User Simulation for Information Retrieval Evaluation: Opportunities and Challenges

ChengXiang ("Cheng") Zhai
Department of Computer Science
(Carl R. Woese Institute for Genomic Biology
School of Information Sciences
Department of Statistics)
University of Illinois at Urbana-Champaign

czhai@illinois.edu http://czhai.cs.illinois.edu/

Two Different Reasons for Evaluation

- Reason 1: Assess the actual utility of a system
  - Needed for decision making (e.g., is my system good enough to be deployed?)
  - Need to measure how useful a system is to a real user when the user uses the system to perform a real task
  - Absolute performance is needed
- Reason 2: Assess the relative strengths/weaknesses of different systems and methods
  - Needed for research (e.g., is a new system proposed better than all existing ones?)
  - Need to be reproducible and reusable
  - Relative performance is sufficient

Current Practices

- Reason 1: Assess the actual utility of a system
  - A/B Test
  - Small-scale user studies (interactive IR evaluation) [Kelly 09, Harman 11]

Non-Reproducible, Non-Reusable, Non-Generalizable, Expensive, …

- Reason 2: Assess the relative strengths/weaknesses of different systems and methods
  - Test Set approach (Cranfield evaluation paradigm) [Voorhees & Harman 05, Sanderson 10, Harman 11]

Inaccurate representation of real users, Limited aspects of utility, Cannot evaluate an interactive system, …

How can we make a fair comparison of multiple IIR systems using reproducible experiments?

Must control the users ➔ Using user simulators!

A Simulation Model of an IR System, 1973

User simulation in IR has already attracted much attention recently with multiple workshops held in the past (see, e.g., [Azzopardi et al. 10, Clarke et al. 13])

Feasibility of simulation-based evaluation has been shown in some recent work (e.g., [Carterette et al. 15], [Zhang et al. 17], [Pääkkönen et al. 17])

Exciting new work (e.g., [Azzopardi et al. ICTIR 21], Best Paper Award)

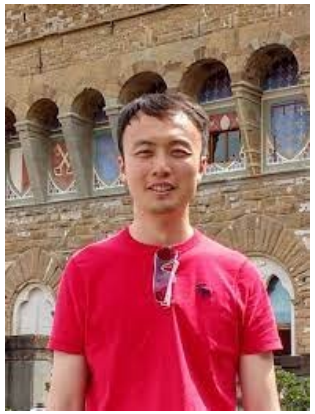
A General Simulation-based Evaluation Methodology

- A collection of user simulators are constructed to approximate real users
- A collection of task simulators are constructed to approximate real tasks
- Both user simulators and task simulators can be parameterized to enable modeling of variation in users and tasks
- Evaluation of a system
  - Have a simulated user perform a simulated task by using (interacting with) the system
  - Compute various measures based on the entire interaction history of the whole “task session”

Rest of the talk

- Some of Our Recent Work
  - A general formal framework for simulation-based evaluation [Zhang et al. ICTIR 17]
  - A cognitive state user model for E-com search [Labhishetty et al. SDM Workshop 20]
  - A tester-based approach to evaluation of user simulators for comparing IIR [Labhishetty & Zhai SIGIR'21]
- Summary & Future Directions

Yinan Zhang (UIUC, Facebook)



Sahiti Labhishetty (UIUC)

1. Yinan Zhang, Xueqing Liu, and ChengXiang Zhai. "Information retrieval evaluation as search simulation: A general formal framework for ir evaluation." In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*, pp. 193-200. 2017
2. Sahiti Labhishetty, Chenxiang Zhai, Suhas Ranganath,Pradeep Ranganathan, A Cognitive User Model for E-Commerce Search, SIAM Data Mining 2020 Workshop on Data Science for Retail and E-Commerce. https://sdm-dsre.github.io/pdf/cognitive_user_model.pdf
3. Sahiti Labhishetty and Chengxiang Zhai. "An Exploration of Tester-based Evaluation of User Simulators for Comparing Interactive Retrieval Systems." SIGIR 2021, to appear.

