Combining Models for Better User Satisfaction in Video Recommendation

Josef Florian
josef.florian@firma.seznam.cz
Seznam.cz
Prague, Czech Republic

Jakub Drdák
jakub.drdak@firma.seznam.cz
Seznam.cz
Prague, Czech Republic

Radek Tomšů
radek.tomsu@firma.seznam.cz
Seznam.cz
Prague, Czech Republic

Karel Koupil
karel.koupil@firma.seznam.cz
Seznam.cz
Prague, Czech Republic

Václav Blahut
vaclav.blahut@firma.seznam.cz
Seznam.cz
Prague, Czech Republic

Jaroslav Kuchař
jaroslav.kuchar@firma.seznam.cz
Seznam.cz
Prague, Czech Republic

Michal Řehoř
michal.rehor@firma.seznam.cz
Seznam.cz
Prague, Czech Republic

ABSTRACT

Watch time has been a subject of interest for recommender systems in recent years. Music, podcast and video recommendations based on or amplified by consumption time optimization are often employed to boost perceived user satisfaction, subscriptions, engagement or to decrease number of bounces. Finding a fragile balance between several different metrics describing users' behaviour might be challenging, especially in the area of video recommendation. In this paper, we design online algorithms for modelling relationship between click probability and expected watch time on a video. We explore means of combining and balancing click-based and watch time optimizing models in an online multi-criteria setting. We present experiments that involve watch time, CTR and watch ratio. Furthermore, the paper describes empirical evaluations on live traffic and illustrates that our approach has succeeded in outperforming a non-trivial baseline in a controlled manner.

CCS CONCEPTS

• **Information systems → Recommender systems**; • **Computing methodologies → Machine learning**.

KEYWORDS

multi-criteria optimization, Dwell time, video recommendation

Reference Format:
Josef Florian, Jakub Drdák, Radek Tomšů, Karel Koupil, Václav Blahut, Jaroslav Kuchař, and Michal Řehoř. 2020. Combining Models for Better User Satisfaction in Video Recommendation. In *3rd Workshop on Online Recommender Systems and User Modeling (ORSUM 2020), in conjunction with the 14th ACM Conference on Recommender Systems, September 25th, 2020, Virtual Event, Brazil.*

ORSUM@ACM RecSys 2020, September 25th, 2020, Virtual Event, Brazil

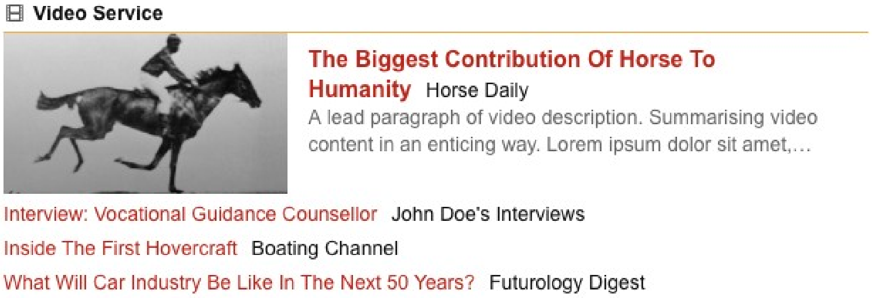
1 INTRODUCTION

Recommendation systems play a significant role in today's world of information overload. Users have only a limited amount of time they want to spend on consuming online content. Therefore media providers are striving to deliver the most relevant content. As Attfield et al. mentioned in [3], user engagement is the emotional, cognitive and behavioural connection that exists, at any point in time and possibly over time, between a user and a resource. We also believe that user engagement utilization has become crucial for any recommender system. In our use case, users are exposed to a personalized selection of online media services, including news, video content, lifestyle, sport, weather forecast and celebrities etc.

In this paper, we address a problem of recommending videos for a content box, shown in Figure 1. Users rarely provide explicit ratings or direct feedback when consuming frequently updated online content. Optimizing algorithm for CTR metric (Click Through Rate, more in part 4) is straightforward, however, it does not capture any post-click user behaviour and might lead to promoting click-bait articles and low-quality content. On the other hand, dwell-time, the time spent on a web page is one such metric and has proven to be a meaningful and reliable metric of user engagement in the context of recommendation tasks [11]. Instead of using the dwell-time on the video page as a whole, we decided to utilize only the time spent watching the video itself, measured via the video player. Favourably, we know the time duration of the video content, we can compute relative watch time w.r.t. length of the video.

