Hard Negative Sampling for Music Recommendation: Real-World Impacts on Accuracy and Diversity

M. Jeffrey Mei[1,*], Oliver Bembom[1] and Andreas F. Ehmann[1]

[1] *SiriusXM Radio Inc., 1221 Avenue of the Americas, New York, New York 10020, USA*

Abstract

Using real user negative feedback as hard negative samples is known to help improve recommendation accuracy and diversity. We evaluate the benefits of implicit and explicit negative feedback and their impacts on model accuracy and recommendation diversity for Pandora and Spotify datasets, including an online test on Pandora users. We find both implicit and explicit negative feedback can increase both accuracy and diversity, and confirm this in the online test. Moreover, mixing in some randomly-sampled negative feedback as well gives the highest diversity, without hurting the accuracy. We evaluate the prediction accuracy for users with different proportions of positive and negative feedback. Lastly, we explore possible connections between prediction accuracy and song diversity that may occur from introducing hard negative samples.

Keywords

music recommendation, sequential recommendation, negative sampling, diversity

1. Introduction

Modern music streaming platforms have vast catalogs comprising millions (if not tens or hundreds of millions) of songs, of which a user is likely only interested in hearing a small proportion. Recommender systems are crucial to effectively filter and personalize these tracks for each user. These recommender systems may be powered by user feedback. Although positive feedback (e.g. clicks, views, or a 'like' button) is generally easier to collect than negative feedback, where available, negative feedback can be incorporated into recommender systems to improve their accuracy [1, 2, 3, 4]. In the music domain, negative feedback may be explicit 'down-thumb'/'dislike' negative feedback, or implicit feedback (e.g. song skips) which may share traits of both positive and negative feedback [5, 4].

Sequential and temporal information have been found to improve music recommendation accuracy [e.g. 6, 7, 4]. Negative feedback can also be included as additional inputs to improve recommendation accuracy [e.g. 3, 1, 4], as well as recommendation diversity [8]. Poor recommendation diversity may also enhance already-extant popularity biases in music recommender systems [9].

In addition to using negative feedback as inputs, negative feedback can also be used as negative targets for prediction during training. In cases where no negative feedback exists, often random negative samples are used: this can lead to model overconfidence as random negatives are often irrelevant and unrelated to the positive target and are therefore unrealistic [10]. A variety of ways to improve on random negative sampling have been explored, such as picking the highest-scoring random negative out of a batch [11], frameworks to avoid false hard negatives [12], customized loss functions [10] and augmenting negative samples to better contrast against positive samples [13].

In this paper, we explore how the recommendation accuracy and diversity for a transformer model are affected by incorporating hard negative samples during training, culminating in an online A/B test. More specifically, we find that:

- The accuracy and diversity of recommendations generally both increase when using some hard negatives; however, too many hard negatives causes diversity to drop

MuRS 2024: 2nd Music Recommender Systems Workshop, October 14th, 2024, Bari, Italy

[*] Corresponding author.

jeffrey.mei@siriusxm.com (M. J. Mei); oliver.bembom@siriusxm.com (O. Bembom); andreas.ehmann@siriusxm.com (A. F. Ehmann)

0000-0002-1083-598X (M. J. Mei); 0009-0002-2617-5776 (O. Bembom); 0009-0003-9589-0666 (A. F. Ehmann)

- The higher accuracy is generally caused by both positive and negative targets being ranked lower, which also affects the recommendation diversity. Negative targets are lowered by a greater amount, causing a gain in overall accuracy.
- The recommendation accuracy is fairly robust with respect to differing proportions of positive/negative feedback in user data, although there is a slight accuracy increase for users with a higher proportion of positive feedback

2. Data

We use two datasets for our analysis. One is Spotify's open-source Sequential Skip Prediction dataset [14] and the other is a proprietary dataset from Pandora. Both are filtered for 'radio stations' only. Summary statistics are shown in Table 1. The collected feedback types in the two datasets are not necessarily equivalent, which is discussed later in Sect. 6.1. Positive feedback ('$+$') for Pandora is explicit up-thumbs given by the user, whereas for Spotify it is song plays (i.e. implicit positive feedback). The negative ('$-$') feedback for Pandora includes both explicit down-thumb and implicit skip feedback, whereas for Spotify only includes implicit skip feedback.

