Localify.org: Locally-focus Music Artist and Event Recommendation

Douglas R. Turnbull
dturnbull@ithaca.edu
Ithaca College
Ithaca, NY, USA

April Trainor
atrainor@ithaca.edu
Ithaca College
Ithaca, NY, USA

Griffin Homan
ghoman@ithaca.edu
Ithaca College
Ithaca, NY, USA

Elizabeth Richards
lrichards2@ithaca.edu
Ithaca College
Ithaca, NY, USA

Kieran Bentley
kbentley@ithaca.edu
Ithaca College
Ithaca, NY, USA

Victoria Conrad
vconrad@ithaca.edu
Ithaca College
Ithaca, NY, USA

Paul Gagliano
pgagliano@ithaca.edu
Ithaca College
Ithaca, NY, USA

Cassandra Raineault
craineault@ithaca.edu
Ithaca College
Ithaca, NY, USA

Thorsten Joachims
tj@cs.cornell.edu
Cornell University
Ithaca, NY, USA

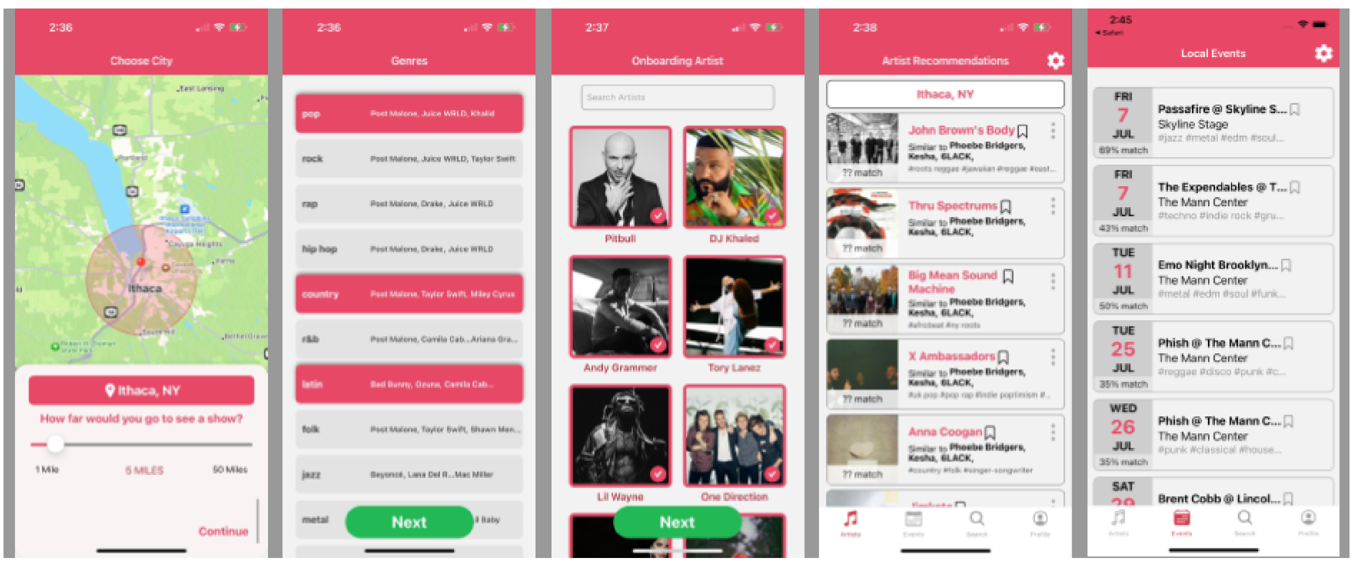

Figure 1: Localify.org - (far left) A user selects a city and how far they might travel to a music event, then (middle left) picks from a list of music genres, and finally (middle) picks preferred artists related to those genres. Based on these selected *seed* artists, Localify recommends both (middle right) *local* artists from that city and (far right) artists who have upcoming events in that city.

ABSTRACT

Cities with strong local music scenes enjoy many social and economic benefits. To this end, we are interested in developing a locally-focused artist and event recommendation system called Localify.org that supports and promotes local music scenes. In this demo paper, we describe both the overall system architecture as well as our core recommendation algorithm. This algorithm uses artist-artist similarity information, as opposed to user-artist preference information, to bootstrap recommendation while we grow the number of users. The overall design of Localify was chosen based on the fact that local artists tend to be relatively obscure and reside in the long tail of the artist popularity distribution. We discuss the role of popularity bias and how we attempt to ameliorate it in the context of local music recommendation.