Search simulation framework [Zhang et al. 17]

- Top level components
  - System: S
  - User / simulator: U
  - Task: T
  - Interaction sequence: I
- Metrics
  - Interaction reward and cost: R(I,T,U,S) and C(I,T,U,S)
  - Simulator reward and cost: R(T,U,S) and C(T,U,S)
    - Expectation w.r.t. p(I|T,U,S)

A lap-level decomposition

- Lap level components
  - Lap, user action and interface card
  - User state and user action model
  - Interaction sequence (refined)
    - A series of (user state, user action, interface card) tuples.
- Metrics
  - *Cumulative* reward and cost: $R^t(I,T,U,S)$ and $C^t(I,T,U,S)$
  - Assume to be the sum of lap-level action reward and cost

Cumulative reward and cost

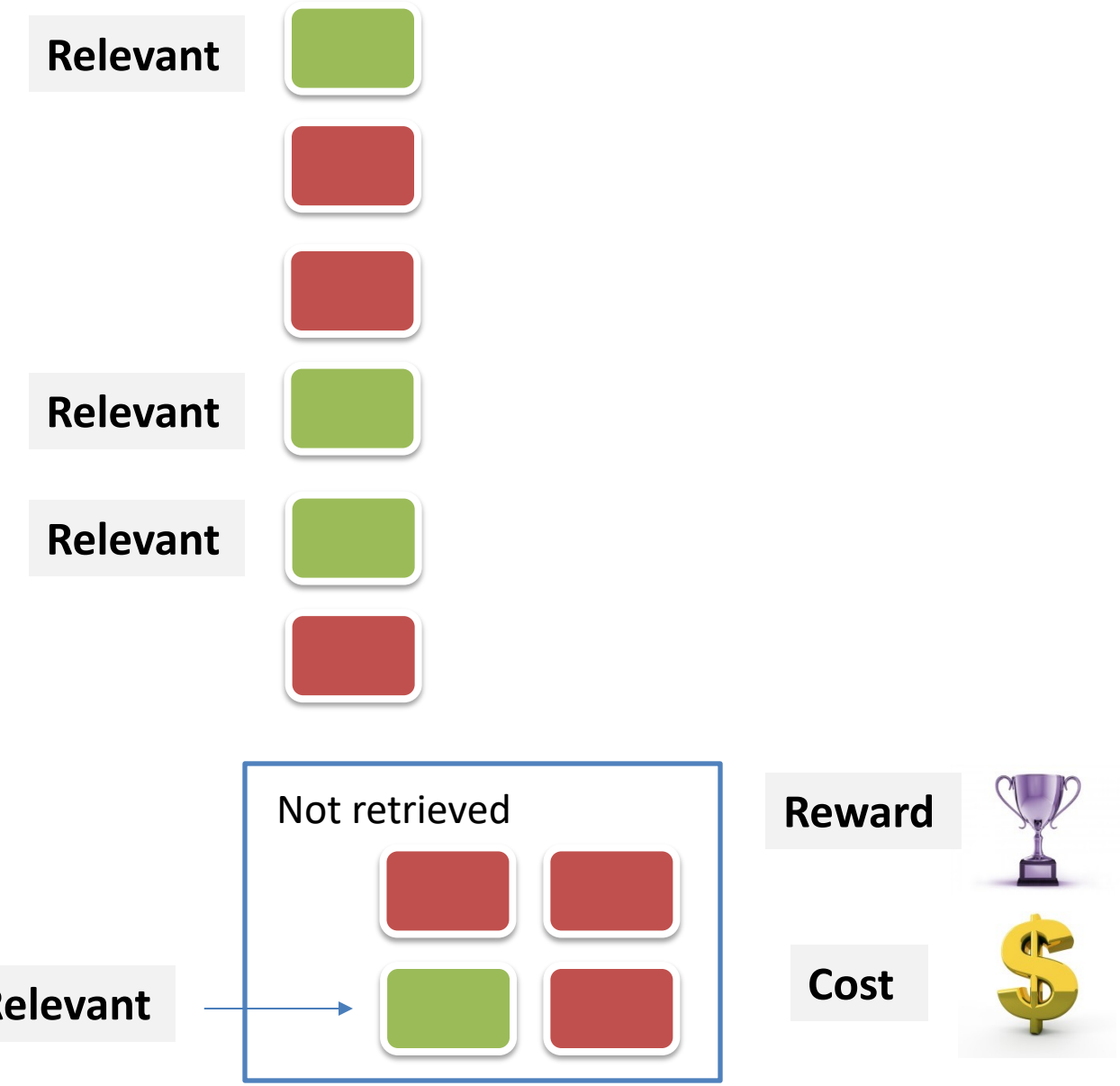
$$R^t(I,T,U,S) = \sum_{i=1}^{t} r(a^i|z^i, q^{i-1})$$

$$C^t(I,T,U,S) = \sum_{i=1}^{t} c(a^i|z^i, q^{i-1})$$

- Remark
  - How to combine the reward and cost measures is application specific
  - The distributions of reward and cost across all interaction sequences are also meaningful

Classical IR simulator

- Task: find (all) relevant documents
- Interface card: document (snippet)
- User action: click / skip (and read next) / stop
  - User always clicks a relevant document
  - User may skip or stop at a non-relevant document
- Lap reward: 1 / 0 for relevant / non-relevant doc
  - Cumulative reward: # relevant docs
- Lap cost: 1 for each doc
  - Cumulative cost: # docs (the simulator scanned through)
- User state: cumulative reward and cost