Table 1
Summary statistics for the processed training sets. Statistics here are not necessarily representative of their user bases.

	Spotify	Pandora
Granularity	Session	User
Feedback types	play, skip	up, skip, down
Ratio of 'positive':'skip':'down'	$1 : 0.9 : 0$	$4 : 13 : 1$
Max. seq. length	20	400
Median seq. length	9	289
Max. lookback	Same day	1 year
No. of sequences	8×10^6	10^7
No. of tracks	3×10^6	10^6

3. Model

The model is based off SASRec [15], with more details given in [16] and [4]. The benefits of using a transformer model vs. matrix factorization have been already quantified in [16] and are not the focus of this work. The benefits in accuracy and coverage of including negative feedback as *inputs* is also covered in [4]. This work aims to quantify the benefits of using negative feedback as *targets* (real hard negative samples) within the same transformer model described above and its impacts on recommendation accuracy and diversity. We define a hyperparameter p_{hard}, which is the proportion of negative samples each epoch that use real negative feedback instead of a randomly-sampled negative. Negative samples are re-sampled each epoch, so for $0 < p_{hard} < 1$, the exact positions for which hard negative samples are used may differ each epoch.

4. Evaluation

The model is evaluated against users who have given both positive and negative feedback in the subsequent month ('test period') after the training time window ($\sim$3 years of sessions for Spotify and $\sim$1 year of user feedback for Pandora). The test periods for the Pandora/Spotify dataset start on 2024-02-01 and 2018-08-15 respectively. For each user who has given both '$+$' and '$-$' feedback, we pick a random ('$+$','$-$') pair (from the same radio station). A successful prediction is when the '$+$' song is correctly scored higher than the '$-$' song. This test accuracy is equivalent to the averaged per-user area under the receiver operating characteristic curve (**AUC**) for all users in the test set [17]. A model that randomly guesses would have an accuracy of 0.5. We use explicit negative feedback for testing because if we test against randomly-sampled negatives, the accuracy $\gg 0.99$, which is not very useful, nor realistic, as our model is used to distinguish and rank similar songs which are on the same user-selected station.

5. Results

We find that using hard negatives generally increases the prediction accuracy, though this lift quickly plateaus after $p_{hard} \approx 0.3$ (Fig. 1a). This is likely because for any $p_{hard} > 0$, all positive targets for prediction will eventually be paired with a hard negative (as each target's negative sample is re-sampled each epoch with probability p_{hard} of being a hard negative). Higher p_{hard} simply means that these hard negatives are used more frequently, which may help the model converge faster (Fig. 1d).

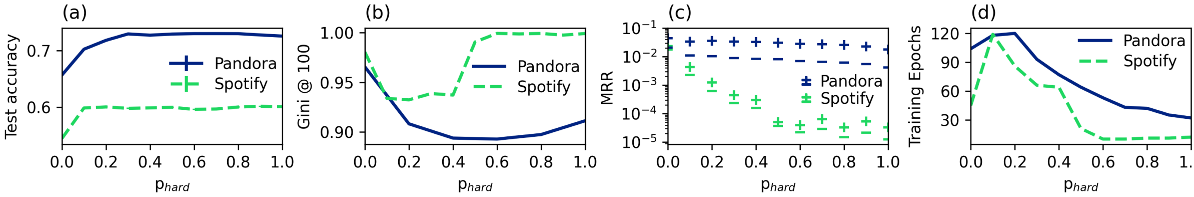

Figure 1: Summarized results showing the impact of differing p_{hard} on (a) model accuracy, (b) recommendation diversity, (c) the ranks of positive and negative targets and (d) training time.