RecSys '23, September 18–22, 2023, Singapore, Singapore

ACM ISBN 979-8-4007-0241-9/23/09.
https://doi.org/10.1145/3604915.3610645

KEYWORDS

music recommendation, long-tail recommendation, popularity bias

ACM Reference Format:
Douglas R. Turnbull, April Trainor, Griffin Homan, Elizabeth Richards, Kieran Bentley, Victoria Conrad, Paul Gagliano, Cassandra Raineault, and Thorsten Joachims. 2023. Localify.org: Locally-focus Music Artist and Event Recommendation. In *Seventeenth ACM Conference on Recommender Systems (RecSys '23), September 18–22, 2023, Singapore, Singapore.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3604915.3610645

1 INTRODUCTION

Urban planners and musicologists have shown that vibrant local music scenes can benefit communities by enhancing social bonding, improving emotional well-being, and increasing economic activity [3, 4, 11, 16]. In an attempt to help strengthen local music scenes, we are developing a music recommendation system called *Localify.org* that is designed to support local artists and promote upcoming events at nearby venues. The user experience, as shown in Figure 1, starts with a user providing a list of favorite artists (which we refer to as *seed* artists) and a city (e.g., a hometown). The user then manually selects seed artists if they are a guest or creates an account with an email address. Alternatively, if the user logs in with an account from a streaming music platform (e.g., Spotify, Apple Music) their seed artists are automatically imported. Our system then produces a ranking of artists who either originate from that city, have frequently played a large proportion of their live events in the city, or have one or more upcoming events in the city. We generically refer to all of these artists as *local* candidate artists. Put succinctly, our goal is to rank the set of local candidate artists given a user's seed artists.

One common problem is that most local artists can be considered *long-tail* artists since they tend to be relatively less well-known when compared to the mainstream artists who are typically played on the radio or featured in popular playlists of streaming providers. Localify provides a platform that we can use to explore and improve recommendation performance for less popular artists, even if this comes at the expense of more popular artists [13].

Another problem with developing a music recommendation system (in an academic setting) is that, initially, we don't have access to a large database of user-artist preference data. Instead, we make use of artist-artist similarity data obtained using public APIs from Spotify and Last.fm, and describe how we can use this artist-artist matrix to bootstrap an initial local artist recommendation system. Our initial recommender system involves adapting a matrix factorization approach proposed by Hu et al. [6] to work with an artist-artist similarity matrix.

2 THREE FORMS OF POPULARITY BIAS

When exploring the task of long-tail item recommendation, the concept of *popularity bias* naturally arises. However, this term is overloaded in that there are at least three different (but related) versions of popularity bias. The first, which we refer to as *popularity-related degradation*, relates to a decrease in recommender system performance as a function of lower item popularity [13]. The second version is *popular item feedback advantage* in which popular items tend to have more user feedback (ratings, streams, clicks) associated with them. These first two versions of popularity bias are related in that recommender systems tend to be optimized per unit of feedback (e.g., each user-item interaction), and since popular items are associated with more feedback, an algorithm that does better on popular items will perform better overall. Our goal is to use Localify to explore ways in which we can reduce or eliminate the feedback advantage and minimize popularity-related degradation.

The third version of popularity bias, called *popularity lift*, is related to the difference in popularity between items in the user's profile (i.e., the inputs) and items that are recommended to the user (the outputs) [1, 8]. The intuition is that if we think of recommender systems as traversing a graph of (embedded) items and users[7], there are more paths to the popular items. This can be pernicious in that it can contribute to a rich-get-richer cultural marketplace [2, 12] in which popular items crowd out less popular items that might be of more value to the user. Crowding out is especially important in the music domain because of the *mere exposure effect* [5, 15] in which listeners have to first become familiar with the music before they will appreciate it.

Our goal for Localify is to undo this popularity lift bias so that local artists have a fair chance to compete for the user's attention. In addition, users are more likely to prefer an artist after they have seen them live [14]. If we recommend (less popular) local artists who play inexpensive shows at nearby venues, the user may find value in a recommender system that recommends these artists over (mainstream) non-local artists they typically might listen to.

