	**+12.1%**	**+12.7%**	-0.6%	-3.6%	0.6907	**+2.7%**	**+2.6%**	-0.4%	+0.1%
	EQ4	0.0104	**+17.6%**	**+18.4%**	+2.5%	+1.1%	0.7633	**+2.6%**	**+2.6%**	+0.1%	+0.3%

Table 7: Logistic AP for podcast age (in days) versus listener count.

	Listener Count			
Podcast Age	LQ1	LQ2	LQ3	LQ4
0 – 14	**+10.9%** **+28.2%** -48.9% -26.7%	+1.6% **+6.4%** -16.2% +3.8%	+11.3% +8.5% -3.3% +12.3%	—
15 – 28	-8.5% **+15.1%** -17.5% -6.8%	—	**+20.4%** +6.4% +9.1% +14.0%	—
29 – inf	**+9.2%** **+14.5%** +1.4% +1.0%	**+15.4%** **+16.4%** +8.6% +12.0%	+27.7% -1.0% +48.9% +50.8%	**+19.6%** **+19.6%** +3.1% +2.8%

Table 8: Logistic AUROC for podcast age (in days) versus listener count.

	Listener Count			
Podcast Age	LQ1	LQ2	LQ3	LQ4
0 – 14	**+1.0%** **+4.5%** -4.2% -3.3%	-1.8% -1.4% -0.9% +1.4%	-1.7% -1.0% -0.4% +1.5%	—
15 – 28	-0.3% **+10.4%** -4.7% -2.1%	—	**+3.7%** **+2.0%** +0.5% +1.0%	—
29 – inf	**+3.8%** **+3.9%** -0.3% +0.5%	**+4.8%** **+4.9%** +0.6% +0.4%	**+1.0%** -2.4% +0.6% +0.2%	**+2.2%** **+2.2%** +0.0% +0.2%

that trial participants had for the youngest podcasts, favoring a KG with PERSON entities.

The 15-28 day podcasts reverse this trend, with $\texttt{KG}_\texttt{lg}$ not providing better performance than the CF features. In fact, the CF features perform the best with this age range, and we will later see similar performance for LQ3 and EQ2. This suggests that the CF features are strongest for podcasts that are neither too young nor too old.

For the oldest group of podcasts, some of which are underserved, both types of KG feature perform similarly, while the CF features provide only a modest improvement. We explore below the subset of older podcasts that have low listener and/or episode counts, i.e., the underserved podcasts.

6.1.2 Are we helping podcasts with low listener counts? When we partition by listener count quantiles, $\texttt{KG}_\texttt{lg}$ outperforms $\texttt{KG}_\texttt{sm}$ only for podcasts in LQ1, with the fewest listeners. A podcast can have a low listener count simply due to its age, and not due to being underserved. It is important to study the relationship between age and listener count to identify underserved podcasts. Tables 7 and 8 show AP and AUROC results for the Logistic model for the joint distribution of podcast age and the number of listeners. In each cell of these tables, the order of the values from top to bottom is $\texttt{KG}_\texttt{sm}$, $\texttt{KG}_\texttt{lg}$, $\texttt{CF}_\texttt{i}$, $\texttt{CF}_\texttt{u}$. KG entries marked in bold improve on Basic, $\texttt{CF}_\texttt{i}$, and $\texttt{CF}_\texttt{u}$ features according to a one-sided test with combined $p < 0.05$. Empty cells have no test data.

Focusing on the oldest podcasts due to their importance for underserved podcasts, both $\texttt{KG}_\texttt{sm}$ and $\texttt{KG}_\texttt{lg}$ improve over Basic, $\texttt{CF}_\texttt{i}$, and $\texttt{CF}_\texttt{u}$ in both AP and AUROC with statistical significance for listener counts in LQ1, LQ2, and LQ4. $\texttt{KG}_\texttt{sm}$ also performed well in LQ3 but CF did better in this quantile.