However, the diversity (measured in terms of the Gini coefficient, where a lower Gini means higher diversity) seems to have a local minimum around $0.2 < p_{hard} < 0.6$ (depending on the dataset), and then increases again for $p_{hard} \gg 0$. The optimal p_{hard} for highest diversity for a given dataset may depend on dataset statistics such as average sequence length (Table 1) or others. Given the accuracy is relatively static for $p_{hard} > 0$, we are able to optimize for higher diversity without having to sacrifice accuracy.

Although the accuracy is relatively static for $p_{hard} > 0$ (Fig. 1a), the mean reciprocal ranks (MRR) are not (Fig. 1c). Both the positive and negative targets are ranked lower as p_{hard} increases. The accuracy gain compared to $p_{hard} = 0$ comes from the negative target generally being ranked/scored lower comparatively more than the positive target (the test accuracy is loosely related to the difference between the positive and negative target song ranks).

Given the relatively flat accuracy beyond $p_{hard} > 0.3$ and the optimal diversity for p_{hard} between 0.4 and 0.6, the model chosen for the Pandora online test was the $p_{hard} = 0.4$ model. This also trains with 30% fewer epochs, as an additional benefit in reducing training costs.

5.1. Online test

8M users are subjected the above model and the control experience over a one-week period. The control experience is a proprietary baseline model using collaborative filtering without individual negative user feedback. The tested $p_{hard} = 0.4$ model increased the key business metric (completed plays) by 0.8%. The recommendation accuracy, which is what the model is trained to optimize, increased by 2.4%. There was also an increase in recommendation diversity, as shown in Table 2.

6. Discussion

6.1. Positive feedback proportion

As the model is training to predict future positive feedback, it is possible that the model accuracy can vary depending on the composition of the input user's feedback. For both the Pandora and Spotify

Table 2
Online A/B test results for key business metrics (completed plays). The thumb-up rate is equivalent to the prediction accuracy and is what the model is trained to optimize. There is also an increase in the song diversity (i.e. decrease in inequality, as measured by the Gini coefficient). All results are statistically significant ($p < 0.01$).

	Online A/B test
Completed track plays	+0.8%
Thumb-up rate	+2.4%
Inequality (Gini) of track plays	−0.3%
Distinct tracks recommended per station	+1.0%
Distinct artists recommended per station	+0.7%

datasets, there is a slight accuracy increase for users with more positive feedback (Fig. 2). They manifest in slightly different ways: the Pandora dataset shows a relatively unchanged accuracy for $p_{hard} > 0$, and then a slight drop in accuracy for $p_{hard} = 0$; the Spotify dataset shows a relatively unchanged accuracy for $p_{hard} < 1$, and then a slight rise in accuracy for $p_{hard} = 1$.

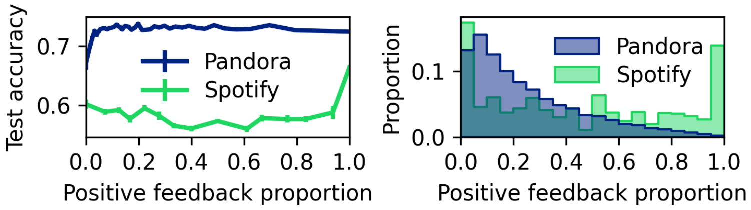

Figure 2: Accuracy is fairly robust with respect to the proportion of positive feedback per user (left). The distribution of feedback proportions differs by dataset (right).

This may reflect the different feedback types that comprise the data, as the Spotify dataset uses implicit positive feedback (plays) and implicit negative feedback (skips) whereas the Pandora dataset contains explicit positive and negative feedback. Users in the Spotify dataset who are almost all-play (i.e. virtually no skips) have scant negative feedback to use as negative targets, and may generally not want to (or not know how to) use the 'skip' button, so using their subsequent plays as the positive sample may overestimate their true preference for that song.

