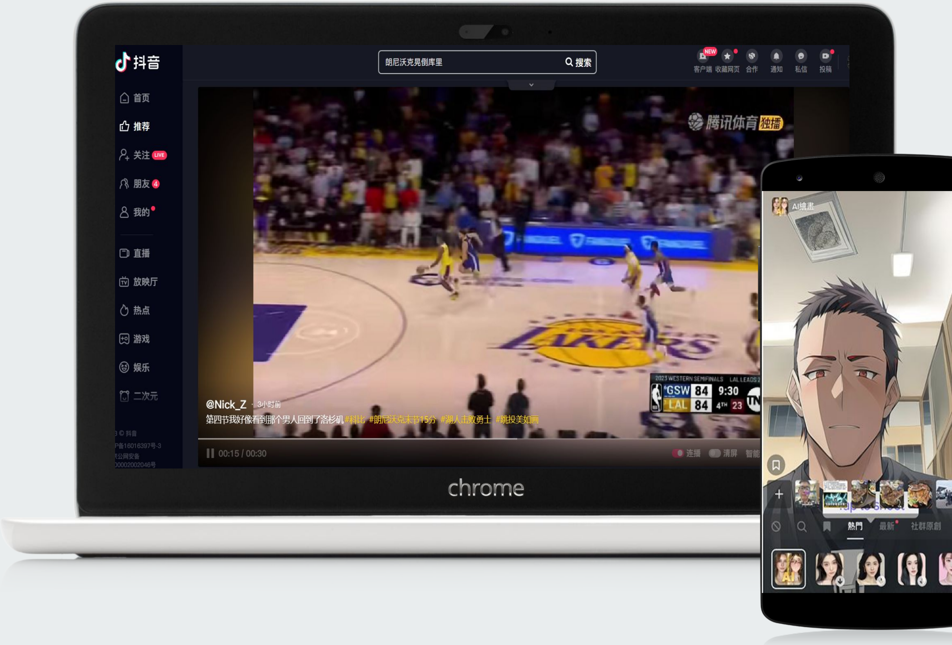
SASRec4SV: Self-Attention Sequential Recommendation for Short Video