3 BOOTSTRAPPING A LOCAL MUSIC RECOMMENDATION SYSTEM

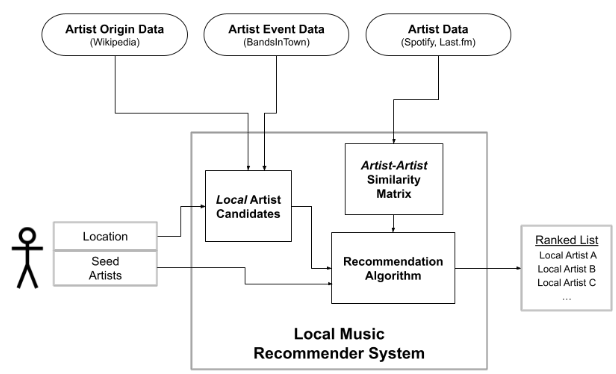

Figure 2: Architecture of Localify.org

A key challenge in developing a novel music recommendation system is that we do not initially have access to a large-scale user-item preference matrix which is the typical input to a recommendation algorithm. However, we do have access to public APIs from music tech companies (Spotify[1], Last.fm[2]) that provide us with both a set of similar artists for a given artist as well as a measure of popularity (score between 0-100 for Spotify, listener count for Last.fm) for each artist. This allows us to create a large artist-artist similarity matrix by repeatedly querying these APIs for an artist, and then that artist's similar artists, and so on (i.e., snowball sampling.) So far, we have collected similarity and popularity information for a set

[1]https://developer.spotify.com/documentation/web-api
[2]https://www.last.fm/api

$\mathcal{A}$ of 600,000 artists. We have also collected music event information for over two million live music events from event aggregation sites (e.g., BandsInTown, Eventbrite) and artist origin information for 43,000 artists from sites like Wikipedia and AllMusic. Finally, our user interface provides users with an opportunity to submit (crowd-sourced) information about artist origins and upcoming events.

A user profile consists of a set of *seed* artists and a location. In general, we expect that a user's seed artists will be popular "landmark" artists (e.g., The Beatles, Drake, Taylor Swift) but they can also be relatively obscure and niche depending on the user's listening preferences. Our recommender system then finds the set of *local* candidate artists which the recommendation algorithm scores and ranks. A local candidate artist can either have a known connection (e.g., birthplace, formed, currently reside) or have an upcoming music event within the user-selected distance from the user-selected location.

Our current recommendation algorithm involves factorizing the artist-artist similarity matrix using Alternating Least Squares (MF-ALS) [6] such that each artist is embedded into a k-dimensional space. First, we represent a user as a sparse $|\mathcal{A}|$-dimensional vector with a value of 1 for each dimension corresponding to each of the user's seed artists. We then embed this sparse user vector into the k-dimensional artist similarity space. Finally, we calculate and sort the distances between this vector and each of the local candidate artist vectors.

For each top-ranked local candidate artist, we also calculate an "explanation score" for each of the seed artists (see Section 5 in [6].) This score represents the extent to which the seed artist influenced the recommendation score for the local candidate artists. We then use the seed artists with the largest explanation scores to explain the recommendation to the user. (I.e., "You should check out local artists A because they are similar to your favorite artists X and Y.") Finally, we calculated a "percent match" for each top-ranked candidate as a percentile score between the candidate and the user vector relative to a large random subset of all artists in our data set. The explanation and the percent match score, along with common music genre tags, are used together to contextualize a novel local artist to the user.

We have found that our modified artist-artist MF-ALS algorithm is able to rank candidate artists such that the genres of the candidate artists that appear at the top of the rankings match the genres of the seed artists (AUC 0.71). We also observe noticeable popularity-based degradation in that artists with relatively high popularity (i.e., Spotify popularity between 75 to 80) have an average AUC of 0.80 while artists with low popularity (20 to 25) have an average AUC of 0.57.

Our future work involves improving recommendation accuracy using MF-ALS, especially for low-popularity artists as opposed to optimizing for all artists in our data set. We also are working on adapting existing recommendation algorithms (e.g., SLIM [10], Mutli-VAE [9]) to work with artist-artist similarity information.

