On Sampling Collaborative Filtering Datasets

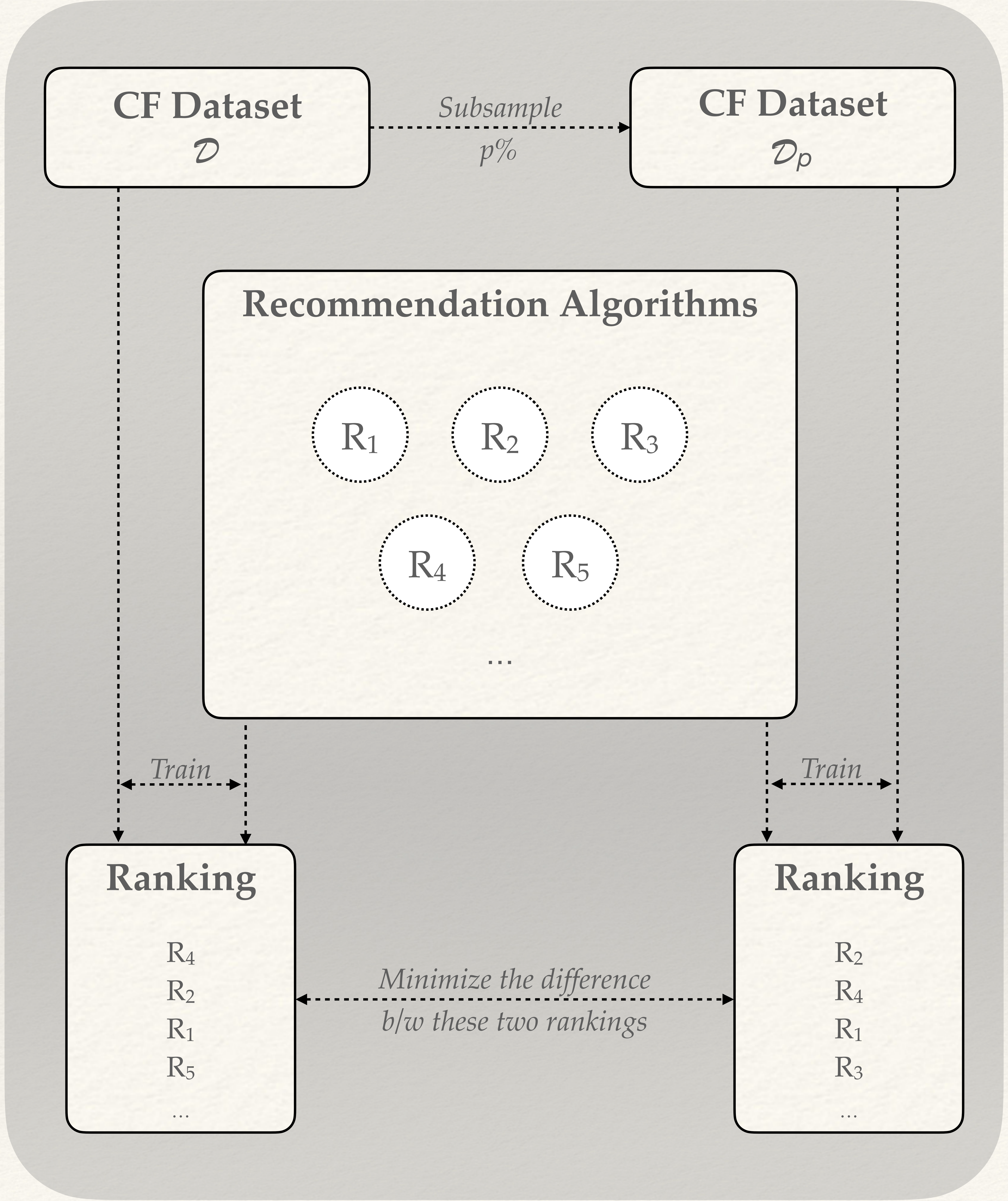
Noveen Sachdeva ; Carole-Jean Wu ; Julian McAuley

UC San Diego FACEBOOK AI

Research Goal

Generate a sample of a collaborative filtering (CF) dataset which can accurately retain the **relative ordering** of different recommendation algorithms

- Minimize the data subset size, such that difference between the two rankings is minimal
- Directly correlates with the confidence in the results of **any** paper comparing different recommendation models trained on sub-sampled data (vast majority)