Relevant
Relevant
Relevant
Not retrieved
Relevant
Reward
Cost

Relevant

Relevant

Relevant

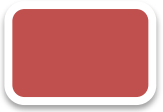
Not retrieved

Relevant

Reward

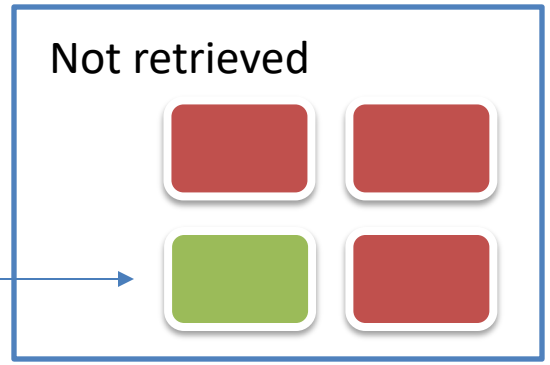
Cost

Relevant



Relevant

Relevant

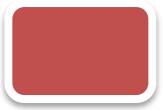


Relevant

Reward

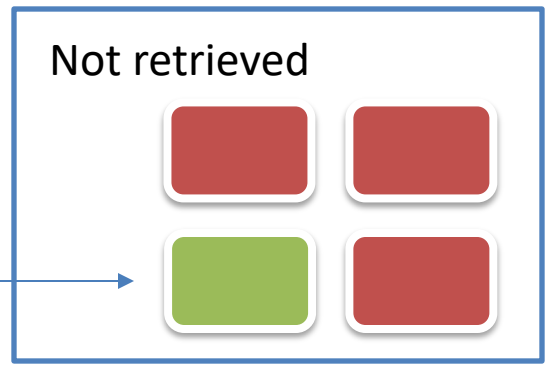
Cost

Relevant



Relevant

Relevant



Relevant

Reward

Cost

Relevant

Relevant

Relevant

Not retrieved

Relevant

Reward

Cost

Relevant

Relevant

Relevant

Not retrieved

Relevant

Reward

Cost

Relevant

Relevant

Relevant

Not retrieved

Relevant

Reward

Cost

Relevant

Relevant

Relevant

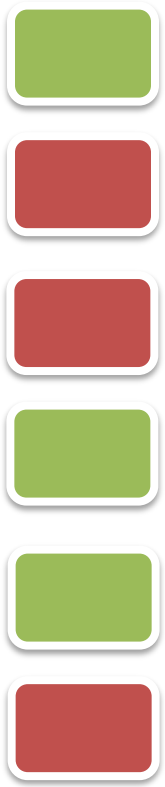
Not retrieved

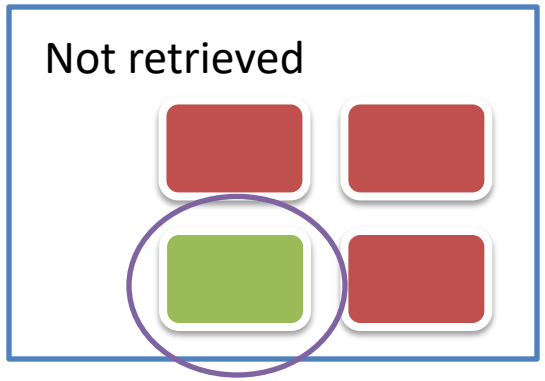
Relevant

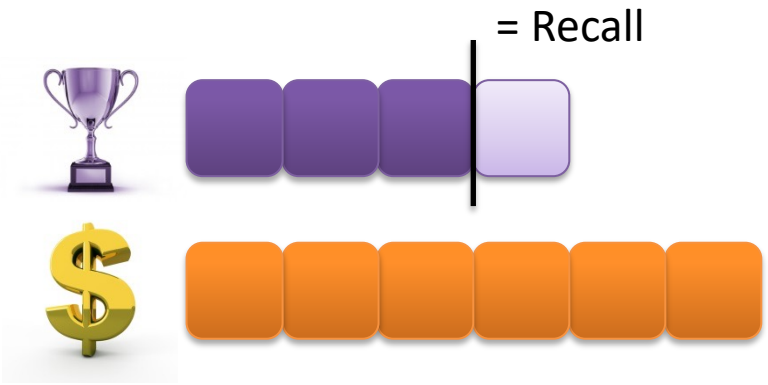
Reward

Precision =

Cost




Not retrieved


= Recall

Metrics in Cranfield test

- Precision
  - R(I,T,U,S) / C(I,T,U,S)
- Recall
  - R(I,T,U,S) / N, N = maximal possible reward
- Remark
  - Assumes user stops when the list is exhausted
  - Precision@K and Recall@K: K = cost budget
  - Precision emphasizes more on cost; Recall emphasizes more on task completion