Our main KPIs (Key Performance Indicators) for users' satisfaction include total watch time (TTS = total time spent) users have spent on a video service together with CTR. Our objective is to maximize user engagement while still considering number of video views.

During experiments, we try first to proxy user satisfaction only via watch time, although this approach leads to a vast harm on CTR. Then we continue with finding a balanced trade-off between CTR and watch time on a video while maximizing users' TTS and reducing harm of CTR. For this purpose, we model the relationship



Video Service

The Biggest Contribution Of Horse To Humanity Horse Daily

A lead paragraph of video description. Summarising video content in an enticing way. Lorem ipsum dolor sit amet,...

Interview: Vocational Guidance Counsellor John Doe's Interviews

Inside The First Hovercraft Boating Channel

What Will Car Industry Be Like In The Next 50 Years? Futurology Digest

Figure 1: Picture representing design and dummy content of a box on the content platform home page. There is always a corresponding image at the first position. Box has 4 positions by default and gets extended by 3 more positions each time a user clicks on any item in the box except for the first one. We used a position feature as one of the signals in training phase to mitigate bias arising from the presence of the picture at the first position.

between CTR and watch time using either parameterized multiplication or parameterized exponential function.

In this paper, we make the following contributions:

- We suggest two different approaches for modelling the relationship between click probability and watch time or watch ratio respectively (Section 3),
- we propose a solution based on generalized linear models suitable for an industry setting (Section 4),
- we evaluate our approach on live traffic (Section 4).

2 RELATED WORK

Several related papers have already analyzed a fundamental question: How to measure time users spend consuming content (also referred to as dwell time or, in our case, watch time) and how it can be utilized in the context of personalization. Yi et al. in [11] explore item-level dwell-time as a proxy of relevance of a content item to a particular user and argue that the amount of time that users spend on content items is an important metric to measure user engagement and should be used as a proxy to user satisfaction for recommended content, complementing and/or replacing click-based signals.

Agichtein et al. [2] has achieved substantially higher precision and recall by considering the comprehensive *UserBehaviour* features that model user interactions after the search and beyond the initial click rather than considering click-through alone.

Kim et al. [6] study relation between user satisfaction and *click-dwell time* (i.e. the time user spends on a clicked result) in terms of web search. They argue that employing previously used fixed duration threshold [4] [10] such as 30 seconds to determine user (dis-)satisfaction is ineffective due to variety of *query-click attributes*, factors such as query type, page topic, page content, and page readability level. Instead, they model (dis-)satisfied clicks' watch time distributions w.r.t. to different segments of clicks and show how such features can improve prediction performance.

Authors of this paper use TTS, which depends on a length of the videos consumed by users, as one of KPI metrics. On the contrary, Lagun et al. in [7] have metrics which do not depend on the amount of content item has, but instead on the proportion of item consumed by users, making it easier to compare item with different amount of content.

There are various methods of combining multiple, often conflicting, objectives to train the model with. Rodriguez et al. in [9] use *semantic match*, a prediction from an existing recommender system trained on CTR, as the base for the final recommendation, which is then boosted by linear combination with indicating coefficients of additional binary relevance features. Finding optimal values for the coefficients is thereafter an optimization problem, for a small number of additional features shown to be solvable via grid search. More recently, Zhao et al. [12] focused on recommending what video to watch next on a large-scale online video-sharing platform, similar to the setup of this paper. Their multimodal objectives are split into two groups, *engagement* (such as clicks or watches) and *satisfaction* (likes, ratings). Each of those objectives is represented as one of Multi-gate Mixture-of-Experts [8] outputs, which are later combined using weighted multiplication, with weights manually tuned for a desired trade-off between each of the objectives. Diversity, novelty, or aspects of fairness have also been considered as features in multi-objective optimization [1] [5].

3 ADDRESSING MULTI-CRITERIA OPTIMIZATION

To maximize TTS, we need to attain fragile balance between videos with high CTR and the ones having potential to gain high watch time. If one naively tries to increase TTS, for instance by recommending very long videos without considering CTR, they may not be watched and final TTS may suffer. On the other hand, less-relevant content might acquire high CTR due to e.g. gratuitously catchy title, which results in early leaves, leading to loss of TTS as well.