SVP-CF

Selection-via-proxy for collaborative filtering data

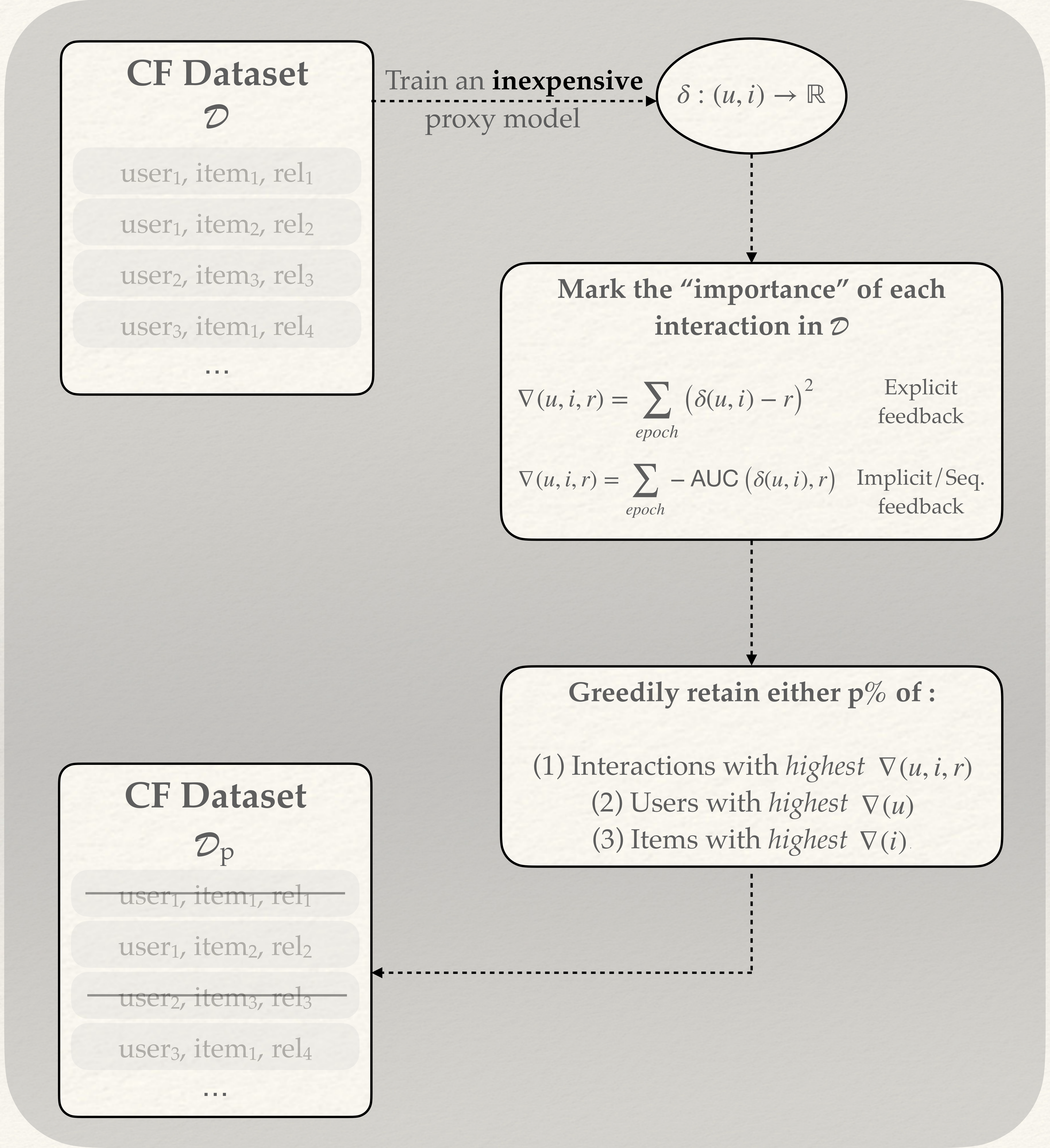
Premise: **Easy** parts of a dataset are most likely **easy** for all recommendation algorithms. Hence, removing such data is unlikely to change the relative ordering of algorithms.

SVP- CF

Selection-via-proxy for collaborative filtering data

Robust framework:

