Steven Nowicki
CS 410
Technology Review
11/12/2022

Introduction

In the paper titled *Recommending What Video to Watch Next: A Multitask Ranking System,* researchers summarize their approach for creating a novel method for recommending video content on a large-scale video streaming platform. Given an input of a video that a user is currently watching and the context in which they are watching it, the recommendation system was designed to return a list of ranked videos that the user may like to watch next. In the publication, the authors discuss challenges they faced in creating the recommendation system, the model they used for ranking, and their method of experimentation, which involved live experiments on YouTube.

Training the Model

The researchers trained their model using user logs containing implicit feedback about video content. Implicit feedback in this study was categorized into two types: engagement behaviors, such as clicks and watches, and satisfaction behaviors, such as likes or dismissals. Researchers noted that, given the scale of the problem, i.e., millions of users streaming millions of videos, reliance upon the implicit feedback they could gain from user logs was the most practical and economical approach. Relying upon explicit feedback from users in something like a survey or scoring system would not have been feasible.

In constructing their training model, the researchers also spent a good deal of effort addressing the issue of positioning bias, wherein users are more likely to click a video that appears at the top of a recommendation list (and conversely less likely to click on a video at the bottom of a recommendation list), even if it is not relevant to them. This type of positioning bias was existent in the researcher's data set, and they took efforts to counteract it when developing their recommendation system. To do so, the researchers performed an analysis of their training data, comparing click through rates to recommendation position, to first prove that the bias existed. Once the bias was verified, they incorporated two methods of accounting for it in their model.

Developing the Recommendation System

Researchers applied what they called a Multi-gate Mixture-of-Experts architecture to create their recommendation system. Given a query of a video that is currently being watched – and the context in which it is being watched – the goal of the system is to return a list of relevant candidates, or videos, that the user may also like to watch.

This system is divided into two sections: candidate generation and ranking. In the candidate generation step, potential videos to recommend are gathered using various comparison algorithms. For example, a candidate may be a video that has a similar topic in its title. Context gleaned from metadata is also used, such as the type of device used to view the video, the time of day the video is viewed, and user demographics. During ranking, features of a candidate are then used as inputs to predict what actions a user might make on the candidate. Actions, here, are the same set of engagement behaviors and satisfaction behaviors present in the training data.

Experimentation

Once developed, the researchers tested their recommendation model in a live environment – YouTube. They compared their model to a baseline model currently in use and discovered that, based on engagement and satisfaction metrics, theirs was significantly better at recommending videos.

Model Architecture	Number of Multiplications	Engagement Metric	Satisfaction Metric
Shared-Bottom	3.7M	/	/
Shared-Bottom	6.1M	+0.1%	+1.89%
MMoE (4 experts)	3.7M	+0.20%	+1.22%
MMoE (8 Experts)	6.1M	+0.45%	+3.07%

In the table above, "Shared-Bottom" represents the baseline model against which researchers compared their MMoE model, and Number of Multiplications represents a measure of model complexity. Comparing equally complex models, MMoE improved video recommendation by engagement and satisfaction in both cases.

Conclusion

The recommendation system summarized above was notable not only for its results but also in its methods of development – namely, its focus on practical constraints. Throughout the study, the researchers were always conscious of what they could accomplish, given their inputs and the problem domain. The use of implicit training data gathered from user logs took advantage of a pre-existing data set that was practically free of cost, compared to manually gathering feedback from participants. The researchers were also realistic about training bias, addressing the specific instance of positioning bias and also acknowledging that many other forms of training bias may exist (while not falling into the trap of trying to eliminate all bias before even performing their experiment). Finally, when the researchers deployed their model in a live setting on YouTube, they were conscious of the scale of such an environment and made sure to keep their model "lightweight", as they described it, acknowledging that while a more complex model performing more calculations could potentially be more accurate, YouTube is an on-demand video streaming service where latency is paramount, and faster recommendation took precedence over perfect recommendation. Given all of the above, this publication struck a good balance between theory and practice, a balancing act that is at the heart of all good programming. What is more, because they kept their approach simple, and focused on discrete and measurable challenges and outcomes, the researchers were successful in creating a system that was better than the one currently in use.

References

Zhao, Z., Hong, L., Wei, L., Chen, J., Nath, A., Andrews, S., Kumthekar, A., Sathiamoorthy, M., Yi, X., Chi, E. (2019, September). Recommending What Video to Watch Next: A Multitask Ranking System. In Proceedings of the 13th ACM Conference on Recommender Systems (p. 43-51).