We consider several methods that consist of *single models* that are combined and compared later to *compound models*. The *single models* are based on generalized linear models (GLM). Specifically, we experimented with the following models:

Log logistic regression (click probability)
Lin linear regression (watch ratio and watch time)
Poiss Poisson regression (number of watched parts)

The **Log** model is a GLM with log-odds link function. The model considers clicks that resulted in played videos as positive examples. All other recommended videos are considered as negative examples. The **Lin** models are GLMs with identity link functions. In the former case, the training samples have labels representing the ratios of consumed times for target videos w.r.t. their durations. In the latter case, the labels represent the total watched time in seconds for the target videos. The **Poiss** model is a GLM with natural logarithm link function. The labels are the number of watch parts sent by the video player every 10 seconds when a video is being played.

We try to use certain combination of these *single models* and optimize them both on the click probability and on the watch time spent on a particular video. Let us assume user $u \in U$ and video $v \in V$. Having the click probability c and the estimation of watch time t, we examine the following ways of combining c and t to obtain the resulting score:

$$s_1(c, t, \alpha) = c^{\alpha} \cdot t^{1-\alpha} \tag{1}$$

$$s_2(c, t, \beta) = t^{(c^{\beta})} \tag{2}$$

Variable c represents probability, thus its values are bound between 0 and 1. Variable t represents watch time on some video and its value can be any non-negative real number. Parameter α regulates the trade-off between c and t. It makes sense to restrict α between 0 and 1. Similarly, β regulates the importance of c and t for equation (2). We restrict β to be any non-negative real number.

We can assume the following function domains, $\mathcal{D}_{s_1} = [0, 1] \times [0, +\infty) \times (0, 1)$ and $\mathcal{D}_{s_2} = [0, 1] \times [0, +\infty) \times (0, +\infty)$. It can be easily seen that both functions (1) and (2) are growing in both c and t on given domain interiors. While function (1) grows like a power function of α, function (2) grows exponentially for fixed t, thus it is more sensitive to subtle differences in CTR (see Figure 2).

Having scores for the particular user and all recommendation candidates, we sort these videos by the score in descending order. Top N videos are then recommended to the user.

4 EXPERIMENTS

Seznam.cz is a Czech technology company specialized in internet-related services with a multitude of products. More than 95% of Czech internet users visit a Seznam website every week. Its products include a web portal, search engine, news service, email, advertising platform or map service with interactive panoramas of streets, rural roads and parks.

Televize Seznam is a video service available via a web browser, both mobile & smart TV apps and digital terrestrial, cable & satellite broadcasting. It holds 110,000+ videos with 9000+ hours of content. Our paper focuses on Televize Seznam's recommendation box on Seznam.cz's web portal with its 3.5 million unique users daily.

Our platform has to manage 4000+ requests per second at peak hours with sub-100-millisecond latency for Seznam.cz's web portal box. To provide personalized recommendations to our users, we mainly use the Vowpal Wabbit machine learning system. With a help of subwabbit[1] library, our application server (based on Python Tornado framework) delivers the recommendations via back end. Subsequent events of users are queued in Kafka. Data are stored in Couchbase, MariaDB, ElasticSearch and OpenStack Swift.

4.1 Data for Experiments

Our data contains detailed information about user features and videos, recorded at the time of recommendation. Videos among others contain information about the channels they are published from. They can have several human-generated tags. Each video has a duration and time of publication. User features consist of information about previously watched videos, user sex, age and a user profile created from the user's history collected from visits to other related web sites. The data also contains user interactions: whether a user clicked, whether a user started playing a video and for how long the video was played.

In a summary, over course of each week, the data contains information about 1.5M active users, 2.5M clicks, 700 recommendable videos. Our catalogue of videos is very diverse in a sense that it contains both minute- and hours-long videos, highly attention attractive videos, news videos, online TV-series and also full movies.

4.2 Setup and Evaluation

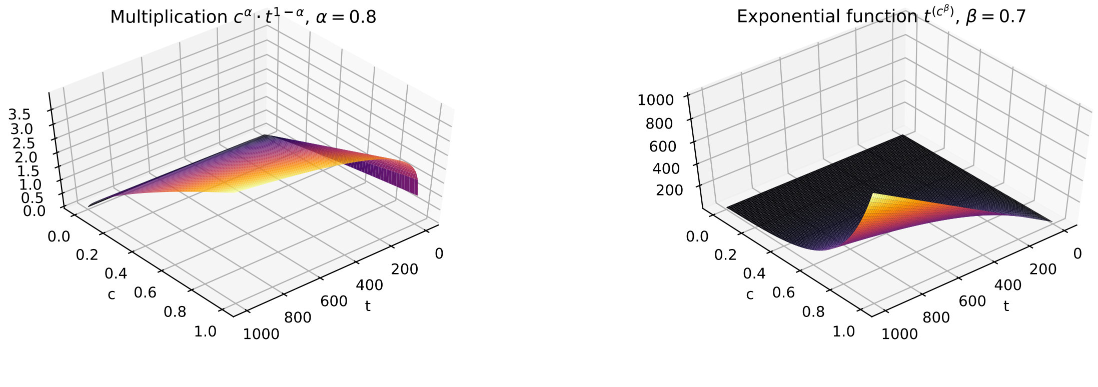
Single models are trained from the combinations of video and user features in order to learn the affinity between the users and the videos. *Compound models* combine outputs from *single models* using combination functions (1) and (2). From now on we'll call a *compound model* using function (1) as *α-compound model* and a model using function (2) as *β-compound model.* In order to explore the relationship between the hyper-parameters and the metrics we perform a grid-search over the subset of values of α and β.