Jiacheng Shen
Anran Wang
Krabs Siew

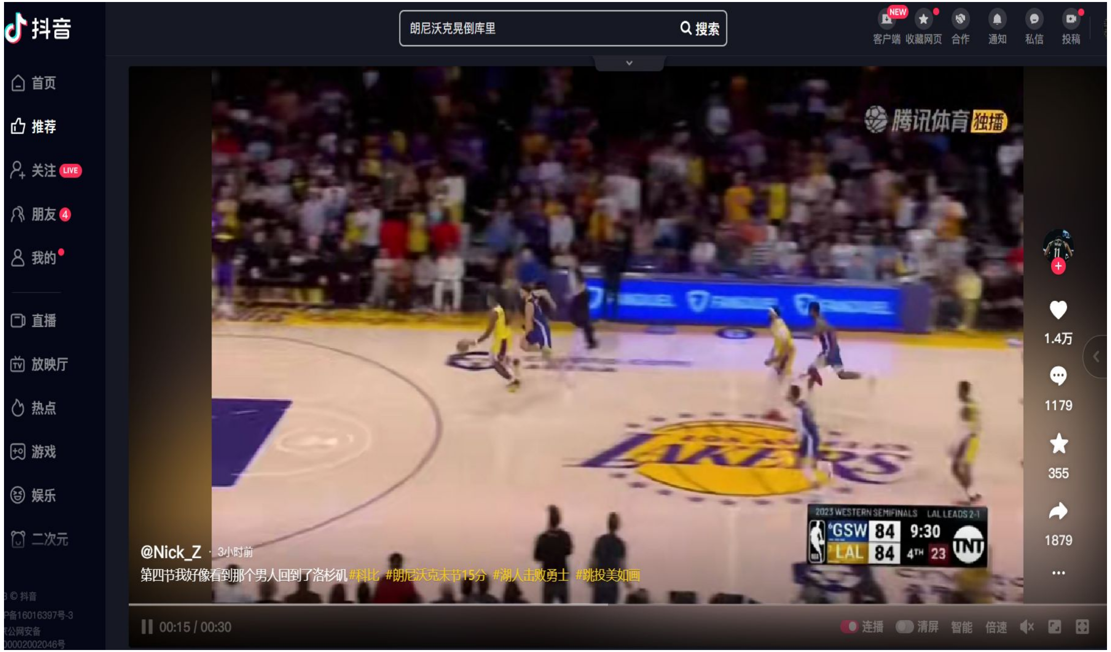


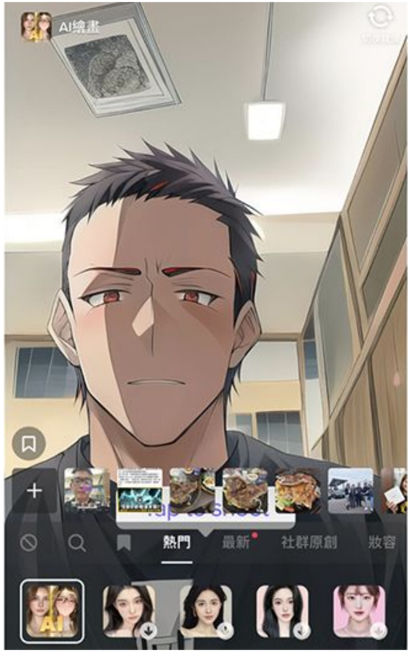
Outline

- Motivation
- Methods
- Experiments
- Results
- Discussions
- Preliminary experimentation on Q-learning

