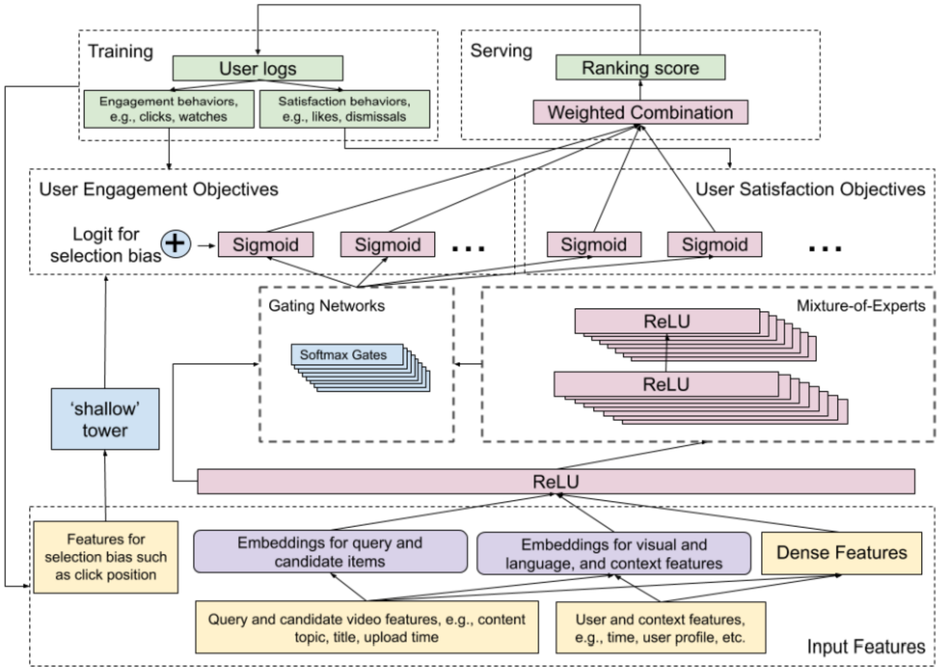
Intro

This paper is about a ranking and recommendation system for Youtube videos that has a complex machine-learning architecture. This review is an overview and explanation of this paper. In general, the system addresses scalability, conflict ranking objectives, and selection bias. The system uses some parameter techniques like Multi-gate Mixture-of-Experts. The paper also uses a variant of the Wide and Deep framework to reduce the bias, position bias especially. By using these methods, the system the paper proposes outperforms the previous system greatly and can be used in the scale of the Youtube platform.

Body


Training
User logs
Engagement behaviors, e.g., clicks, watches
Satisfaction behaviors, e.g., likes, dismissals
Serving
Ranking score
Weighted Combination
User Engagement Objectives
Logit for selection bias
Sigmoid
Sigmoid
...
User Satisfaction Objectives
Sigmoid
Sigmoid
...
Gating Networks
Softmax Gates
Mixture-of-Experts
ReLU
ReLU
'shallow' tower
ReLU
Features for selection bias such as click position
Embeddings for query and candidate items
Embeddings for visual and language, and context features
Dense Features
Query and candidate video features, e.g., content topic, title, upload time
User and context features, e.g., time, user profile, etc.
Input Features

Picture 1 architecture of the system[1]

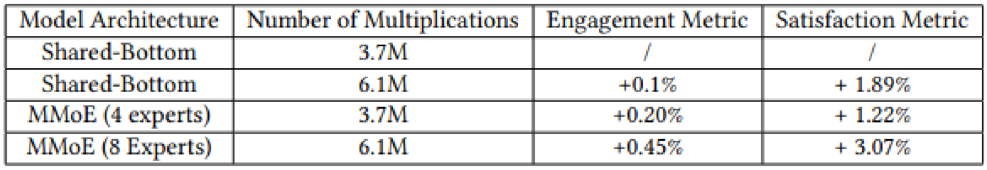
Here is the architecture of the system. This review will demonstrate it by following the order of execution of the system.

Before recommendation, the system will need a reasonable number of candidates since it is impossible to rank all the videos for each user on the scale of Youtube. Instead, the Youtube system uses a combination of candidate-choosing algorithms to select a few hundred candidates. Some of them can be as simple as logistic regression over different features of the video. Some are based on the correlation of other videos based on the frequency of watching together.

The next step is to rank these candidates. One of the important issues this paper address is how to solve the conflict between different objectives. There are two main groups: Engagement objectives(user actions like clicking) and Satisfaction objectives (likes, shares, etc.). The solution for this conflict is Multi-gate Mixture-of-Experts. The idea of this solution is to have multiple experts(subnetworks), each corresponding to a specific task. These experts will also share the bottom features to reduce the parameters. However, some tasks can be beneficial from other tasks. To address this problem, the authors use gates to control this feedback. They expect the architecture to learn which tasks can be beneficial to the current tasks and uses the weights of the gates to control them.

The next problem is about implicit bias, position bias especially. Sometimes, users click the recommended video because it is the first recommendation on the list. It does not mean that the user likes this video. If the system ignores this bias, the training result will be highly biased. To address this, the authors used a shallow tower. It is used to train the bias terms from features like recommendation positions. During the test time, the system will fix the position and use that to reduce the bias.

Conclusion



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%

Picture 2 performance [1]

Based on the chart, we can see that Multi-gate Mixture-of-Experts effectively reduces the number of parameters and improves the metrics. The paper also uses a relatively simple shallow tower to minimize bias. Following that, the authors performed experiments on Youtube to prove that the recommendation is better in the real-world case.

Citation

[1]Zhe Zhao, Lichan Hong, Li Wei, Jilin Chen, Aniruddh Nath, Shawn Andrews, Aditee Kumthekar, Maheswaran Sathiamoorthy, Xinyang Yi, and Ed Chi. 2019. Recommending what video to watch next: a multitask ranking system. In Proceedings of the 13th ACM Conference on Recommender Systems (RecSys '19). Association for Computing Machinery, New York, NY, USA, 43–51. https://doi.org/10.1145/3298689.3346997