4 DEMO

Localify.org is both a web app and two native mobile apps for Android and iOS. Using Localify, users can interact with local artist and event recommendations, as well as contribute information about artists, venues, and events. In addition, users can automatically generate personalized playlists on Spotify or Apple Music (if the user has linked their account) that alternate songs from a user's seed artists and their recommended local artists. This provides a balance between familiar music from known artists (exploitation) and new music from local artists (exploration). It also allows the user to hear the quality of the local music in the context of often more familiar (mainstream) music. Lastly, users can subscribe to a weekly email with personalized recommendations for newly-listed and upcoming music events.

Our demo will also show our administrative dashboard which we use to monitor user statistics, crowd-sourced user contributions, web scraping statistics, and recommender system performance over time. Our core (delayed) evaluation metric is the number of local artists who have entered a user's heavy listening rotation in the days or weeks after we have recommended them to the user. In addition, we track how often a user explores a recommended local artist (clicks on the detailed artist page, listens to song clip) or recommended upcoming event (clicks on the detailed event page, adds the event to their calendar.) These evaluation metrics are intended to measure the impact of Localify on getting users to increase engagement with their local music scenes.

Localify is an academic project that is free to use, does not show advertisements, does not sell user data, and is not biased by corporate interests [2, 12]. Our core design goal is to use music recommendations to support local artists and small music venues so as to promote a healthy local music scene. A narrated screen capture of our app can be found at Localify.org.

ACKNOWLEDGMENTS

This research was supported in part by NSF Award IIS-1901168/IIS-1901330. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

REFERENCES

[1] Himan Abdollahpouri, Masoud Mansoury, Robin Burke, and Bamshad Mobasher. 2019. The Unfairness of Popularity Bias in Recommendation. *arXiv preprint arXiv:1907.13286* (2019).

[2] Luis Aguiar and Joel Waldfogel. 2021. Platforms, power, and promotion: Evidence from Spotify playlists. *The Journal of Industrial Economics* 69, 3 (2021), 653–691.

[3] Sarah Baker, Raphaël Nowak, Paul Long, Jez Collins, and Zelmarie Cantillon. 2020. Community Well-Being, Post-Industrial Music Cities and the Turn to Popular Music Heritage. In *Music Cities*. Springer, 43–61.

[4] Titan Music Group (Firm). 2015. *The Austin Music Census: A Data-Driven Assessment of Austin's Commercial Music Economy*. Titan Music Group, LLC. https://books.google.com/books?id=fStOjwEACAAJ

[5] Anders C Green, Klaus B Bærentsen, Hans Stødkilde-Jørgensen, Andreas Roepstorff, and Peter Vuust. 2012. Listen, learn, like! Dorsolateral prefrontal cortex involved in the mere exposure effect in music. *Neurology research international* 2012 (2012).

[6] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE International Conference on Data Mining*. Ieee, 263–272.

[7] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.

[8] Dominik Kowald, Markus Schedl, and Elisabeth Lex. 2020. The unfairness of popularity bias in music recommendation: A reproducibility study. In *European Conference on Information Retrieval*. Springer, 35–42.

[9] Dawen Liang, Rahul G Krishnan, Matthew D Hoffman, and Tony Jebara. 2018. Variational Autoencoders for Collaborative Filtering. In *Proceedings of the 2018 world wide web conference*. 689–698.

[10] Xia Ning and George Karypis. 2011. Slim: Sparse Linear Methods for Top-N Recommender Systems. In *2011 IEEE 11th International Conference on Data Mining*. IEEE, 497–506.

[11] Steve Oakes and Gary Warnaby. 2011. Conceptualizing the Management and Consumption of Live Music in Urban Space. *Marketing Theory* 11, 4 (2011), 405–418.

[12] Liz Pelly. June 21, 2017. The Secret Lives of Playlists. *Cash Music* (June 21, 2017). https://watt.cashmusic.org/writing/thesecretlivesofplaylists

[13] Harald Steck. 2011. Item popularity and recommendation accuracy. In *Proceedings of the fifth ACM conference on Recommender systems*. 125–132.

[14] Jan-Christian Tonon, Jörg Claussen, and Christian Peukert. 2012. The Eff ect of Live Performances on Artist Popularity. *Available at SSRN 2142353* (2012).

[15] Robert B. Zajonc. 2001. Mere Exposure: A Gateway to the Subliminal. *Current Directions in Psychological Science* 10 (2001), 224 – 228.

[16] Michael Zhou, Andrew McGraw, and Douglas R Turnbull. 2022. Towards Quantifying the Strength of Music Scenes Using Live Event Data. *Population* 18 (2022), 21.