For podcasts of any age in LQ1, $\texttt{KG}_\texttt{lg}$ substantially outperformed $\texttt{KG}_\texttt{sm}$ in terms of both AP and AUROC. The improvement for LQ1 is therefore not merely due to the correlation between age and low listener counts. The publisher and person information seems important for recommending podcasts of any age with few listeners.

Recommendation performance for young podcasts with a small number of listeners is decreased by CF based features. This is likely due to a lack of podcast consumption data when podcasts are young, leading to missing or poor quality CF features. This is precisely where CF strategies underperform; this showcases the strength of KG approaches for this group of podcasts.

6.1.3 Are we helping podcasts with low episode counts? In contrast to listener count quantiles, $\texttt{KG}_\texttt{lg}$ does not greatly outperform

Table 9: Logistic AP for podcast age versus episode count.

Podcast Age	Episode Count EQ1	EQ2	EQ3	EQ4
0 – 14	**+11.8%** **+26.7%** -39.5% -18.2%	—	—	—
15 – 28	**+19.7%** +7.4% +9.6% +14.2%	—	—	—
29 – inf	+1.6% -2.1% +98.4% +95.0%	+24.6% +19.3% +10.4% +24.8%	**+12.1%** **+12.7%** -0.6% -3.6%	**+17.6%** **+18.4%** +2.5% +1.1%

Table 10: Logistic AUROC for podcast age versus episode count.

Podcast Age	Episode Count EQ1	EQ2	EQ3	EQ4
0 – 14	+0.1% **+3.7%** -2.2% -1.0%	—	—	—
15 – 28	**+3.5%** **+2.3%** +0.4% +1.0%	—	—	—
29 – inf	-0.5% -2.3% +1.9% -0.4%	**+3.9%** **+3.3%** +0.8% +0.7%	**+2.7%** **+2.6%** -0.4% +0.1%	**+2.6%** **+2.6%** +0.1% +0.3%

$\texttt{KG}_{\texttt{sm}}$ for any group, and the scale of improvements does not seem correlated with the quantile. This suggests that improvements for podcasts is independent of their episode counts.

Tables 9 and 10 show AP and AUROC results for the Logistic model for the joint distribution of podcast age and the number of episodes. In each cell of these tables, the order of the values from top to bottom is $\texttt{KG}_{\texttt{sm}}$, $\texttt{KG}_{\texttt{lg}}$, $\texttt{CF}_{\texttt{i}}$, $\texttt{CF}_{\texttt{u}}$. KG entries marked in bold improve on Basic, $\texttt{CF}_{\texttt{i}}$, and $\texttt{CF}_{\texttt{u}}$ features according to a one-sided test with combined $p < 0.05$. Empty cells have no test data.

Among the oldest podcasts, CF performs anomalously well in EQ1 episode counts with a boost in AP of almost 100%. The reason that the KG features underperform here may be because much of their information comes from entities such as topics and guests extracted from episode titles and descriptions. Podcasts with fewer episodes simply have less information in the KG features. Despite this, the KG features were able to provide a significant increase in AP for younger shows with episode counts in EQ1, explaining why the KG features perform better than CF overall in EQ1. We leave the challenge of improving KG features for podcasts with very few episodes to future work.

In summary, the results presented in this section showed that KG features are effective at recommending underserved podcasts successfully. We believe this is due to the importance of matching users' interests in terms of topics, guests of podcasts, etc.

6.2 GAM Results

We now discuss our performance for the more complex GAM model. Recall that this model is quite similar to the Logistic model, except that each feature has an independent nonlinear contribution to the log odds. This stronger model achieves higher overall AP and AUROC scores than the Logistic model. We summarize its results, stressing differences compared to the Logistic model, and then discuss the functions learned for the KG and CF features in detail.

