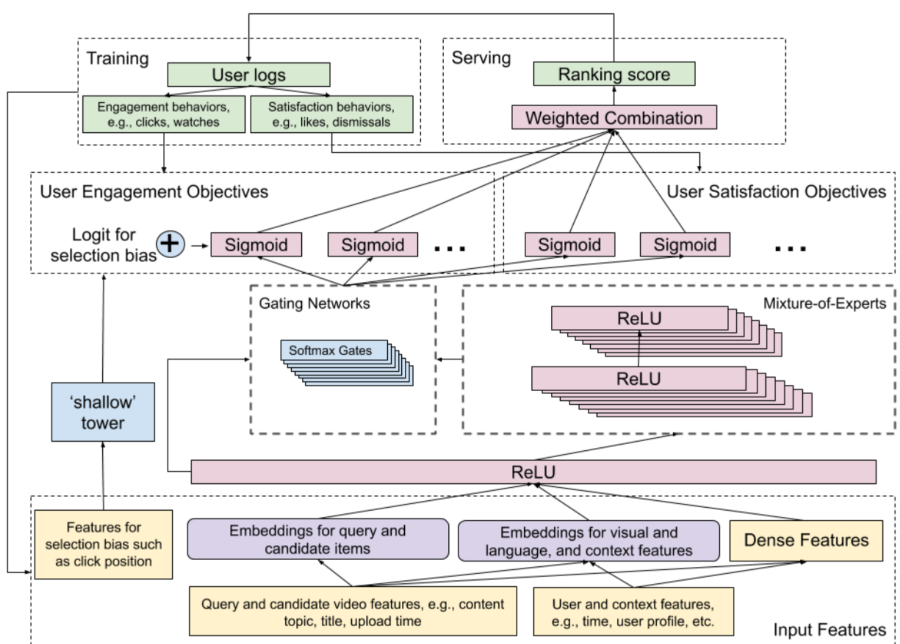
Evaluation of Google Multitask Ranking System

Introduction

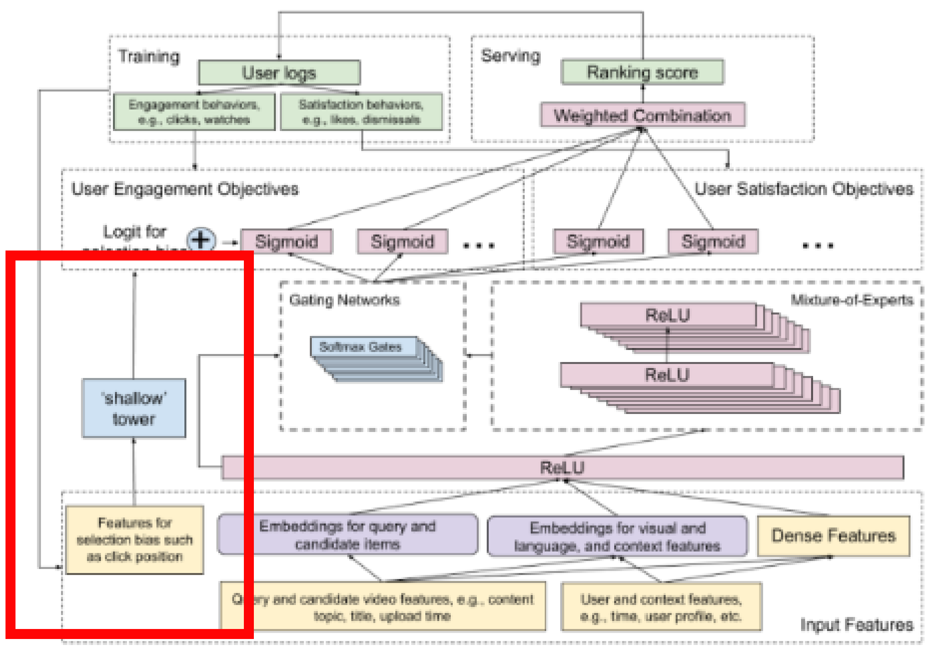
In this Tech review, I will briefly introduce the basics about Google Multitask Ranking System and evaluate its differences and similarities with other recommendation systems. Google Multitask Ranking System is a two-stage recommender system capable of multiple objective functions and removal of implicit bias.It is designed and developed for large-scale video recommendation on Youtube, tackling two main challenges: solving conflicting optimization objectives in ranking; eliminating implicit bias in the system during model training [1].