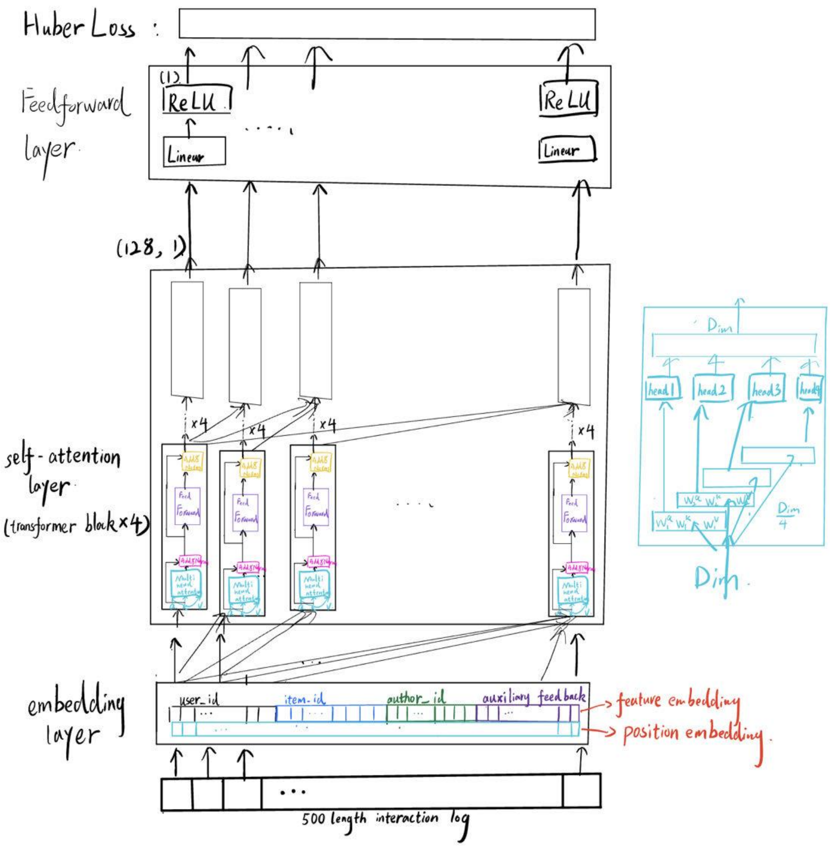
Motivation

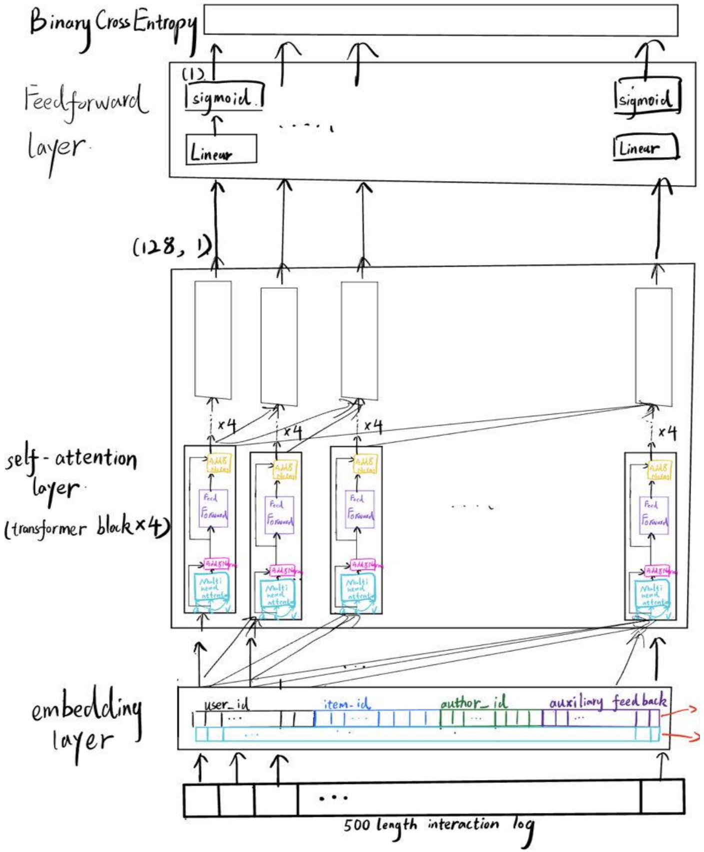
The monthly active users on TikTok have reached one billion worldwide(Tik 2022), with recommendation algorithms playing crucial roles. How do we recommend short videos precisely for users?





Methods: SASRec for sequence based





Hyperparameters:

Epoch:75

Lr: 1e-3

Weight decay: 1e-3

Batch size: 200

Methods: WTG

How do we defined the relevance? - implicit feedback

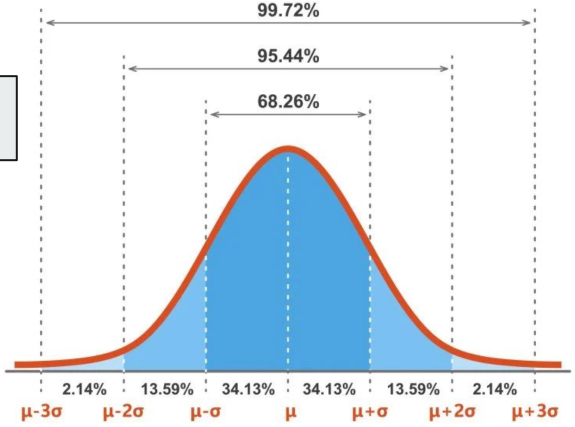
The dataset does not provide an explicit label on whether the user likes the video clip or not. Therefore, we use watch duration as an implicit feedback for the relevance.