For initial training of new models we use data logged in the last 3 hours. According to our experiments, such new models then perform with acceptable initial KPIs. After the initial phase we train the models incrementally online. During the incremental learning we regularly retrain the models every 5 minutes where the video-user interactions are considered for the training only when the user did not interact with a video for at least 30 seconds. In this setting it is possible to train a user's updated interaction multiple times, eg. if the user starts watching a video, pauses the video for a while and then continues watching the video.

In order to assess performance of the models, we perform A/B/n tests on live traffic and for each variant we additionally run A/A test. In the initial phase we use data logged from a control variant to train all the new models. To mitigate the interference between the variants, we train the models in the incremental phase only from the respective A/B/n variant. We use several metrics that should encompass both user engagement and business performance. We measure the number of video views (Views), total time spent watching videos (TTS) and averaged video duration (VD). The last metric that we use is CTR, which is the total number of clicks divided by the total number of pageviews.

The logistic regression model **Log** modelling a probability that a user clicks on a video is considered as a baseline model to which all the results are compared to. In the first experiment we compare performances of *single models* optimizing time metrics. These models are trained from positive feedback only, we do not use not-clicks. The results of the experiment are given in Table 1. All results in the table (and in the following tables) show relative changes compared to the values in first rows.

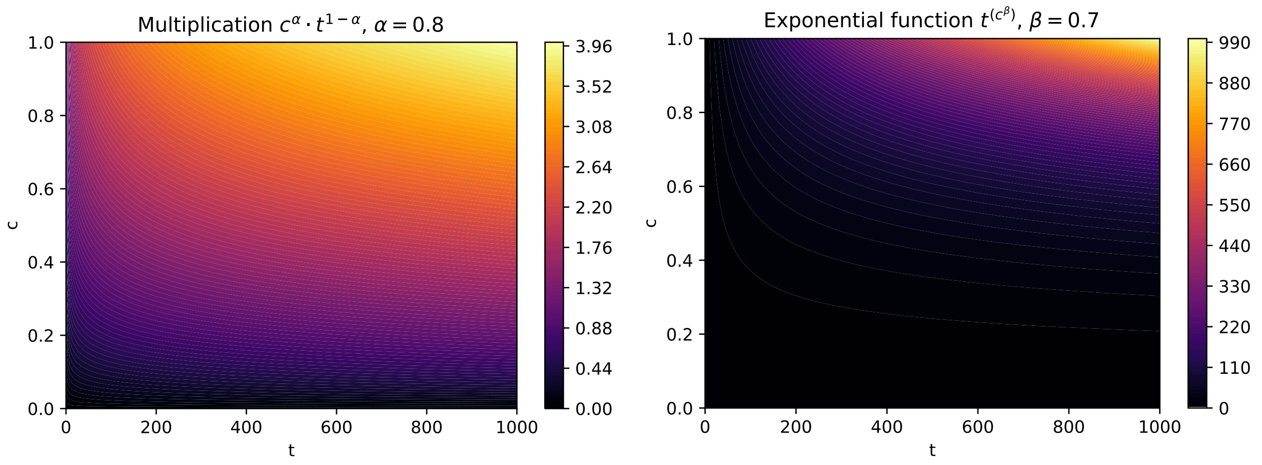
[1] https://pypi.org/project/subwabbit/



(a) Multiplication (1)

(b) Exponential function (2)

Figure 2: Shape of the non-linear manifold generated by equations (1) and (2) using our best performing α and β (see Table 4)



(a) Multiplication (1)

(b) Exponential function (2)

Figure 3: Contour plot of scores generated by equations (1) and (2) using our best performing α and β (see Table 4)

Model	CTR	$Views$	TTS	VD
Log	1	1	1	1
Lin-WR × VD	0.45	0.33	0.61	2.77
Lin-TTS	0.51	0.31	0.77	4.09
Poiss-TTS	0.52	0.50	0.51	0.92

Table 1: *Lin-WR* × VD is a model learning and predicting watch ratio of a video and the prediction is at the end multiplied by the video's duration, *Lin-TTS* and *Poiss-TTS* both learn and predict video watch time.