Mechanism of Google Multitask Ranking System

Google Multitask Ranking System utilizes Multi-gate Mixture-of-Experts(MMoE) to achieve multitask learning while introducing a shallow tower to deal with selection bias with the recommendation results, as shown in the figure below [1].



The MMoE is a combination of multi-layer neural networks connected to ReLU activations, where each of the experts in the layer is responsible for learning a unique feature from the user input. Then, the outputs of the MMoE layer move into Gating Networks, processed by all the objective functions. At this point, the developer of the Google Multitask Ranking System categorizes the objectives into two major parts: engagement and satisfaction [1]. Engagement objectives are the clicks, comments, views, and all other status of the recommended video that are related with the degree of engagement from other users. And Satisfaction objectives are the likes, rating, and sharing count of the recommended video that are correlated to the other users satisfaction. When training the model, these objectives find relevant objective functions that are represented by Sigmoid activation, and these objective functions decide whether to share these input objectives with other functions, forming a two way choice.



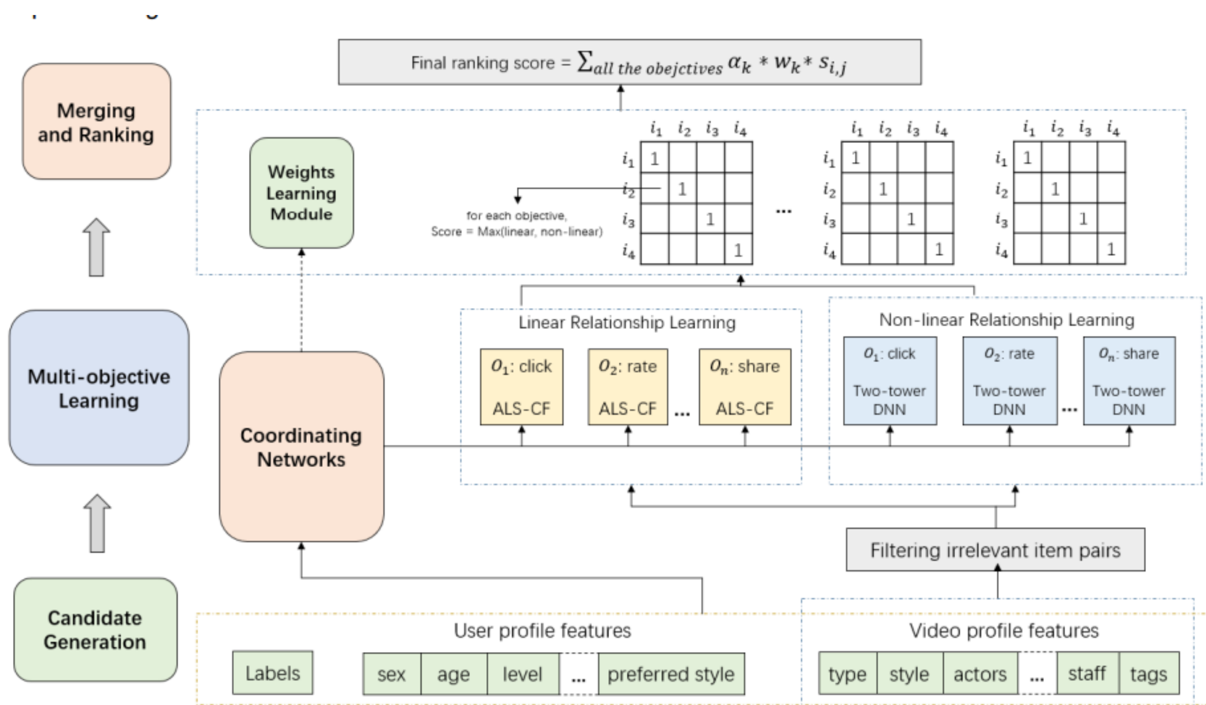
On the other hand, biases are handled at the left bottom part of the system diagram as shown in the figure above, where the key part of the user input is recorded and passed into the model. The system will ask the users for feedback on the recommendations. Though explicit feedback data is ideal for training the model, the developers of the Google Multitask Ranking System choose to use implicit feedback from the users to replace the expensive and merely unavailable direct feedback. Since a user clicking on the recommended video suggests that this recommendation is satisfying to some extent, this click data is used for recommendation feedback and is exactly where the bias comes in. Users may simply click on the videos on the recommendation list because they see some similar elements on the video cover image, the video has an intriguing title, or the video is just on the top of the recommendation list. In these cases, the bias needs to be eliminated. The Google Multitask ranking system has a shallow tower trained to sensibly react to features causing bias such as position in the recommendation list, potential bias components, and release time or view counts of the video. These biases are considered in the engagement objectives. The output from the shallow tower is processed back into the objective function in the user engagement parts for the model to learn and remove the possible bias [2].