Watch Time ⇨ Relevance ⇨ Performance

$$Z = \frac{x - \mu}{\sigma}$$

If z > threshold, then relevant

If z < threshold, then non-relevant



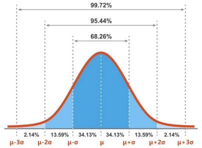
Credit

Methods: from WTG to Relevance

WTG (Watch Time Gain) as an unbiased matrix

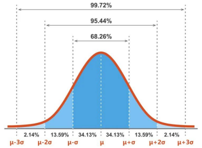
Group 1

0-1 min



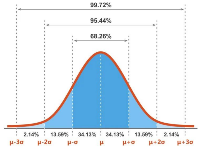
Group 2

1-2 mins



Group 3

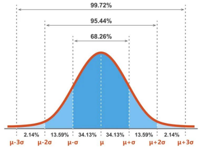
2-3 mins



……

Group m

15-20 mins



$$Z = \frac{x - \mu}{\sigma}$$

If z > threshold, then relevant

If z < threshold, then non-relevant

Methods: Duration Bias Analysis

Does our model has a bias against the duration of a video?

We can apply the idea of nDCG to WTG

Recommended Items

- Item 1
- Item 2
- Item 3

Relevant Items

- Item 1
- Item 2
- Item 3

Group 1

Group 2

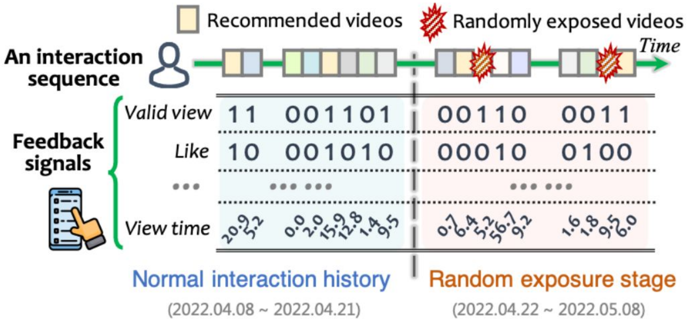
Group 3

......

Group m

Experiments: Dataset

KuaiRand Dataset 1k



	user_id	video_id	date	hourmin	time_ms	play_time_ms	duration_ms	WTRatio	author	auxi_feedback
0	0	4354972	20220409	900	1649467982289	0	70100	0.000000	8075419	0
1	0	1329429	20220409	900	1649467982289	0	51422	0.000000	8151955	0
2	0	346081	20220409	900	1649467982289	0	11696	0.000000	3016953	0
3	0	2058916	20220409	900	1649467982289	0	66433	0.000000	2717033	0
4	0	2528540	20220409	900	1649467982289	5332	11450	0.465677	1330404	1
...	...	...	...	...	...	...	...	...	...	...
4645541	999	411299	20220421	2300	1650552207405	14895	13166	1.131323	8539640	1
4645542	999	694615	20220421	2300	1650552207405	3573	35040	0.101969	8296507	0
4645543	999	1240364	20220421	2300	1650552207405	927	63566	0.014583	4326229	0
4645544	999	2514654	20220421	2300	1650552207405	65048	99100	0.656387	505389	1
4645545	999	2897178	20220421	2300	1650552339920	93384	137480	0.679255	8325519	1