We note that the bimodal distribution of positive feedback proportions in the Spotify dataset (Fig. 2) aligns with prior research clustering user types into 'mostly-play' and 'mostly-skip' [5]. In contrast, the Pandora dataset (with explicit feedback only) shows that almost all users use some skips, and that most users' feedback is dominated by skips (in line with Table 1). Using implicit feedback like skips and/or plays allows for most users to be covered, though there may be some contexts in which users may not give skips as readily (such as while driving, or or when casting music at a party). In such contexts the recommendation accuracy may be expected to be lower while also being hard to quantify due to the lack of feedback.

Overall, for both datasets, the accuracy with respect to positive feedback proportion is fairly robust, which is promising for real-world implementation of this model without fear of hurting the experience for certain users. Further work should be done to better understand different user segments and to distinguish the differences between explicit and implicit positive feedback.

6.2. p_{hard} and positive/negative song ranks

Both positive and negative targets are scored/ranked lower as p_{hard} increases (Fig. 1c). This may be connected to the increase in diversity, as popular songs generally attract more feedback (both positive and negative) and are therefore more likely to occur as negative samples for nonzero p_{hard}. This would in turn lead the model to give these popular songs lower scores/ranks and thus recommend other, less popular songs. The model thus learns to recommend less popular songs in a personalized way for each user, leading to an increase in both diversity and accuracy (Table 2).

7. Conclusion

Incorporating real user feedback as negative targets allows for higher prediction accuracy as well as increased song diversity. This was evaluated via an offline simulation and then confirmed in an online A/B test. The model, which is trained on users with varying proportions of positive and negative feedback, is able to also cover users with only positive feedback, only negative feedback, and everything in between, with fairly consistent accuracy. The model diversity is particularly sensitive to the proportion of hard negatives used as negative targets, and more research should follow in how this optimal proportion may depend on dataset attributes.

References

[1] P. Seshadri, P. Knees, Leveraging Negative Signals with Self-Attention for Sequential Music Recommendation, arXiv preprint arXiv:2309.11623 (2023).

[2] Y. Wang, Y. Halpern, S. Chang, J. Feng, E. Y. Le, L. Li, X. Liang, M.-C. Huang, S. Li, A. Beutel, Y. Zhang, S. Bi, Learning from Negative User Feedback and Measuring Responsiveness for Sequential Recommenders, in: Proceedings of the 17th ACM Conference on Recommender Systems, RecSys '23, Association for Computing Machinery, New York, NY, USA, 2023, p. 1049–1053. URL: https://doi.org/10.1145/3604915.3610244. doi:10.1145/3604915.3610244.

[3] S. Chen, J. Chen, S. Zhou, B. Wang, S. Han, C. Su, Y. Yuan, C. Wang, SIGformer: Sign-aware Graph Transformer for Recommendation, in: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '24, Association for Computing Machinery, New York, NY, USA, 2024, p. 1274–1284. URL: https://doi.org/10.1145/3626772.3657747. doi:10.1145/3626772.3657747.

[4] M. J. Mei, O. Bembom, A. Ehmann, Negative Feedback for Music Recommendation, in: Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization, UMAP '24', Association for Computing Machinery, New York, NY, USA, 2024.

[5] F. Meggetto, C. Revie, J. Levine, Y. Moshfeghi, On Skipping Behaviour Types in Music Streaming Sessions, in: Proceedings of the 30th ACM International Conference on Information & Knowledge Management, CIKM '21, Association for Computing Machinery, New York, NY, USA, 2021, p. 3333–3337. URL: https://doi.org/10.1145/3459637.3482123. doi:10.1145/3459637.3482123.

[6] B. L. Pereira, A. Ueda, G. Penha, R. L. T. Santos, N. Ziviani, Online learning to rank for sequential music recommendation, in: Proceedings of the 13th ACM Conference on Recommender Systems, RecSys '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 237–245. URL: https://doi.org/10.1145/3298689.3347019. doi:10.1145/3298689.3347019.