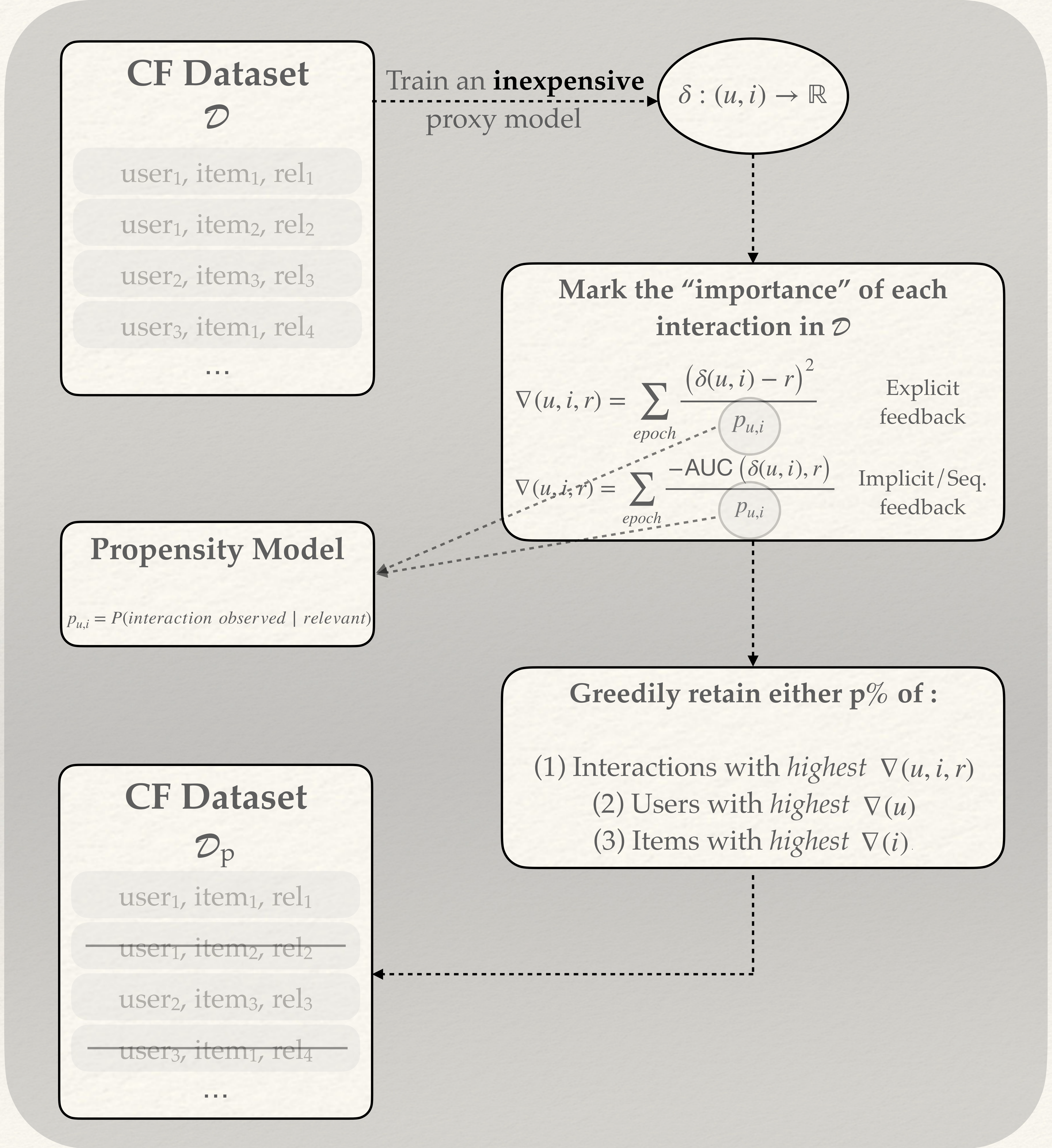
- Uses a proxy model to tag the **importance** of each interaction
- Efficiently handle multiple recommendation scenarios *e.g.* explicit, implicit, sequential, etc.
- Sample across varieties of data modalities: interactions, users, items, or even combinations of them



SVP- CF- Prop

Handling the missing-not-at-random characteristics

- Re-weigh the importance scores in SVP-CF using the probability of a user-item interaction going missing (propensity).
- Implicitly also handles the long-tail and data sparsity issues in user-item interaction data.



Data-Genie

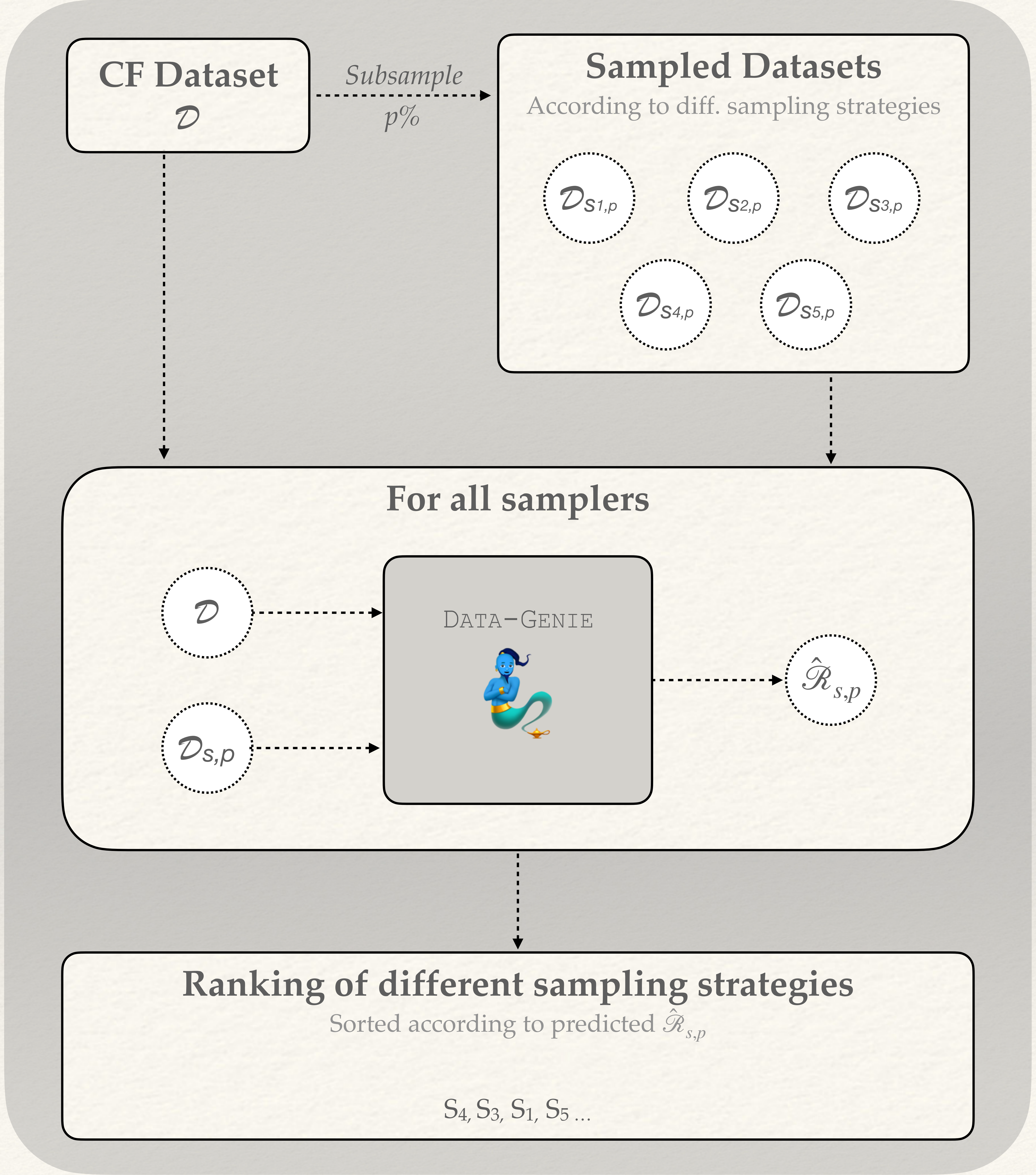
Which sampler is best for me?