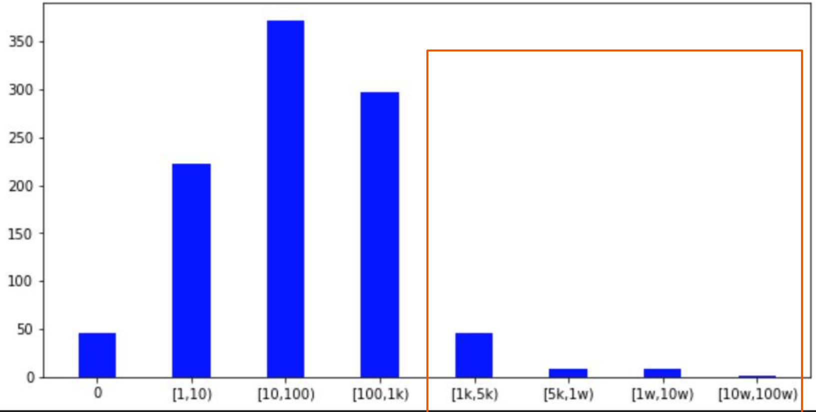
4645546 rows × 10 columns

1. Transform auxiliary feedback into numbers
2. Different from movielens: Consider two options: 1. watch time ratio as ground truth and MSE as loss function. 2. Set binary label: watch time ratio>0.5 as 1 assuming user is satisfied and cross entropy as loss function,similarly watch time ratio<=0.5 as 0.
3. Long interaction sequence(Average 5000 pieces): set max sequence length; Padding or take the newest max sequence length log.
4. Split the dataset: from April 9th to April 20th as training set, April 21st as validation set, and April 22nd as test set.

https://kuairand.com/

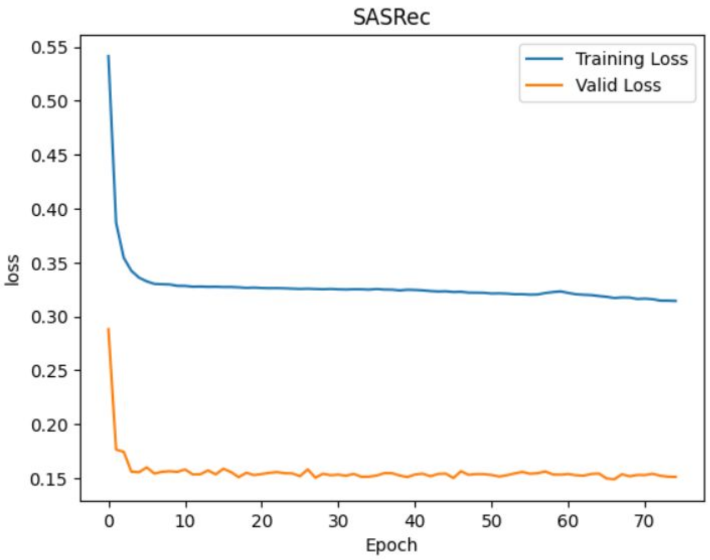
Experiments: Popularity Bias

	A	B	C	D	E	F	G	H	I
1	user_id	user_active_degree	is_lowactive_period	is_live_streamer	is_video_author	follow_user_num	follow_user_num_ra	fans_user_num	fans_user_num_rang
2	0	full_active	0	1	1	514	500+	150	[100,1k)
3	1	full_active	0	-124	1	457	(250,500]	20	[10,100)
4	2	full_active	0	-124	1	8	(0,10]	26	[10,100)
5	3	full_active	0	1	1	91	(50,100]	2166	[1k,5k)
6	4	full_active	0	-124	1	261	(250,500]	10	[10,100)
7	5	full_active	0	-124	1	15	(10,50]	23	[10,100)
8	6	full_active	0	-124	0	75	(50,100]	26	[10,100)
9	7	full_active	0	-124	1	2103	500+	56	[10,100)
10	8	high_active	0	-124	1	265	(250,500]	27	[10,100)
11	9	high_active	0	-124	0	71	(50,100]	0	0
12	10	2_14_day_new	0	-124	1	54	(50,100]	6	[1,10)
13	11	full_active	0	-124	1	22	(10,50]	9	[1,10)
14	12	middle_active	0	-124	1	420	(250,500]	254	[100,1k)
15	13	middle_active	0	1	1	421	(250,500]	743	[100,1k)
16	14	full_active	0	-124	1	217	(150,250]	34	[10,100)
17	15	full_active	0	1	1	451	(250,500]	717	[100,1k)
18	16	full_active	0	-124	1	47	(10,50]	197	[100,1k)
19	17	high_active	0	-124	0	14	(10,50]	0	0
20	18	full_active	0	1	1	740	500+	89	[10,100)
21	19	full_active	0	-124	1	309	(250,500]	802	[100,1k)
22	20	high_active	0	-124	0	5	(0,10]	1	[1,10)
23	21	full_active	0	-124	1	53	(50,100]	51	[10,100)
24	22	full_active	0	-124	1	1533	500+	150	[100,1k)
25	23	full_active	0	1	1	1935	500+	627	[100,1k)
26	24	middle_active	0	-124	1	266	(250,500]	454	[100,1k)

