Evolution of YouTube's Recommendation Systems

Introduction

This technology review focuses on the various recommendation and ranking systems developed by Google to power their Youtube platform, by analyzing their publicly disclosed research papers. In the following sections, high-level design and architecture for each system are discussed, where the relevant background information is provided prior to each section for aid in understanding. In the conclusion section, the ranking systems are summarized with mention of how the systems evolved over a decade. The final section provides a list of references to the systems and relevant information mentioned in this review.

Ranking Model in 2010

Given that YouTube was founded in 2005, the recommendation system proposed in [2] was one of the earliest and oldest that once powered the massive video platform. Before getting into the system details, it is useful to understand a key element of the system: association rule mining [6]. The idea with association rule mining is, given an N by P data vector where N is the number of observations and P is the number of features, the output is the set of features that show up together and/or are correlated. This concept is analogous to the discovery of syntagmatically related words in information retrieval.

The input to the recommender were two types of data: the content data, pertaining to video metadata such as its title and description; and user activity data, pertaining to how a user interacted with a video, such as by commenting or liking a video. The user's activity data was used to set the seed for the system where the set of videos related to a user in some way, whether it by them commenting, liking, sharing, or searching a video, were in the seed. Then, the system applied association rule mining for each seed video to determine the set of videos that were related to it and so candidates for recommendation which would later be ranked. If only videos that were directly related to a user were considered, the set of candidate videos would be relatively shallow and focused, thus the system considered videos up to a certain distance measure away from a seed video.

With the candidate video set determined (based on a time period, e.g. last 24 hours), they served as input to the ranking system where they were scored on video quality, user specificity, and diversity of the videos. The algorithm considered features like the number of views for a video in regards to video quality, and the video watch time by a user for user specificity. A simple linear combination of these factors was taken with weights to balance exploration/diversification with exploitation. The set of videos recommended to the user were simply the top N scoring of candidate videos that met a minimum score threshold.

Ranking Model in 2016

The recommendation system described in [5] was developed in 2016 and took a different approach than the previous. The system relied on deep neural networks [1] for both its generation of candidate videos to recommend to a user and for ranking the candidate videos. A high-level

description of a deep neural network requires the definition of a neural network: a collection of nodes, arranged in layers (input, hidden, output), that process a bunch of input signals (features) and output a value that is fed to the next layer. The idea of a node is a process that mimics a human neuron using a number of bodily signals to possibly emit a response. A deep neural network can be generalized as a neural network that has many hidden layers and is purposed to discover global knowledge to a high degree of effectiveness. The video recommendation system in this section relies on ReLu's [3], or rectified linear units, which is a choice of the activation functions that nodes use to decide, given the signals, whether or not to emit a response signal.

In the 2016 system, there were two parts: the candidate generation and ranking. The input to the candidate generation was the entire corpus of videos and a given user's activity history where the output contained several hundred videos. The candidate generation applied collaborative filtering via a feed-forward deep neural network; the end goal was classification among millions of video classes. The shape of the data fed to this neural network, which consisted of a few layers of ReLus, was in the form of embeddings: dense vectors containing several vectors of user and video data. The output/task was to find learned embeddings used to discriminate videos based on the class assigned to them. Despite millions of classes, given the need for scalability/speed requirement, a key observation was that only the top N should be returned, thus only N classes are needed. Then, The problem, reduced to the N nearest neighbours, or candidates, given the output embedding from the neural network.

The ranking system took the candidates from the previous step and generated scores for those videos with a similar structured deep neural network. In this step, additional features were used, such as video impression data, to fine-tune the predictions on whether a user may watch a video and for what reason (e.g. top of the list, or interest). There was a significant amount of work in feature tuning, handling of continuous variables, treatment of factor/categorical variables, normalization, etc. prior to the DNN to arrange them in a sequence of activities the user engaged in with their relation to the videos. Weighted logistic regression was then performed with the DNN and the top scoring videos were returned.

