Learning on Negative User Feedback in Next Item Recommendation

Artem Nekrasov

Faculty of Computer Science,
National Research University Higher School of Economics,
Moscow, Russia
ainekrasov@edu.hse.ru

***Abstract*—In this work, we propose a novel recommender model, gSASRec (Petrov and Macdonald, 2023) with an improved loss function—adopting the concept from "Learning from Negative User Feedback and Measuring Responsiveness for Sequential Recommenders"—to encode negative user feedback (Wang et al., 2023). Our method incorporates the merits of the gSASRec architecture and negative labels provided explicitly. Rather than the standard binary cross-entropy, we utilize an objective which maximizes explicitly the probability of "not recommending" products that were rated negatively.**

This modification renders the training process more sensitive to negative signals. Not only does the model capture user interests effectively, but also it suppresses contradictory recommendations more quickly. We carried out a set of experiments on several datasets, verifying that learning from positive and negative examples enhances recommendation quality, decreases the frequency of undesirable recommendations, and leads to more stable convergence. Furthermore, it provides flexibility in working with both explicit (e.g., "dislikes") and implicit (e.g., quick skips of content) negative feedback, enhancing user satisfaction and trust in the system.

I. Introduction

Our work is in the field of recommender systems that generate user-personalized lists of content based on sequences of user behavior. Over the past several years, such models have seen broad application as data volumes have increased and the demand for successful filtering of large catalogs of products, videos, articles, and other content has increased.

The problem we tackle is the inability to properly take into account negative feedback during recommendation generation. With negative signals, we refer to both explicit complaints and dislikes, and implicit hints like quick content skipping, which reflect the incompatibility between recommended items and the real users' interests. Failing to take into account such signals results in users still receiving non-relevant recommendations, decreasing their trust in the system and possibly damaging the overall user experience.

To solve this issue, we suggest a novel model derived from the gSASRec architecture with an improved loss function that aims at "not recommending" already rejected items. We draw inspiration from works on learning from negative feedback and also on analyzing how recommender systems react to negative feedback.

Our research follows efforts to adapt Transformers (Vaswani et al., 2017) to recommender tasks (e.g., SASRec (Kang and McAuley, 2018)), and investigations demonstrating that the explicit incorporation of negative samples (instead of regular binary cross-entropy) can enhance recommendation quality and users' general perception of the system. The novelty of our approach is that we not only log "unwanted" products, but also actively increase the chances of their exclusion from future recommendations. Consequently, the model becomes more sensitive to negative feedback and more responsive to shifting interests.

We anticipate that experimental findings will demonstrate how our adjusted loss function assists (1) in considerably minimizing repeat occurrences of objectionable content, (2) enhancing recommendation accuracy, and (3) building users' trust in the system.

II. Literature review

In the field of recommender systems that work with sequences of user interactions, one of the fundamental problems is still the correct prediction of the next item (product, song, or video), considering context and personal taste. Over the past several years, the research community has investigated numerous promising directions for better recommendation quality, from self-attention models (SASRec) to the use of users' negative signals for influencing the final recommendation lists.

One of the first well-known papers on the use of attention mechanisms for sequential recommendations is SASRec, introduced in the paper "Self-Attentive Sequential Recommendation" by Kang and McAuley. The authors proposed leaving out traditional recurrent networks (GRU, LSTM (Chung et al., 2014)) and convolutional architectures in favor of attention blocks similar to NLP transformers. The intuition is that, at every attention step, the model "balances" the importance of items interacted with previously, thus enabling more flexible consideration of the relevance of the next item. Experiments show that SASRec performs competitively on a range of datasets, including relatively sparse ones, since it adaptively "adjusts" to patterns in short- or long-term interactions and has high computational efficiency.

However, empirical application of SASRec manifested some limitations. For example, when negative sampling and standard binary cross-entropy are used, the model is prone to the ”overconfidence” issue, predicting excessively high probability scores for top-ranked items. In the paper ”gSASRec: Reducing Overconfidence in Sequential Recommendation Trained with Negative Sampling” (Petrov and Macdonald), the authors propose a modification of SASRec named gSASRec, where the loss function is adapted to pay more attention to negative examples. The key ingredient is Generalised BCE (gBCE), which dissuades excessively confident predictions and strengthens the model's ability to rank items in the right order. They demonstrate better accuracy (Recall, NDCG) on several publicly available datasets, with more stable training convergence.