The models perform differently regarding the individual metrics. For example, **Lin-TTS** boosts four times longer videos than the baseline CTR model. This results in the highest TTS and the smallest $Views$ from the benchmarked models. **Lin-WR** × VD model recommends on average shorter videos than **Lin-TTS** but still more than two times longer than the baseline. The last model in this experiment, **Poiss-TTS**, performed well in CTR and $Views$ but scored the worst in TTS.

The next two experiments display the results of *compound models*. All the *compound models* use the **Log** model as c in the combination functions (1) and (2). Since we wanted to improve user engagement with the emphasis on moderately long and diverse videos, we concluded to use **Lin-WR** × VD as a model representing t in the combination equations. This model gave consistent results over time and did not emphasize too long videos compared to the rest. The Tables 2 and 3 display the results of a grid-search over the

Model	CTR	$Views$	TTS	VD
Log ($\alpha = 1$)	1	1	1	1
$\alpha = 0.8$	0.97	0.85	1.33	1.55
$\alpha = 0.6$	0.72	0.56	1.28	2.55
$\alpha = 0.4$	0.55	0.4	1.15	3.46
$\alpha = 0.2$	0.44	0.3	0.95	3.88

Table 2: Results of α-*compound models* for various levels of α

Model	CTR	$Views$	TTS	VD
Log	1	1	1	1
$\beta = 0.1$	0.52	0.35	1.00	3.33
$\beta = 0.4$	0.76	0.61	1.21	2.17
$\beta = 0.7$	0.85	0.72	1.19	1.71
$\beta = 1$	0.87	0.77	1.16	1.54

Table 3: Results of β-*compound models* for various levels of β

subset of values of hyper-parameters α and β. In both cases we can observe concave trend in TTS metric.

As the experiments are performed during different time periods, we perform one more experiment that uses the best performing models from the previous runs to verify our observations. This is necessary due to the shifts in data – completely different sets of videos are recommended at different times, user behaviour may change substantially, etc. The best performing α-*compound model* was the one with the value $\alpha = 0.8$. This model has the highest CTR, $Views$ and TTS. The best β-*compound model* is harder to select. From TTS perspective, the best results are obtained for $\beta = 0.4$, however CTR and $Views$ metrics are substantially worse than for $\beta = 0.7$. For this reason we selected the latter model for the final comparison. The results are given in Table 4.

Model	CTR	$Views$	TTS	VD
Log	1	1	1	1
$\beta = 0.7$	0.885	0.791	1.279	1.721
$\alpha = 0.8$	0.916	0.819	1.261	1.605

Table 4: Experiment comparing the best performing models from the previous experiments.

4.3 Discussion

Our goal was to maximize user engagement which we represented by the combination of metrics: TTS and CTR. Since we did not succeed at maximizing all the metrics at once, we focused on maximizing TTS and at the same time we required only minor drop in performance regarding the number of Views and CTR. We argue that if users are willing to spend more time with the service, they are more engaged with it. Apart from that, we also took into consideration the average length of recommended videos because our experiments suggested that the longer the videos were, the smaller the number of video views was. However, we aimed at increasing TTS together with maximizing the total number of views.

The first approach, modelling video watch time directly, did not meet our expectations. It resulted in recommending longer videos while decreasing all other metrics. In the second step, we combined different single models' predictions together using two different parameterized functions (1) and (2) and created α-*compound model* and β-*compound model*. It is important to compare models trained and evaluated on the same time period as the results in Tables 2, 3 and 4 suggest. The takeaway is that eventually, the best α-*compound model* and the best β-*compound model* performed similarly and were able to consistently and substantially boost TTS, compared to the best performing click model.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we demonstrated how to approach a desirable trade-off between CTR and TTS with respect to clicks within a personalized video recommendation system. Two different methods are proposed for modelling the relationship between click probability and watch time on a video. In addition, we proposed algorithms based on generalized linear models suitable for online learning in an industry setting.

Models that were optimized only for watch time did not succeed at increasing TTS. However, combining these models with the model optimizing CTR resulted in the best-balanced results and increased TTS by nearly 28%. Desired objective was achieved as the combined models provide higher user engagement.