Premise: Can we build an oracle-model which given (1) a dataset, (2) list of sampling strategies, and (3) a sampling budget, can **automatically predict** which sampling scheme would be the best?

DATA-GENIE

Which Sampler is best for me?

- Dynamically predicts the **performance** of a sampling strategy for any given CF-dataset.
- A trained DATA-GENIE model can transfer to **any** dataset, and can predict the utility of **any** sampling strategy.



DATA-GENIE

How is it trained?

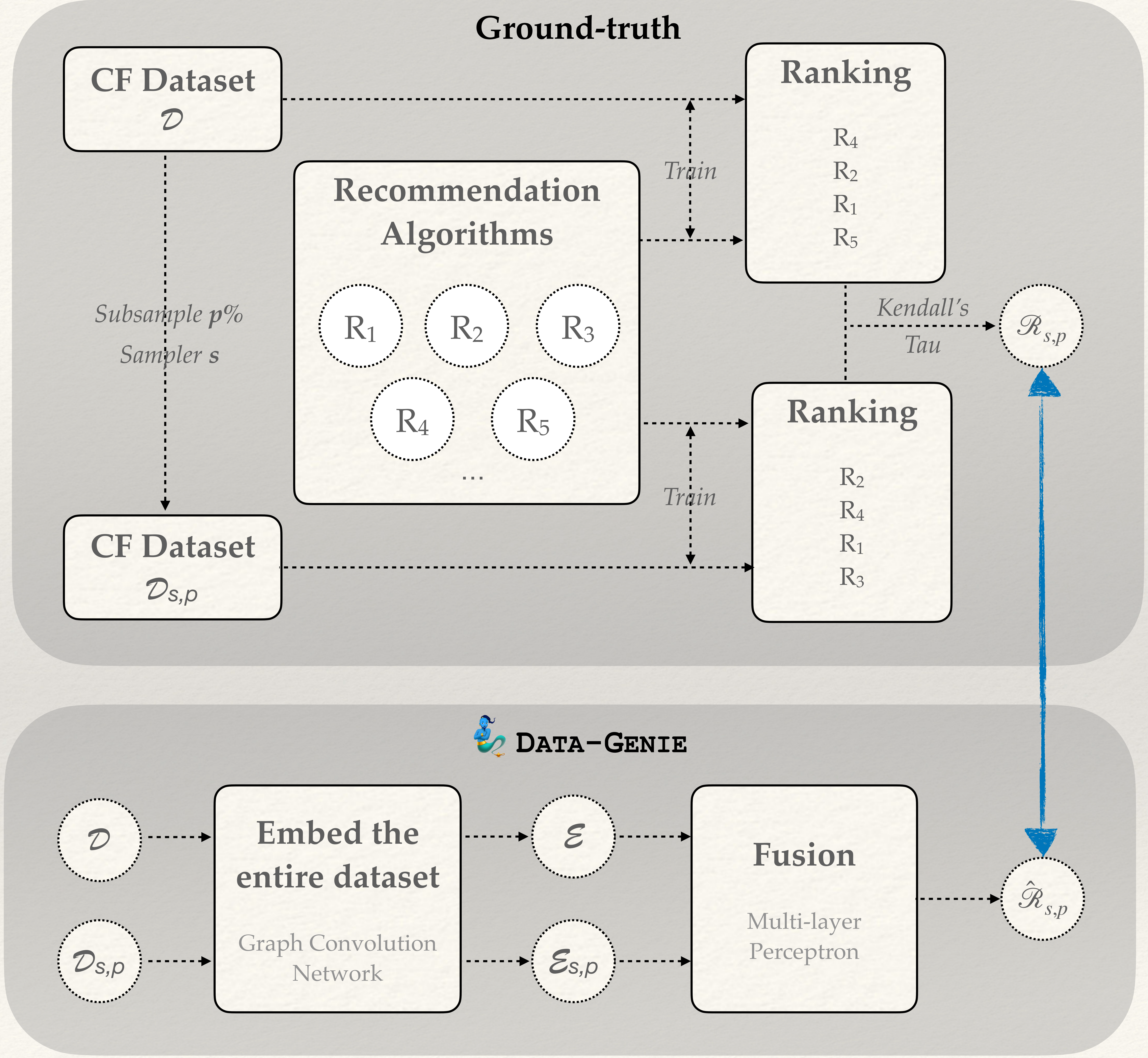
- Circumvents the time-consuming process of training and benchmarking various algorithms.