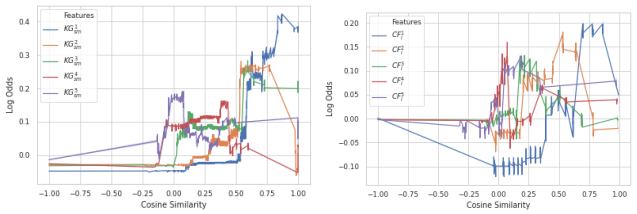


Figure 2: GAM feature scoring functions for $\texttt{KG}_{\texttt{sm}}$ and $\texttt{CF}_{\texttt{i}}$. Given a particular feature value (x axis), this reports the contribution to the predicted log odds of a stream (y axis).

6.2.1 Comparison to Logistic Performance. $\texttt{KG}_{\texttt{sm}}$ improves AUROC and AP both overall and for underserved podcasts, but $\texttt{CF}_{\texttt{u}}$ outperforms $\texttt{KG}_{\texttt{sm}}$ in terms of AP for underserved podcasts. We also note that $\texttt{KG}_{\texttt{lg}}$ does well in the groups where it outperforms $\texttt{KG}_{\texttt{sm}}$ in the Logistic model, namely podcasts which are 0-14 days old or in LQ1. The gains are more modest than for the Logistic model, which may be related to the much stronger performance for the Basic features (it is harder to further improve a better model).

6.2.2 KG and CF Feature Importance. Figure 2 reports the $\texttt{KG}_{\texttt{sm}}$ and $\texttt{CF}_{\texttt{i}}$ features' contributions to the GAM model. In this model, a feature scoring function, the "shape function," is trained independently for each feature using a gradient-boosted decision tree [19]. The shape functions can be interpreted because they do not contain any interactions between features. For this reason we used the GAM for our feature analysis. The final GAM model predicts the log likelihood of a user streaming a podcast by taking the sum of each feature scoring function applied to a data point. The x-axis is the cosine similarity between the recommendation and one of the user's top 5 podcasts, and the y-axis is the log odds of the user streaming the podcast given that cosine similarity. The $\texttt{KG}_{\texttt{sm}}$ and $\texttt{CF}_{\texttt{i}}$ features are numbered from 1 to 5 ($\texttt{KG}^1_{\texttt{sm}}$, $\texttt{KG}^2_{\texttt{sm}}$, …, $\texttt{KG}^5_{\texttt{sm}}$) in order of decreasing podcast similarity.

The scales of the y axes show that $\texttt{KG}_{\texttt{sm}}$ features contribute more to the model log odds than $\texttt{CF}_{\texttt{i}}$ features. Also, the $\texttt{KG}_{\texttt{sm}}$ features with the two highest cosine similarity values ($\texttt{KG}^1_{\texttt{sm}}$ and $\texttt{KG}^2_{\texttt{sm}}$) contribute more at higher values. Further, $\texttt{KG}^4_{\texttt{sm}}$ and $\texttt{KG}^5_{\texttt{sm}}$ increase the log odds even though the cosine similarity to the recommended podcast is negative. This happens because having any value above -1 indicates that the user has recently listened to at least four or five (respectively) different podcasts, and such users are more likely to stream a recommended podcast (see the Podcast Listeners row at the bottom of Table 1). Likewise, while small positive values of $\texttt{CF}^1_{\texttt{i}}$ and $\texttt{CF}^2_{\texttt{i}}$ are negatively correlated with streaming, even small similarities for $\texttt{CF}^3_{\texttt{i}}$, $\texttt{CF}^4_{\texttt{i}}$, and $\texttt{CF}^5_{\texttt{i}}$ are positively correlated, presumably because of the user's larger interest in podcasts.

Overall, the larger impact of the KG features on the model log odds reveals that the KG features are more important than the CF features for the model's final predictions.

6.3 CF features versus KG features

Across our results, CF-based features underperform on both the metrics, AP and AUROC. One factor contributing to this is that

Table 11: GAM. The improvements over Basic are denoted by + and the deteriorations by −. KG results in bold are significantly greater than Basic, CF_i, and CF_u according to a one-sided test with combined p-value< 0.05.

