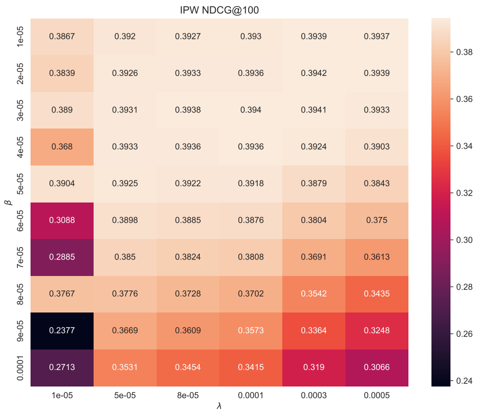

Figure 9: an Example of Hyper-parameter Search for IPW on NFLX Prize

	Movie Lens	Netflix Prize	Million Song
LR	2.0	1.0	40.0
Adversarial LR	1.0	1.0	1.0
λ	8e-6	5e-5	1e-5
Advantage Score	As in Eq.(7)	As in Eq.(7)	As in Eq.(7)
τ	1.5	1.0	0.5
Adversarial Model	2 Layer MLP	2 Layer MLP	2 Layer MLP

Table 6: Parameters used in our evaluation. Here LR represents learning rate.

A EXPERIMENT DETAILS

We set training epochs $T = 50$, $m = 0.9$ in algorithm 1. We use a batch size of 1024 for MovieLens and 8192 for Million Song and Netflix Prize. We set the momentum of SGD optimizer as 0.9. We use 10k users for validation and another 10k users for testing on Movie Lens. We increase this number 10k to 40k on Netflix Prize and 50k on Million Song dataset respectively. Some datasets provide explicit preference information like Movie Lens. We follow the literature [27] and convert them into implicit settings through a preference threshold. Preference score greater than this threshold is set to 1 and otherwise it is set to 0.

We conduct the experiments on machines with 64G memory and 64 CPUs without GPU. This choice is due to budget concerns. We can finish the training on CPU as our model is relative small and reasonably fast enough on CPU. It typically takes 8 hours to train POSIT on MovieLens, 1.5 days on Netflix Prize and 1 day on Million Song. The major overhead is due to gradient optimization on these very large datasets. Speed can be potentially improved vastsly with GPU based training.

We conduct hyper-parameter search. Figures 8 and 9 are two examples of the parameter search for baseline IPW on MovieLens and NFLX Prize. Note that the best parameter can be found around the center of the figure. And we can empirically observe changing hyper-parameter in any other directions will simply bring down the performance. Note that we may fine-tune multiple factors and the results won't necessarily be presented as a 2-dimensional figure. For example, for baseline IPW, we also fine-tune the learning rate. Therefore, we empirically conduct hyper-parameter searching for multiple times to better pinpoint an suitable range on different hyper-parameter combinations.

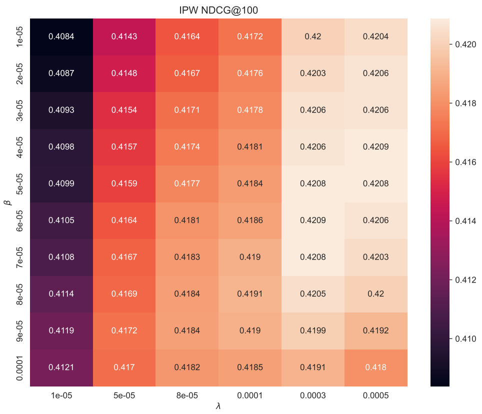

Figure 8: an Example of Hyper-parameter Search for IPW on MovieLens

Table 6 lists all the hyper-parameters that is needed to exactly reproduce our results. We will also release our codes upon acceptance.