- DATA-GENIE-regression:

$$\arg\min \sum_{\mathcal{D},\, s,\, p} \left(\mathcal{R}_{s,p} - \hat{\mathcal{R}}_{s,p}\right)^2$$

- DATA-GENIE-ranking:

$$\arg\min \sum_{\mathcal{D},\, p} \sum_{\mathcal{R}_{s_i,p} > \mathcal{R}_{s_j,p}} - \ln \sigma\left(\hat{\mathcal{R}}_{s_i,p} - \hat{\mathcal{R}}_{s_j,p}\right)$$



Experiments

Setup

	Sampling strategy
Interaction sampling	Random
	Stratified
	Temporal
	SVP-CF *w/* MF
	SVP-CF *w/* Bias-only
	SVP-CF-Prop *w/* MF
	SVP-CF-Prop *w/* Bias-only
User sampling	Random
	Head
	SVP-CF *w/* MF
	SVP-CF *w/* Bias-only
	SVP-CF-Prop *w/* MF
	SVP-CF-Prop *w/* Bias-only
Graph	Centrality
	Random-walk
	Forest-fire

Table 1: Sampling strategies used in our experiments

- 16 different sampling strategies
- 6 collaborative filtering datasets
- Explicit / Implicit / Sequential feedback for each CF-dataset
- 7 recommendation algorithms in our benchmarking suite
- A total of ***400k*** recommendation models trained! (~9 months of compute time!)

Experiments

Major Results

	Sampling strategy	*Average* Kendall's Tau
Interaction sampling	Random	0.407
	Stratified	0.343
	Temporal	0.405
	SVP-CF *w/* MF	0.484
	SVP-CF *w/* Bias-only	0.468
	SVP-CF-Prop *w/* MF	0.43
	SVP-CF-Prop *w/* Bias-only	0.458
User sampling	Random	0.431
	Head	0.19
	SVP-CF *w/* MF	0.344
	SVP-CF *w/* Bias-only	0.343
	SVP-CF-Prop *w/* MF	0.429
	SVP-CF-Prop *w/* Bias-only	0.445
Graph	Centrality	0.266
	Random-walk	0.396
	Forest-fire	0.382

Table 1: Average Kendall's Tau of various sampling strategies

- Widely used practice of making dense data subsets (e.g. Head-user, centrality) seem to be the worst ideas of all sampling strategies.
- SVP-CF significantly outperforms other samplers in retaining the ranking of different recommendation algorithms.

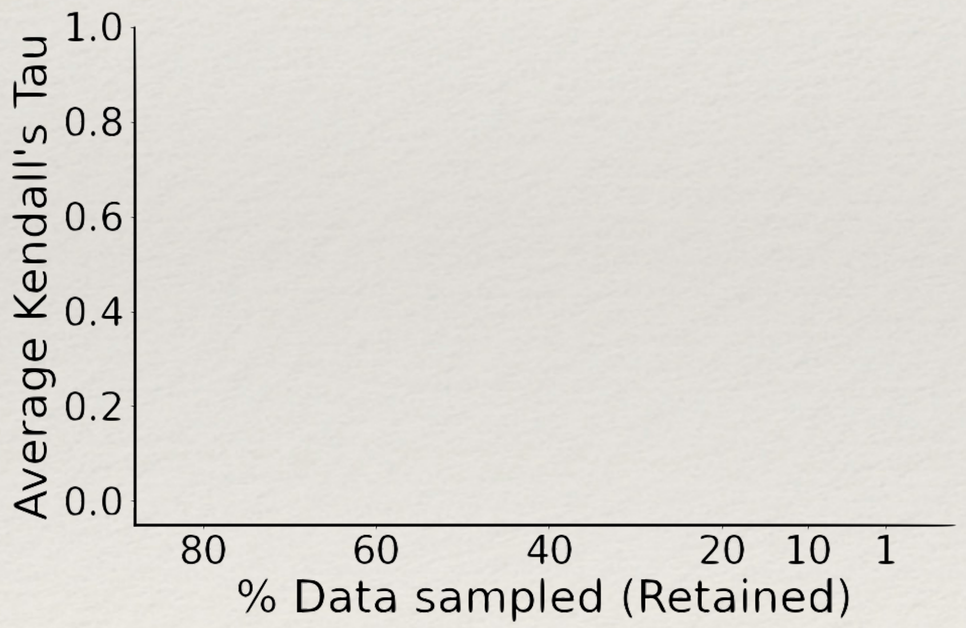


Figure 2: Does Data-Genie improve sampling performance with extreme sampling?

Experiments

Major Results

Sampling strategy		*Average* Kendall's Tau
Interaction sampling	Random	0.407
	Stratified	0.343
	Temporal	0.405
	SVP-CF *w/* MF	0.484
	SVP-CF *w/* Bias-only	0.468
	SVP-CF-Prop *w/* MF	0.43
	SVP-CF-Prop *w/* Bias-only	0.458
User sampling	Random	0.431
	Head	0.19
	SVP-CF *w/* MF	0.344
	SVP-CF *w/* Bias-only	0.343
	SVP-CF-Prop *w/* MF	0.429
	SVP-CF-Prop *w/* Bias-only	0.445
Graph	Centrality	0.266
	Random-walk	0.396
	Forest-fire	0.382

Table 1: Average Kendall's Tau of various sampling strategies

- Widely used practice of making dense data subsets (e.g. Head-user, centrality) seem to be the worst ideas of all sampling strategies.
- SVP-CF significantly outperforms other samplers in retaining the ranking of different recommendation algorithms.