		Average Precision					AUROC				
		Basic	+KG_{sm}	+KG_{lg}	+CF_i	+CF_u	Basic	+KG_{sm}	+KG_{lg}	+CF_i	+CF_u
All Rows		0.0092	**+1.5%**	-1.1%	-1.9%	+0.6%	0.7408	**+1.0%**	**+0.7%**	+0.0%	+0.2%
Underserved		0.0043	+0.6%	-0.5%	+0.4%	+1.6%	0.7241	**+0.6%**	**+0.5%**	-0.2%	+0.1%
Podcast Age	0 – 14	0.0089	-0.7%	**+8.1%**	-14.5%	-11.3%	0.7692	+0.0%	**+1.3%**	-0.0%	+0.2%
	15 – 28	0.0057	**+21.2%**	+3.5%	+10.0%	+15.5%	0.7055	**+2.2%**	**+0.9%**	+0.7%	+0.4%
	29 – inf	0.0104	-0.2%	-2.0%	-2.4%	-0.2%	0.7492	**+0.8%**	**+0.6%**	-0.1%	+0.1%
Listener Count	LQ1	0.0065	+0.1%	**+8.6%**	-5.3%	-4.4%	0.7508	+0.0%	**+1.0%**	-0.2%	+0.1%
	LQ2	0.0035	-4.4%	+3.4%	+4.5%	+5.2%	0.6926	**+2.1%**	**+1.5%**	-0.1%	+0.2%
	LQ3	0.0054	+2.3%	-8.7%	+4.5%	+6.6%	0.7039	**+1.7%**	+0.6%	+0.6%	+0.5%
	LQ4	0.0128	**+1.6%**	-0.8%	-2.9%	+0.1%	0.7578	**+0.9%**	**+0.5%**	-0.2%	+0.1%
Episode Count	EQ1	0.0065	**+13.7%**	+4.4%	+4.3%	+8.9%	0.7209	**+1.6%**	**+1.0%**	+0.5%	+0.4%
	EQ2	0.0040	+0.0%	**+10.5%**	+7.1%	+7.7%	0.7132	**+1.4%**	**+1.4%**	-0.1%	+0.1%
	EQ3	0.0048	-2.9%	-4.2%	-4.9%	-2.3%	0.7323	-0.1%	-0.2%	-0.3%	+0.0%
	EQ4	0.0128	-0.5%	-2.4%	-2.5%	-0.4%	0.7500	**+1.0%**	**+0.6%**	-0.1%	+0.1%

Table 12: Sparsity of KG and CF features. We report the number of examples with each count of top-5 CF and KG features having values above -1.

	Training Set					
Num. Feat. > -1	0	1	2	3	4	5
KG_{sm}/KG_{lg}	1.8M	536K	277K	181K	181K	497K
CF_i	3.2M	986	8.5K	14K	37K	193K
	Test Set					
Num. Feat. > -1	0	1	2	3	4	5
KG_{sm}/KG_{lg}	787K	278K	150K	100K	100K	283K
CF_i	1.55M	764	4.5K	7.8K	20K	108K

many new podcasts do not have a long enough history to train CF vectors. This results in most CF-based features having values -1 (least similar), which is not necessarily true and is thus misleading. This explanation is supported by Table 12, which shows the number of recommendation examples by the top five KG and CF features across both the test set and the training set. The number decreases with each additional podcast, except for an increase at five caused by the grouping of users having five or more podcasts. There are far more KG features than there are CF features.

A second contributing factor might be that adding semantic information about the podcast helps matching users to podcasts better than CF does. For instance, CF, by design, depends on popularity signals, which might not be helpful for underserved podcasts in particular because such podcasts by definition are not popular.