Ranking Model in 2019

The latest recommender system developed to power YouTube was released in 2019 [7]. To understand how it works requires the understanding of the core ML framework it adopted known as Multi-Gate Mixture of Experts (MMoE) [4]. Essentially, the goal of MMoE is to optimize multiple objectives by learning individual objectives and leveraging shared information among objectives. MMoE consists of multiple experts, each feeding to a gating network that determines the best weightings for the experts in making an optimal prediction. The output from those gating networks can then be fed into other components as needed depending on the application, which itself may be another model. An expert is a neural network that optimizes for predictions in a subdomain or subtask of the grander task of maximizing a given objective; it can be thought of as the effectiveness of a feature for a particular range in a dimension. Note that this model is quite powerful in that the task/objective similarity can change how much the experts are shared via the gating networks: less similarity results in penalization of expert sharing, thus gating networks learn to make use of different experts.

In this version of YouTube's recommendation system the candidate generation is not discussed in detail, instead, the paper focuses on the ranking portion. The model involved multiple objectives that were broken down into two categories: user engagement such as clicks and user satisfaction such as a user liking a video. The input to the system was the candidate set of videos and user logs, from which rich vector embeddings were generated containing compact representations of user data such as their watch history. The model considered each video individually, and took a pointwise learning approach instead of pairwise (e.g. collaborative filtering) for efficiency due to the scale at which YouTube operates. The goal of the model was to learn how to rank by predicting the probabilities of a user's actions such as dismissing a video. The previously mentioned input was piped to a shared base layer among all experts in the model to reduce the overall training cost, which then fed into the MMoE architecture that trained multiple experts, each of which output to a gating network for another layer of predictions. The gating networks assigned weights to the experts, determining which were useful, where the output weights were fed into different objective functions that applied a sigmoid activation function. Prior to the final predictions which took the objective function output with a weighted combination to compute a final score, a shallow deep neural network model was simultaneously trained to reduce bias. The bias model considered a subset of the original input features and attempted to predict whether things such as where a user clicked had any effect on a user's engagement. The output for the bias model was fed, specifically, to the engagement objective functions. Finally, with the video ranks computed, the top scoring videos were recommended to the user.

Conclusion

YouTube is a video platform that has billions of users and billions of videos in its corpus. The problem of recommending videos to users is critical in keeping those users engaged and satisfied in order to retain them, making the recommendation system play a critical role in the business. Over the years, the recommendation system has evolved many times, from straightforward algorithms to multi-level deep neural networks. The system has tried multiple different approaches such as collaborative filtering, pointwise ranking, and raw matrix multiplication. What has remained the same throughout the systems is the separation of candidate video generation and the ranking of the candidate videos. However, the methods for each stage have increasingly become more complex with application of multiple algorithms and models at each step. The scale at which YouTube operates has presented many challenges in the designing of the system, particularly with the number of features and data available making recommendations one of the most volatile and exciting systems in the company, and an ongoing problem in academia.

References

[1] A. K. Listlink, “Deep Neural Networks,” *KDnuggets*, Feb-2020. [Online]. Available: https://www.kdnuggets.com/2020/02/deep-neural-networks.html. [Accessed: 31-Oct-2021].

[2] Davidson, James & Liebald, Benjamin & Liu, Junning & Nandy, Palash & Vleet, Taylor & Gargi, Ullas & Gupta, Sujoy & He, Yu & Lambert, Michel & Livingston, Blake & Sampath, Dasarathi. (2010). The YouTube video recommendation system. 293-296. 10.1145/1864708.1864770.

[3] J. Brownlee, “A gentle introduction to the rectified linear unit (ReLU),” *Machine Learning Mastery*, 20-Aug-2020. [Online]. Available: https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/. [Accessed: 31-Oct-2021].

[4] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H. Chi. 2018. Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-of-Experts. *In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '18)*. Association for Computing Machinery, New York, NY, USA, 1930–1939. DOI:https://doi.org/10.1145/3219819.3220007

[5] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep Neural Networks for YouTube Recommendations. *In Proceedings of the 10th ACM Conference on Recommender Systems (RecSys '16)*. Association for Computing Machinery, New York, NY, USA, 191–198. DOI:https://doi.org/10.1145/2959100.2959190

[6] S. Remanan, “Association rule mining,” *Medium*, 02-Nov-2018. [Online]. Available: https://towardsdatascience.com/association-rule-mining-be4122fc1793. [Accessed: 31-Oct-2021].

[7] 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. DOI:https://doi.org/10.1145/3298689.3346997