One more crucial area of research is explicit and implicit user signals expressing the negative sentiment towards recommendations (dislikes, skipping tracks, etc.). As an example, the work ”Learning from Negative User Feedback and Measuring Responsiveness for Sequential Recommenders” (Wang et al.) illustrates that incorporating negative labels during training and explicitly decreasing the likelihood of recommending items the user has previously rejected can bring beneficial effects to the ultimate recommendations. The authors propose the idea of ”not-to-recommend loss” and outline an approach to estimating a system's ”responsiveness,” i.e., the speed with which the model removes recommendations for items that are close to rejected ones. Large-scale industrial experiments on big data sets demonstrated that systems trained with negative signals reduce the share of unwanted recommendations and improve user satisfaction.

To complement this subject is the paper ”Improving Sequential Music Recommendation using Negative Feedback-aware Contrastive Learning” (Seshadri et al., 2024) with a focus on music streaming. The authors point out skipped songs as an implicit negative signal and introduce an additional contrastive subtask that is intended to ”push apart” embeddings of songs a user skipped in the vector space while ”pulling closer” the songs viewed positively. This approach is a step up from the traditional approaches (e.g., GRU4Rec (Ma et al., 2019), SASRec, BERT4Rec (Sun et al., 2019), ALBERT4Rec (Petrov and Macdonald, 2022)) in that it introduces a novel InfoNCE-based loss function, thereby enhancing the relevance of recommendations and minimizing the likelihood of re-recommending tracks that have been rejected previously.

Overall, several key takeaways emerge from these works:

1) Self-attention architecture (SASRec) provides extremely high accuracy for dense and sparse datasets, which dynamically determines what to focus on from past items.
2) gSASRec variant addresses the ”overconfidence” problem through a novel loss function (gBCE) and through oversampling of negative examples so that ranking is more balanced.
3) Incorporating negative signals—both explicit and implicit—serves to improve quality metrics overall, from lowering the ratio of unwanted content to better Recall and NDCG. Both the loss function modification (not-to-recommend loss) and the use of contrastive sub-tasks for better structuring the embedding space are critical components.
4) These techniques are successfully applied to industrial services (music, video, etc.) where user conduct (skips, dislikes) offers more feedback, allowing adaptation to evolving tastes at a better rate.

III. Methodology

There were three datasets utilized in this study: MovieLens 1M (ML1M), RetailRocket, and RC15. The first one is already explained in the original gSasRec paper, but the last two contain user interactions. Events are divided into ”view” and ”addtocart” (”click”). In this case, ”view” was employed as negative feedback, and ”addtocart” (”click”) as positive. This added another context to the original approach, where in the basic version of gSasRec, negative samples were randomly selected via negative sampling. In this new configuration, every time a ”view” event was observed, that item was marked as potentially undesirable and needed to be accounted for more carefully during training so that the model would pay less attention to it when ranking.

Right before generating samples, a preprocessing step removed overly short sequences: any sequence that consisted of fewer than three interactions was filtered out (Xin et al., 2021). This restriction is required because gSasRec and other self-attention–based models need at least a bit of context in order to predict the next step accurately. It was also needed to eliminate edge cases whenever the user had clicked just once or put one item by itself, because these kinds of data rows cannot provide any significant training, particularly whenever negative labels are taken into account.

Following data preparation, the default gSasRec framework was taken. However, instead of simply changing the loss function, a concept of action embeddings was introduced. That is, along with the embedding of the item or the object itself, the model was also provided with a feature that represented whether a particular interaction was positive (”addtocart” or ”click”) or negative (”view”). This change detailed how logits were being computed: the model would not just take into account the item ID that the user watched or added but the interaction mode as well. Therefore, when the self-attention block was creating the ”weights” for the previous interactions, it could distinguish how much each item was leading to positive or negative feedback.

When sampling negative instances for RetailRocket or RC15, the model marked any products with an observed ”view” interaction as ”bad” candidates. In ML1M, as in the baseline method, negatives were sampled randomly, due to the fact that the original data lacks direct counterparts of ”view” and ”addtocart”/”click” events. Various strategies to the detection of negative signals were thus used across the datasets, offering a great chance to verify the flexibility of the suggested method.

Having formulated the feature sets and chosen the negative sampling approach, the gSasRec model was trained using a improved loss function. Furthermore, key hyperparameters—embedding size, regularization weights, number of self-attention layers, dropout rate, learning rate, sequence length, and negative-to-positive interaction ratio—were optimized. The goal was to prevent overestimation of top-position probabilities and enable more precise handling of undesired items in the model outputs via special interactions.