Ongoing experiments using the above mentioned combined models also show promising results for an article recommendation task at news and lifestyle content services.

As our future work we are going to experiment with different models and model combinations, use hyper-parameter optimization to achieve the best trade-off between TTS and CTR, and compare our results to different competing models.

REFERENCES

[1] Himan Abdollahpouri, Gediminas Adomavicius, Robin Burke, Ido Guy, Dietmar Jannach, Toshihiro Kamishima, Jan Krasnodebski, and Luiz Pizzato. 2020. Multistakeholder recommendation: Survey and research directions. *User Modeling and User-Adapted Interaction* 30, 1 (Jan 2020), 127–158. https://doi.org/10.1007/s11257-019-09256-1

[2] Eugene Agichtein, Eric Brill, Susan Dumais, and Robert Ragno. 2006. Learning User Interaction Models for Predicting Web Search Result Preferences. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Seattle, Washington, USA) *(SIGIR '06)*. Association for Computing Machinery, New York, NY, USA, 3–10. https://doi.org/10.1145/1148170.1148175

[3] Simon Attfield, Gabriella Kazai, Mounia Lalmas, and Benjamin Piwowarski. 2011. Towards a science of user engagement (Position Paper). (01 2011).

[4] Georg Buscher, Ludger van Elst, and Andreas Dengel. 2009. Segment-Level Display Time as Implicit Feedback: A Comparison to Eye Tracking. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Boston, MA, USA) *(SIGIR '09)*. Association for Computing Machinery, New York, NY, USA, 67–74. https://doi.org/10.1145/1571941.1571955

[5] Bingrui Geng, Lingling Li, Licheng Jiao, Maoguo Gong, Qing Cai, and Yue Wu. 2015. NNIA-RS: A multi-objective optimization based recommender system. *Physica A: Statistical Mechanics and its Applications* 424 (04 2015). https://doi.org/10.1016/j.physa.2015.01.007

[6] Youngho Kim, Ahmed Hassan Awadallah, Ryen W. White, and Imed Zitouni. 2014. Modeling Dwell Time to Predict Click-level Satsifaction. In *The 7th Annual International ACM Conference on Web Search and Data Mining (WSDM 2014)* (the 7th annual international acm conference on web search and data mining

(wsdm 2014) ed.). ACM. https://www.microsoft.com/en-us/research/publication/modeling-dwell-time-to-predict-click-level-satsifaction/

[7] Dmitry Lagun and Mounia Lalmas. 2016. Understanding User Attention and Engagement in Online News Reading. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining* (San Francisco, California, USA) *(WSDM '16)*. Association for Computing Machinery, New York, NY, USA, 113–122. https://doi.org/10.1145/2835776.2835833

[8] 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* (London, United Kingdom) *(KDD '18)*. Association for Computing Machinery, New York, NY, USA, 1930–1939. https://doi.org/10.1145/3219819.3220007

[9] Mario Rodriguez, Christian Posse, and Ethan Zhang. 2012. Multiple Objective Optimization in Recommender Systems. In *Proceedings of the Sixth ACM Conference on Recommender Systems* (Dublin, Ireland) *(RecSys '12)*. Association for Computing Machinery, New York, NY, USA, 11–18. https://doi.org/10.1145/2365952.2365961

[10] Ryen W. White and Diane Kelly. 2006. A Study on the Effects of Personalization and Task Information on Implicit Feedback Performance. In *15th Annual ACM CIKM Conference on Information and Knowledge Management (CIKM 2006), November 5-11, 2006, Arlington, Virginia, USA* (15th annual acm cikm conference on information and knowledge management (cikm 2006), november 5–11, 2006, arlington, virginia, usa ed.). 297–306. https://www.microsoft.com/en-us/research/publication/study-effects-personalization-task-information-implicit-feedback-performance/

[11] Xing Yi, Liangjie Hong, Erheng Zhong, Nanthan Nan Liu, and Suju Rajan. 2014. Beyond Clicks: Dwell Time for Personalization. In *Proceedings of the 8th ACM Conference on Recommender Systems* (Foster City, Silicon Valley, California, USA) *(RecSys '14)*. Association for Computing Machinery, New York, NY, USA, 113–120. https://doi.org/10.1145/2645710.2645724

[12] 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* (Copenhagen, Denmark) *(RecSys '19)*. Association for Computing Machinery, New York, NY, USA, 43–51. https://doi.org/10.1145/3298689.3346997