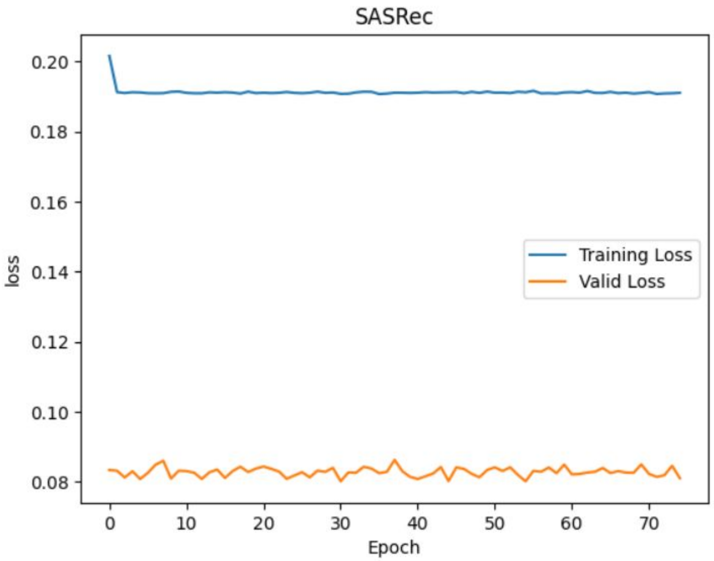


Results: Training the Model

Cross Entropy Loss with binary watch time label



Huber Loss with real watch time ratio as ground truth.



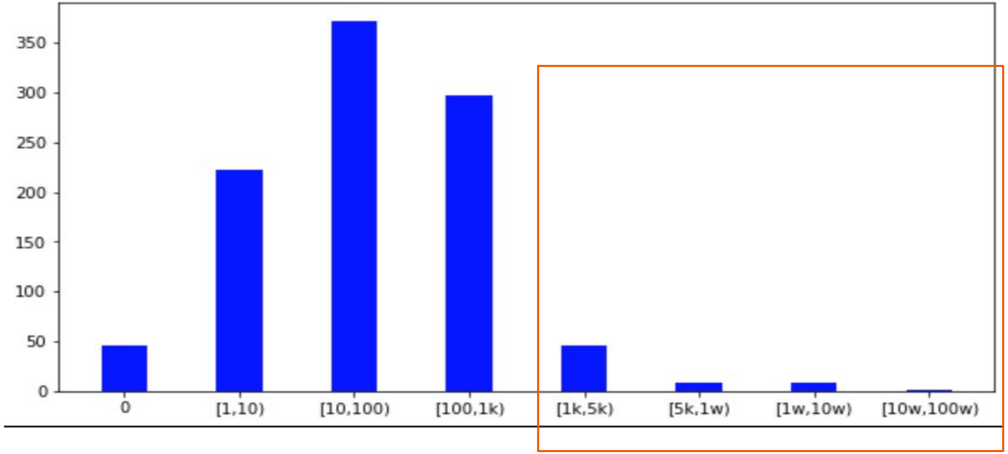
Results: Training the Model

Only evaluate the model itself, not on users

SASRec	F1	Recall	Precision	Accuracy
Binary Setting	0.5765	0.5393	0.6215	0.8597

SASRec	RMSE
Watch time prediction setting	2.9390

Results: Popularity Bias



	Normal users *with fans less than 1000	Influencers *with fans more than 1000
Accuracy	0.9099	0.8323
F1	0.5829	0.5743

Discussions & Contributions

1. We reproduced SASRec(Kang 2018) and adapted it into short video recommendation task.
2. We roughly verified that traditional Q-learning on short video recommendation is extremely difficult to converge(Xin 2020).