[7] C. Hansen, C. Hansen, L. Maystre, R. Mehrotra, B. Brost, F. Tomasi, M. Lalmas, Contextual and Sequential User Embeddings for Large-Scale Music Recommendation, in: Proceedings of the 14th ACM Conference on Recommender Systems, RecSys '20, Association for Computing Machinery, New York, NY, USA, 2020, p. 53–62. URL: https://doi.org/10.1145/3383313.3412248. doi:10.1145/3383313.3412248.

[8] E. Mena-Maldonado, R. Cañamares, P. Castells, Y. Ren, M. Sanderson, Agreement and Disagreement between True and False-Positive Metrics in Recommender Systems Evaluation, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '20, Association for Computing Machinery, New York, NY, USA, 2020, p. 841–850. URL: https://doi.org/10.1145/3397271.3401096. doi:10.1145/3397271.3401096.

[9] M. Mansoury, H. Abdollahpouri, M. Pechenizkiy, B. Mobasher, R. Burke, Feedback Loop and Bias Amplification in Recommender Systems, in: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, CIKM '20, Association for Computing

Machinery, New York, NY, USA, 2020, p. 2145–2148. URL: https://doi.org/10.1145/3340531.3412152. doi:10.1145/3340531.3412152.

[10] A. V. Petrov, C. Macdonald, GSASRec: Reducing Overconfidence in Sequential Recommendation Trained with Negative Sampling, in: Proceedings of the 17th ACM Conference on Recommender Systems, RecSys '23, Association for Computing Machinery, New York, NY, USA, 2023, p. 116–128. URL: https://doi.org/10.1145/3604915.3608783. doi:10.1145/3604915.3608783.

[11] T. Wilm, P. Normann, S. Baumeister, P.-V. Kobow, Scaling session-based transformer recommendations using optimized negative sampling and loss functions, in: Proceedings of the 17th ACM Conference on Recommender Systems, RecSys '23, Association for Computing Machinery, New York, NY, USA, 2023, pp. 1023–1026. URL: https://doi.org/10.1145/3604915.3610236. doi:10.1145/3604915.3610236.

[12] H. Ma, R. Xie, L. Meng, X. Chen, X. Zhang, L. Lin, J. Zhou, Exploring False Hard Negative Sample in Cross-Domain Recommendation, in: Proceedings of the 17th ACM Conference on Recommender Systems, RecSys '23, Association for Computing Machinery, New York, NY, USA, 2023, p. 502–514. URL: https://doi.org/10.1145/3604915.3608791. doi:10.1145/3604915.3608791.

[13] Y. Zhao, R. Chen, R. Lai, Q. Han, H. Song, L. Chen, Augmented Negative Sampling for Collaborative Filtering, in: Proceedings of the 17th ACM Conference on Recommender Systems, RecSys '23, Association for Computing Machinery, New York, NY, USA, 2023, p. 256–266. URL: https://doi.org/10.1145/3604915.3608811. doi:10.1145/3604915.3608811.

[14] B. Brost, R. Mehrotra, T. Jehan, The Music Streaming Sessions Dataset, in: Proceedings of the 2019 Web Conference, ACM, 2019.

[15] W.-C. Kang, J. McAuley, Self-attentive sequential recommendation, in: 2018 IEEE International Conference on Data Mining (ICDM), IEEE, 2018, pp. 197–206. doi:10.1109/ICDM.2018.00035.

[16] M. J. Mei, O. Bembom, A. Ehmann, Station and Track Attribute-Aware Music Personalization, in: Proceedings of the 17th ACM Conference on Recommender Systems, RecSys '23, Association for Computing Machinery, New York, NY, USA, 2023, p. 1031–1035. URL: https://doi.org/10.1145/3604915.3610239. doi:10.1145/3604915.3610239.

[17] J. A. Hanley, B. J. McNeil, The meaning and use of the area under a receiver operating characteristic (ROC) curve., Radiology 143 (1982) 29–36.