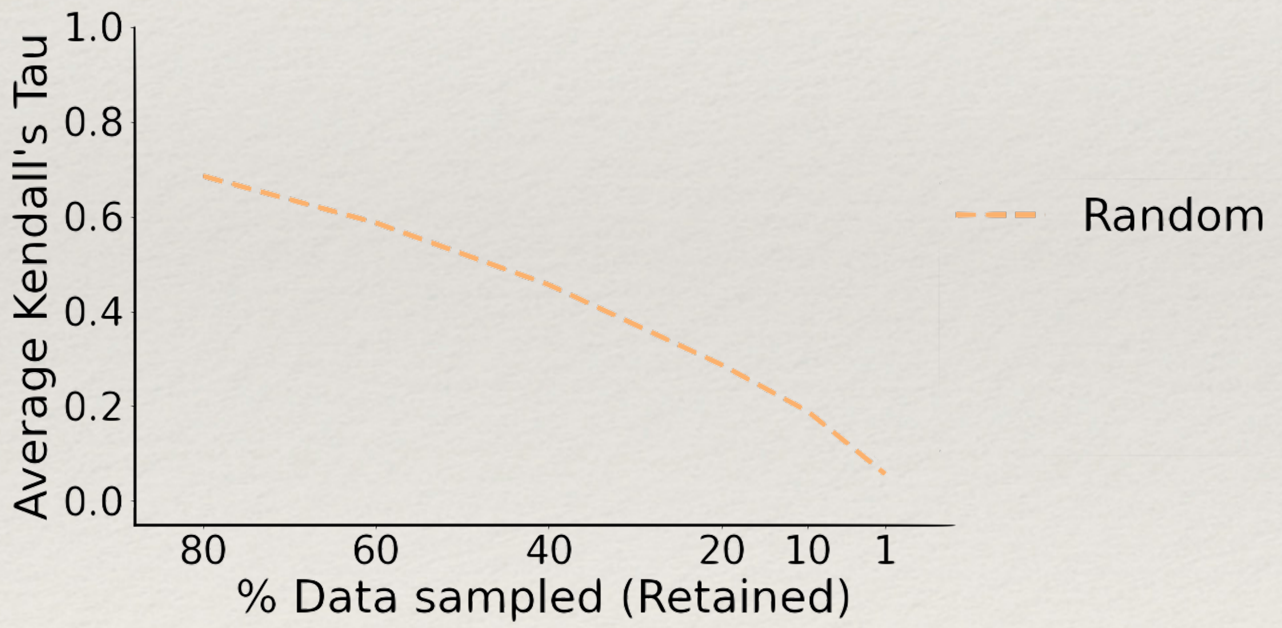


Figure 2: Does Data-Genie improve sampling performance with extreme sampling?

Experiments

Major Results

	Sampling strategy	*Average* Kendall's Tau
Interaction sampling	Random	0.407
	Stratified	0.343
	Temporal	0.405
	SVP-CF *w/* MF	<u>0.484</u>
	SVP-CF *w/* Bias-only	0.468
	SVP-CF-Prop *w/* MF	0.43
	SVP-CF-Prop *w/* Bias-only	0.458
User sampling	Random	0.431
	Head	0.19
	SVP-CF *w/* MF	0.344
	SVP-CF *w/* Bias-only	0.343
	SVP-CF-Prop *w/* MF	0.429
	SVP-CF-Prop *w/* Bias-only	0.445
Graph	Centrality	0.266
	Random-walk	0.396
	Forest-fire	0.382

Table 1: Average Kendall's Tau of various sampling strategies

- Widely used practice of making dense data subsets (e.g. Head-user, centrality) seem to be the worst ideas of all sampling strategies.

- SVP-CF significantly outperforms other samplers in retaining the ranking of different recommendation algorithms.