Future Directions:

1. Capture more information in auxiliary feedback.

2. Deal with duration bias in short videos from model side(Yu 2022).

3.Hyper parameter tuning.

4. Figure out the reason why validation loss is less than training loss.

Preliminary Experimentation: Q-learning

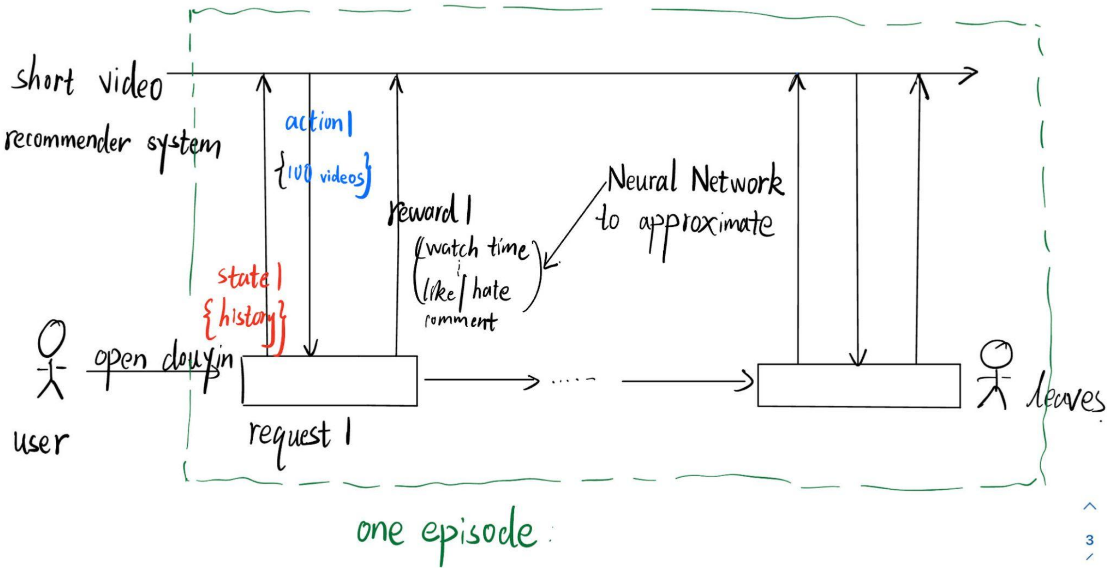
Action space: short videos

Agent: recommender model

Environment: users

Reward: user's watch time

Mapping last 200 recommended videos into a state



We use pfrl and gym to construct the environment and training. However, it occupied 99% of my RAM and lasted 30 minutes without any result.

The reason could be the action space is too sparse(4 million videos) so the agen is hard to find an optimal strategy.

Works Cited

Kang, Wang-Cheng, and Julian McAuley. "Self-attentive sequential recommendation." 2018 IEEE international conference on data mining (ICDM). IEEE, 2018.

Cai, Qingpeng, et al. "Constrained Reinforcement Learning for Short Video Recommendation." arXiv preprint arXiv:2205.13248 (2022).

Zheng, Yu, et al. "DVR: Micro-Video Recommendation Optimizing Watch-Time-Gain under Duration Bias." Proceedings of the 30th ACM International Conference on Multimedia. 2022.

Gao, Chongming, et al. "KuaiRand: An Unbiased Sequential Recommendation Dataset with Randomly Exposed Videos." Proceedings of the 31st ACM International Conference on Information & Knowledge Management. 2022.

Kaye, D. Bondy Valdovinos, Jing Zeng, and Patrik Wikstrom. TikTok: Creativity and culture in short video. John Wiley & Sons, 2022.