6.4 Discussion

The aim of this work was to see how we can address the underserved podcasts problem through using semantic information. We proposed to do this via means of knowledge graphs. Our experiments show that semantic information can help alleviate this problem. The main takeaways are:

(1) Adding semantic information helps bring in different types of knowledge, which can encompass the diverse reasons listeners may enjoy more niche podcasts. Semantic information is particularly important for not only helping in better matching underserved podcasts to potentially interested listeners, but also in striking a balance between recommending popular podcasts versus more niche (and often underserved) podcasts.
(2) In general, a model that is able to provide more nuance into the importance of the features leads to better results, as well as an understanding of how the features help.
(3) Although not the topic of this paper, the incorporation of semantic information via models incorporating KG embeddings are relatively easy to deploy for large-scale usage. Such an approach using KG embeddings has been shown to be successful in the past for trajectory-based podcast recommendations [4].

Our work is currently being further expanded to incorporate an important dimension, assessing podcast quality. Our work focused on quality underserved podcasts, and we need to consider all underserved podcasts, not just those we are sure of the quality by simply looking at the publishers of such podcasts. Quality will be an important factor in deciding which underserved podcasts should a recommender system help in growing an audience.

7 CONCLUSION

This work explored underserved podcasts. We found that users are more likely to stream podcasts that are less than 28 days old or that have many listeners and episodes, which we used to define "underserved podcasts." Two observational studies then showed that underserved podcasts have the potential to grow their audiences. Finally, we empirically showed that a semantic based approach, via KGs, can aid the discovery of underserved podcasts. We found that the smaller KG, omitting guests and publishers, was adequate to achieve the observed performance gains, although we expect future work to find utility in these entities. For podcasts with more niche appeal, recommendations based on a user's interests in podcast topics, guests, etc. are critical to help them build larger audiences.

REFERENCES

[1] Vito Walter Anelli, Tommaso Di Noia, Eugenio Di Sciascio, Azzurra Ragone, and Joseph Trotta. How to make latent factors interpretable by feeding factorization machines with knowledge graphs. In *International Semantic Web Conference*, pages 38–56. Springer, 2019.

[2] Javed A Aslam, Emine Yilmaz, and Virgiliu Pavlu. A geometric interpretation of r-precision and its correlation with average precision. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 573–574, 2005.

[3] Mohammad Hossein Bateni, Yiwei Chen, Dragos Florin Ciocan, and Vahab Mirrokni. Fair resource allocation in a volatile marketplace. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pages 819–819, 2016.

[4] Greg Benton, Ghazal Fazelnia, Alice Wang, and Ben Carterette. Trajectory based podcast recommendation. *arXiv preprint arXiv:2009.03859*, 2020.

[5] Robert M Bond, Christopher J Fariss, Jason J Jones, Adam DI Kramer, Cameron Marlow, Jaime E Settle, and James H Fowler. A 61-million-person experiment in social influence and political mobilization. *Nature*, 489(7415):295–298, 2012.

[6] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795, 2013.

[7] Òscar Celma and Pedro Cano. From hits to niches? or how popular artists can bias music recommendation and discovery. In *Proceedings of the 2nd KDD Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition*, pages 1–8, 2008.

[8] Nitesh V. Chawla, Nathalie Japkowicz, and Aleksander Kotcz. Editorial: Special issue on learning from imbalanced data sets. *SIGKDD Explor. Newsl.*, 6(1):1–6, June 2004. ISSN 1931-0145. doi: 10.1145/1007730.1007733. URL https://doi.org/10.1145/1007730.1007733.

[9] Long-Sheng Chen, Fei-Hao Hsu, Mu-Chen Chen, and Yuan-Chia Hsu. Developing recommender systems with the consideration of product profitability for sellers. *Information Sciences*, 178(4):1032–1048, 2008.

[10] Judith A. Chevalier and Dina Mayzlin. The effect of word of mouth on sales: Online book reviews. *Journal of Marketing Research*, 2006.

[11] Li Dong, Furu Wei, Ming Zhou, and Ke Xu. Question answering over freebase with multi-column convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 260–269, 2015.