Following a coarse hyperparameter search initially, a finer grid search was conducted on a validation split of the data. The stopping criteria were metric value stability (Recall@k, NDCG@k) and failure to improve over several consecutive epochs. Ultimately, the best-performing configurations were established for each of the three datasets: ML1M, RetailRocket, and RC15. The study found that adding an action embedding, and once more re-selecting the negative sampling approach, made the model rank items more precisely, especially if a ”view” event explicitly indicated user disinterest. Thus, the use of action embeddings and consideration of the particularities of each dataset makes the solution more generic and versatile, minus gSasRec’s most essential advantage: adequately modeling sequential patterns through the application of the self-attention mechanism.

IV. Results

The performance on the ML-1M dataset was as follows: while the basic SASRec model obtains a Recall@10 of 0.247 and an NDCG@10 of 0.131, gSasRec raises these results to 0.300 and 0.176, respectively. Our extension, adding an action embedding to gSasRec, enhances Recall@10 slightly (0.301) without a difference in NDCG@10 (0.176). The results of this study validate that the methodology suggested here significantly enhances recommendation accuracy through the inclusion of a larger number of negative examples, thus enabling a more effective assessment of undesirable content. Overall comparison metrics are given in the following table:

Model	**Recall@10**	**NDCG@10**
SasRec	0.247	0.131
gSasRec	0.300	0.176
Our Model	**0.301**	**0.176**

TABLE I
Performance comparison of different models

V. Conclusion

The suggested method has been shown to have the ability to enhance the baseline gSasRec model’s performance by incorporating negative feedback better, either learned through implicit events (”addtocart” positive, ”view” negative) or via popular negative sampling. Our model balances the strengths of the self-attention mechanism, which elegantly deals with sequences of different lengths and calculates pertinent interactions, with the incorporation of additional action signals to form strong ranking against ”undesirable” content. Experiments on several datasets, such as ML-1M, RetailRocket, and RC15, validate that the extended loss function and addition of the special action embedding allow for better modeling of patterns in user behavior and handling more negative examples.

The results show not just the improvement of standard measures (Recall and NDCG), but also the increased capacity of the model to promptly ”dismiss” items a user is not interested in. The method we propose adapts according to the nature of the data obtained, blending traditional negative sampling techniques with novel approaches in a way that will more accurately accommodate user behavior into the models. The result is a recommendation list of improved quality with minimal undesired items being presented within the final output without any compromise to the convergence rate or accuracy.

References

Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. Available at: `https://arxiv.org/abs/1412.3555`.

Kang, W.-C. and McAuley, J. (2018). Self-attentive sequential recommendation. Available at: `https://arxiv.org/abs/1808.09781`.

Ma, C., Kang, P., and Liu, X. (2019). Hierarchical gating networks for sequential recommendation. Available at: `https://arxiv.org/abs/1906.09217`.

Petrov, A. and Macdonald, C. (2022). A systematic review and replicability study of bert4rec for sequential recommendation. Available at: `https://arxiv.org/abs/2207.07483`.

Petrov, A. V. and Macdonald, C. (2023). gsasrec: Reducing overconfidence in sequential recommendation trained with negative sampling. In *Proceedings of the 17th ACM Conference on Recommender Systems*, RecSys 23, pages 116–128. ACM. Available at: `https://arxiv.org/abs/2308.07192`.

Seshadri, P., Shashaani, S., and Knees, P. (2024). Enhancing sequential music recommendation with negative feedback-informed contrastive learning. In *18th ACM Conference on Recommender Systems*, RecSys 24, pages 1028–1032. ACM. Available at: `https://arxiv.org/abs/2409.07367`.

Sun, F., Liu, J., Wu, J., Pei, C., Lin, X., Ou, W., and Jiang, P. (2019). Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. Available at: `https://arxiv.org/abs/1904.06690`.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. Available at: https://arxiv.org/abs/1706.03762.

Wang, Y., Halpern, Y., Chang, S., Feng, J., Le, E. Y., Li, L., Liang, X., Huang, M.-C., Li, S., Beutel, A., Zhang, Y., and Bi, S. (2023). Learning from negative user feedback and measuring responsiveness for sequential recommenders. In *Proceedings of the 17th ACM Conference on Recommender Systems*, RecSys 23, pages 1049–1053. ACM. Available at: https://arxiv.org/abs/2308.12256.

Xin, X., Karatzoglou, A., Arapakis, I., and Jose, J. M. (2021). Supervised advantage actor-critic for recommender systems. Available at: https://arxiv.org/abs/2111.03474.

[1]

[1] word count = 1831