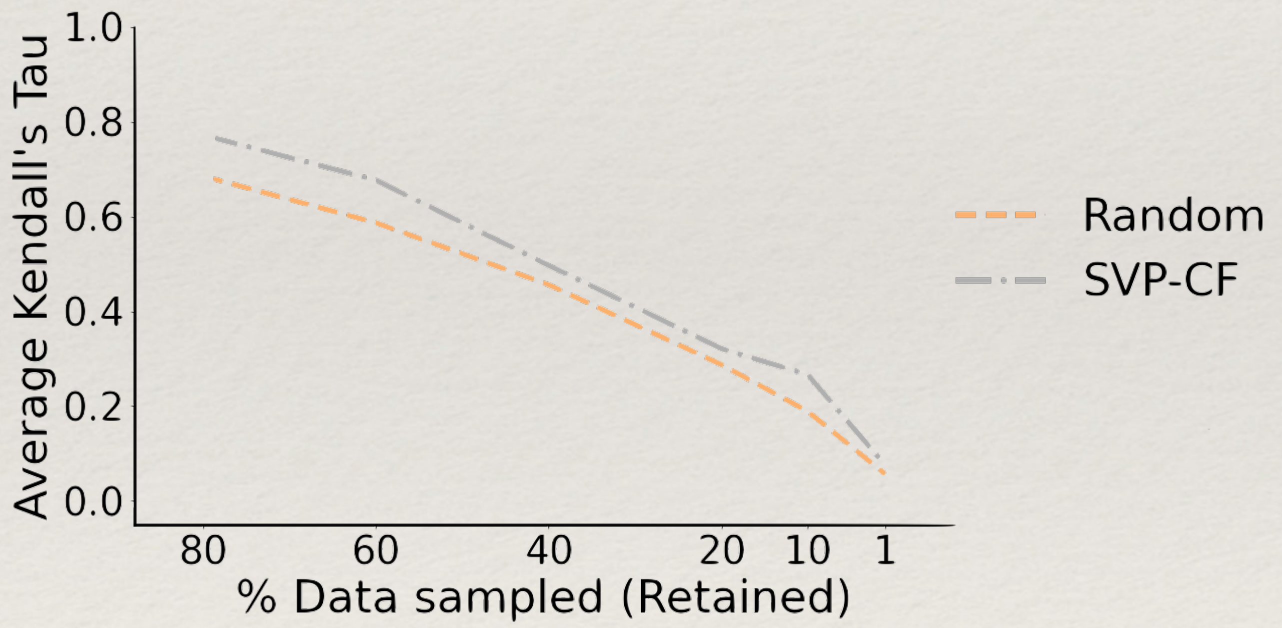


Figure 2: Does Data-Genie improve sampling performance with extreme sampling?

Experiments

Major Results

	Sampling strategy	*Average* Kendall's Tau
Interaction sampling	Random	0.407
	Stratified	0.343
	Temporal	0.405
	SVP-CF *w/* MF	<u>0.484</u>
	SVP-CF *w/* Bias-only	0.468
	SVP-CF-Prop *w/* MF	0.43
	SVP-CF-Prop *w/* Bias-only	0.458
User sampling	Random	0.431
	Head	0.19
	SVP-CF *w/* MF	0.344
	SVP-CF *w/* Bias-only	0.343
	SVP-CF-Prop *w/* MF	0.429
	SVP-CF-Prop *w/* Bias-only	0.445
Graph	Centrality	0.266
	Random-walk	0.396
	Forest-fire	0.382

Table 1: Average Kendall's Tau of various sampling strategies

- Widely used practice of making dense data subsets (e.g. Head-user, centrality) seem to be the worst ideas of all sampling strategies.
- SVP-CF significantly outperforms other samplers in retaining the ranking of different recommendation algorithms.

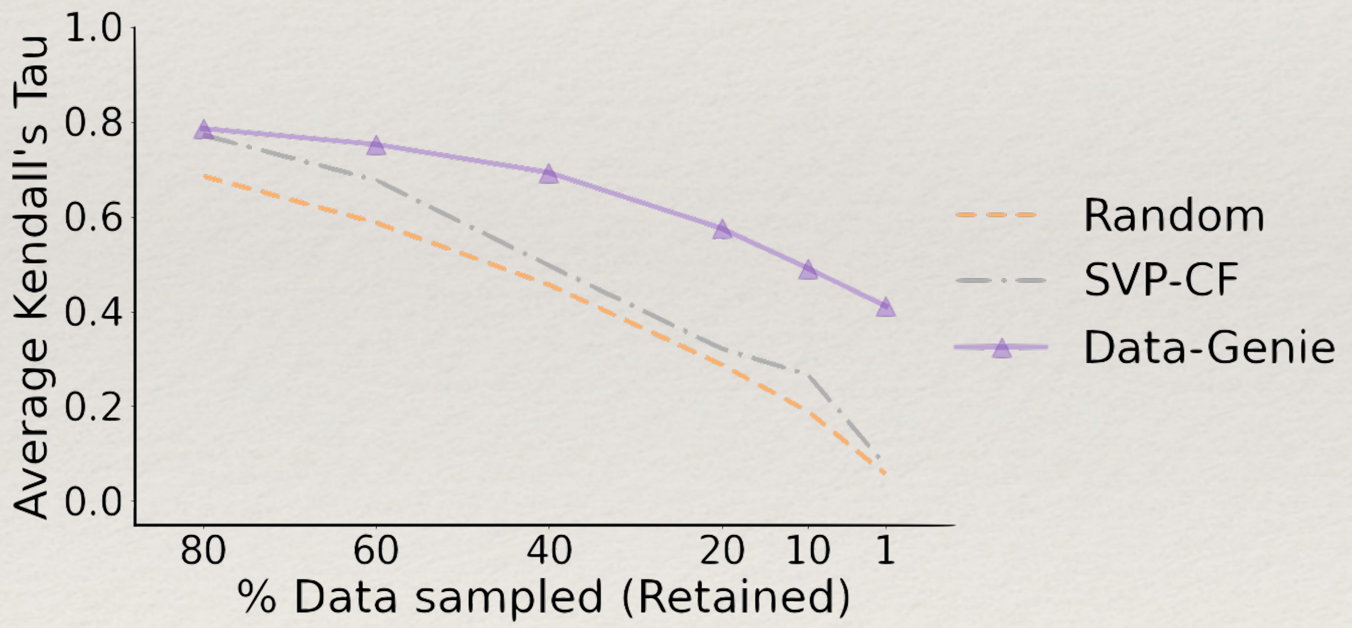


Figure 2: Does Data-Genie improve sampling performance with extreme sampling?