[12] Xu Han, Shulin Cao, Lv Xin, Yankai Lin, Zhiyuan Liu, Maosong Sun, and Juanzi Li. Openke: An open toolkit for knowledge embedding. In *Proceedings of EMNLP*, 2018.

[13] Trevor J Hastie and Robert J Tibshirani. *Generalized additive models*, volume 43. CRC press, 1990.

[14] Dietmar Jannach and Gediminas Adomavicius. Recommendations with a purpose. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pages 7–10, 2016.

[15] Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and Philip S Yu. A survey on knowledge graphs: Representation, acquisition and applications. *arXiv preprint arXiv:2002.00388*, 2020.

[16] Maciej Kula. Metadata embeddings for user and item cold-start recommendations. *arXiv preprint arXiv:1507.08439*, 2015.

[17] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[18] Hanxiao Liu, Yuexin Wu, and Yiming Yang. Analogical inference for multi-relational embeddings. *arXiv preprint arXiv:1705.02426*, 2017.

[19] Yin Lou, Rich Caruana, and Johannes Gehrke. Intelligible models for classification and regression. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, page 150–158, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450314626. doi: 10.1145/2339530.2339556. URL https://doi.org/10.1145/2339530.2339556.

[20] Tsunenori Mine, Tomoyuki Kakuta, and Akira Ono. Reciprocal recommendation for job matching with bidirectional feedback. In *2013 Second IIAI International Conference on Advanced Applied Informatics*, pages 39–44. IEEE, 2013.

[21] Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. A three-way model for collective learning on multi-relational data. In *Icml*, volume 11, pages 809–816, 2011.

[22] Harsha Nori, Samuel Jenkins, Paul Koch, and Rich Caruana. Interpretml: A unified framework for machine learning interpretability. *arXiv preprint arXiv:1909.09223*, 2019.

[23] Enrico Palumbo, Giuseppe Rizzo, and Raphaël Troncy. Entity2rec: Learning user-item relatedness from knowledge graphs for top-n item recommendation. In *Proceedings of the eleventh ACM conference on recommender systems*, pages 32–36, 2017.

[24] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[25] Matthew J Salganik, Peter Sheridan Dodds, and Duncan J Watts. Experimental study of inequality and unpredictability in an artificial cultural market. *science*, 311(5762):854–856, 2006.

[26] Sean Szumlanski and Fernando Gomez. Automatically acquiring a semantic network of related concepts. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 19–28, 2010.

[27] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. International Conference on Machine Learning (ICML), 2016.

[28] Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 417–426, 2018.

[29] Jizhe Wang, Pipei Huang, Huan Zhao, Zhibo Zhang, Binqiang Zhao, and Dik Lun Lee. Billion-scale commodity embedding for e-commerce recommendation in alibaba. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 839–848, 2018.

[30] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12):2724–2743, 2017.

[31] Xiang Wang, Dingxian Wang, Canran Xu, Xiangnan He, Yixin Cao, and Tat-Seng Chua. Explainable reasoning over knowledge graphs for recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5329–5336, 2019.

[32] Xiang Wang, Tinglin Huang, Dingxian Wang, Yancheng Yuan, Zhenguang Liu, Xiangnan He, and Tat-Seng Chua. Learning intents behind interactions with knowledge graph for recommendation. *arXiv preprint arXiv:2102.07057*, 2021.

[33] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575*, 2014.

[34] Xiao Yu, Xiang Ren, Yizhou Sun, Quanquan Gu, Bradley Sturt, Urvashi Khandelwal, Brandon Norick, and Jiawei Han. Personalized entity recommendation: A heterogeneous information network approach. In *Proceedings of the 7th ACM international conference on Web search and data mining*, pages 283–292, 2014.

[35] Desheng Zhang, Tian He, Yunhuai Liu, Shan Lin, and John A Stankovic. A carpooling recommendation system for taxicab services. *IEEE Transactions on Emerging Topics in Computing*, 2(3):254–266, 2014.