Comparison with other recommendation systems

Google Multitask Ranking System is an efficient model for video recommendation. Modified from the Wide and Deep model architecture, the MMoE experts with the help of the shallow tower capture more matrices on engagement and satisfaction respectively while dealing with biases effectively [3]. As a recommender system, a Push model, Google Multitask Ranking System is simplifying user behavior data to the minimum and relieving the burden on data collections and retrievals. Similarly, Amazon is also simplifying the sample user data and the architecture of their recommendation system.

Amazon uses collaborative filtering in their item to item recommendation [4]. In the past decade, Amazon has shifted from user based collaborative filtering to item-to-item collaborative filtering. User-based collaborative filtering is focused on finding similar customers. Users of Amazon will be matched with other customers who have similar purchase histories where the system can gain recommendation from. On the other hand, the item-to-item based collaborative filtering emphasizes the correlations between products. In this case, data is still gathered from the user's purchase history. However, the information is now retrieved when this user buys item A unusually after buying item B, suggesting the correlation between these two items [4]. This shift of emphasis in collaborative filtering simplifies the data retrieval and training process from evaluating the purchase histories of all customers of Amazon and building up user models to evaluating the correlations between products and producing a list of potential related products. At the same time, it also reduces the time complexity of the model from finding a group of customers who have the similar purchasing behavior of the user to just finding the related products that the user is viewing. On average, a given product sold on Amazon is purchased by only a tiny portion of its customer while the customer base and their purchase history are huge amounts of data. After this change of focus, Amazon tried to improve the algorithm by emphasizing the likelihood of buyers of product A and B, adding factors like discounts, brands, and fashion styles, and adding time recommendations as learning targets. Now, Amazon's recommendation system is not only effective and efficient, but also widely applicable in other Amazon services such as Amazon Music and Amazon Prime Videos.

Apart from all these personalized recommendation systems, there is a multi-task learning algorithm for non-personalized recommendations provided by BILIBILI, a Chinese online video sharing platform, aimed to provide non-personalized video recommendations. The basic idea of this model is to improve the previous ranking list provided to all users of the website. The old ranking list consists of the videos trending on the website based on view counts, likes, sharing counts, and all the general status compared with all videos uploaded on the website in a certain period of time(usually on a weekly basis). Now, BILIBILI wants this list to become less general to avoid unhealthy competitions among content creators. The main concept of the algorithm is shown below. This recommendation system first creates multiple user profiles and video profiles, learns their correlations in both linear and non-linear basis, and merges them into matrix formats. As a result, the system outputs a list of partially generalized recommendations [5]. In this way, BILIBILI has avoided abusive personal data uses, bad competitions focusing only on view counts among content creators, and making use of trending lists for content promotion at the same time.



Conclusion

Overall, Google Multitask Ranking System is a simple and effective recommendation system for Youtube Videos. It has merely the simplest model architecture and the most effective way to deal with biases in user input. However, I think Google may consider taking elements from videos as the input to categorize videos, adding the video attributes to the recommendations, or even build up a new recommender system. The next step of the recommender systems could be taking both user input and attributes of elements being recommended at the same time.

References

[1]: Zhe, Z. Lichan, H. and 6 other co writers. *Recommending What Video to Watch Next*. (2019). Google. https://daiwk.github.io/assets/youtube-multitask.pdf

[2]: Suneet, B. *A Multitask Ranking System: How YouTube recommends the Next Videos*. (2020). https://medium.com/@bhatia.suneet/a-multitask-ranking-system-how-youtube-recommends-the-next-videos-a23a63476073

[3]: Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. *Wide & deep learning for recommender systems. In Proceedings of the 1st workshop on deep learning for recommender systems.* (2016). ACM, 7–10.

[4]: Larry, H. *The history of Amazon's recommendation algorithm*. (2019). Amazon. https://www.amazon.science/the-history-of-amazons-recommendation-algorithm

[5]: *A Multi-Task Learning Algorithm for Non-personalized Recommendations*. (2021). BILIBILI. https://openreview.net/pdf?id=xGZcxaYbJBF