Experiments

Major Results

	Sampling strategy	*Average* Kendall's Tau
Interaction sampling	Random	0.407
	Stratified	0.343
	Temporal	0.405
	SVP-CF *w/* MF	0.484
	SVP-CF *w/* Bias-only	0.468
	SVP-CF-Prop *w/* MF	0.43
	SVP-CF-Prop *w/* Bias-only	0.458
User sampling	Random	0.431
	Head	0.19
	SVP-CF *w/* MF	0.344
	SVP-CF *w/* Bias-only	0.343
	SVP-CF-Prop *w/* MF	0.429
	SVP-CF-Prop *w/* Bias-only	0.445
Graph	Centrality	0.266
	Random-walk	0.396
	Forest-fire	0.382

Table 1: Average Kendall's Tau of various sampling strategies

- Widely used practice of making dense data subsets (e.g. Head-user, centrality) seem to be the worst ideas of all sampling strategies.
- SVP-CF significantly outperforms other samplers in retaining the ranking of different recommendation algorithms.

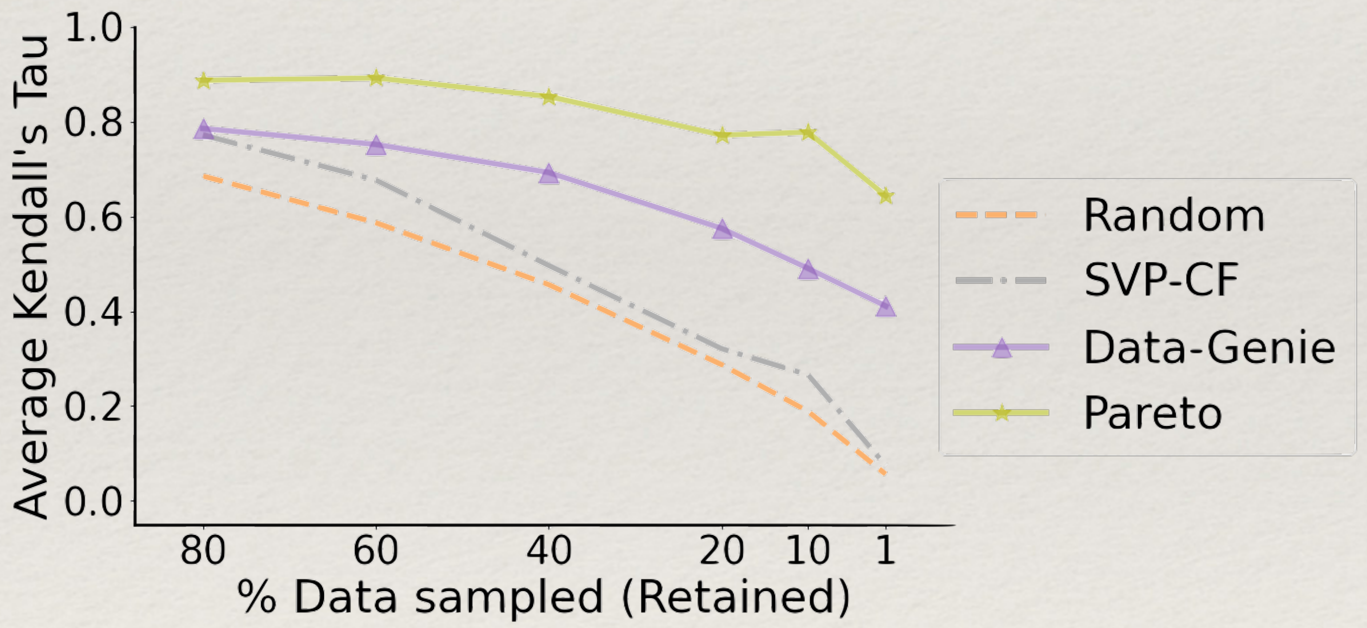


Figure 2: Does Data-Genie improve sampling performance with extreme sampling?

- Using SVP-CF, we can efficiently gauge the ranking of different algorithms with adequate confidence on **40-50%** data sub-samples, leading in an **~2x** time speedup.
- Data-Genie enjoys the same level of performance with only **10%** of the original data, equating to **~5.8x** time speedup!

Environmental Consequences

Consumption	CO_2e (lbs.)
1 person, NY↔SF flight	2k
Human life, 1 year avg.	11k
Weekly RecSys development cycle	20k
” w/ DATA-GENIE	3.4k

Table 1: CO_2 emissions comparison

Given an average weekly RecSys development cycle consisting of:

- Training / testing various recommendation algorithms
- On a medium-sized industrial dataset
- Over a modest GPU setup

We compare the downstream CO_2 emissions of a brute-force search *vs.* DATA-GENIE

Future Directions

- Relative ordering of recommendation algorithms is just a start — encourage the community to think more about general coresets in the context of recommendation.
- Analyzing the fairness aspects of training recommendation algorithms on data subsets.
- Transfer to other domains — classification, clustering, graphs, etc.

Thanks!

@noveens97

For paper, code, and these slides:

SCAN ME