Mean Average Precision (MAP)

- Variable-recall simulator
  - Classical IR simulator with task of finding N' relevant documents (N' between 1 and N)
  - Stops and only stops when the task is finished
- Average Precision (AP)
  - Average R(I,T,U,S) / C(I,T,U,S) across N variable-recall simulators with N' ranging from 1 to N respectively
  - AP@K: K = cost budget

task = 1 task = 2 task = 3 task = 4

Relevant

Relevant

Relevant

Not retrieved

Relevant

task = 1 task = 2 task = 3 task = 4

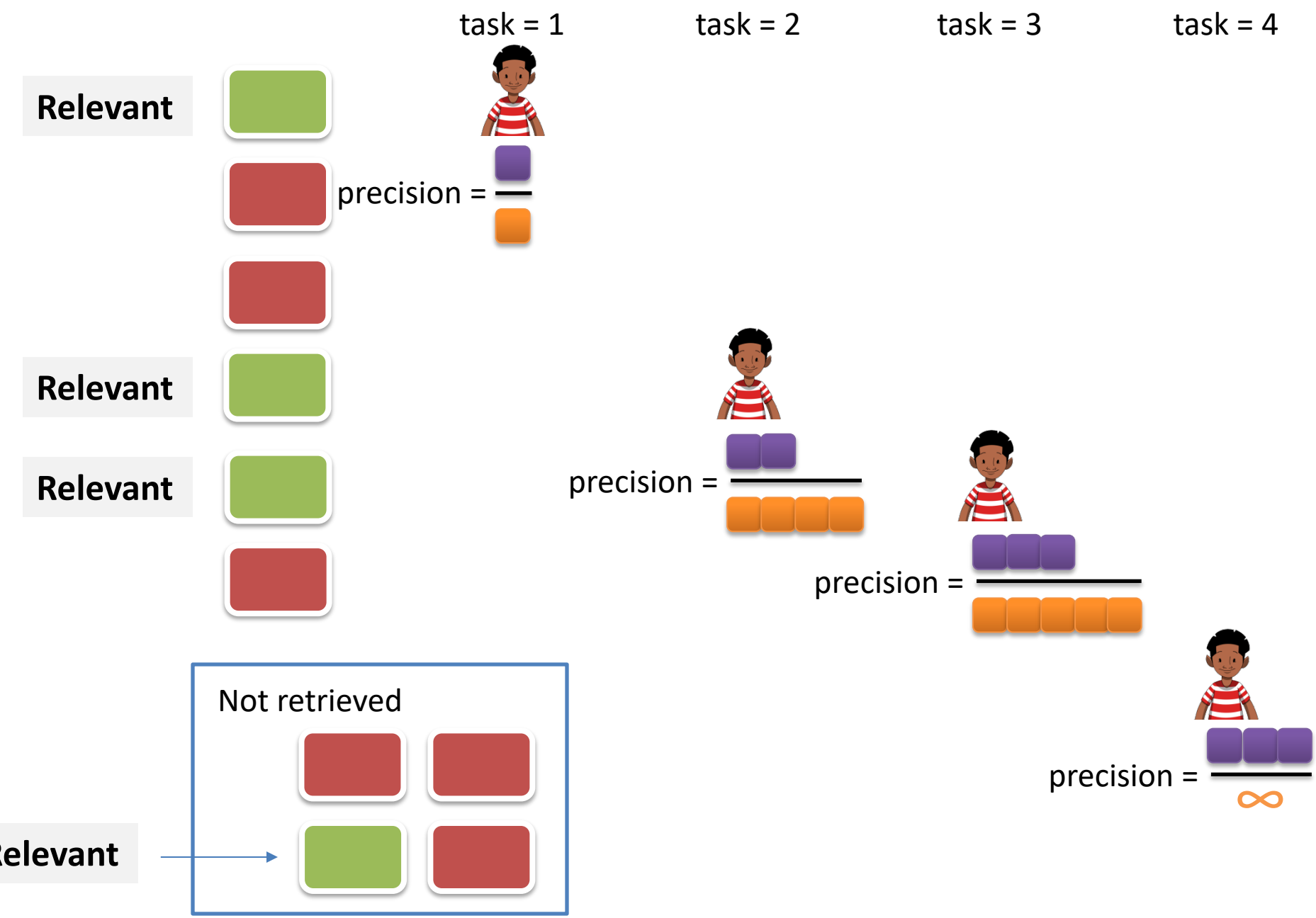
Relevant

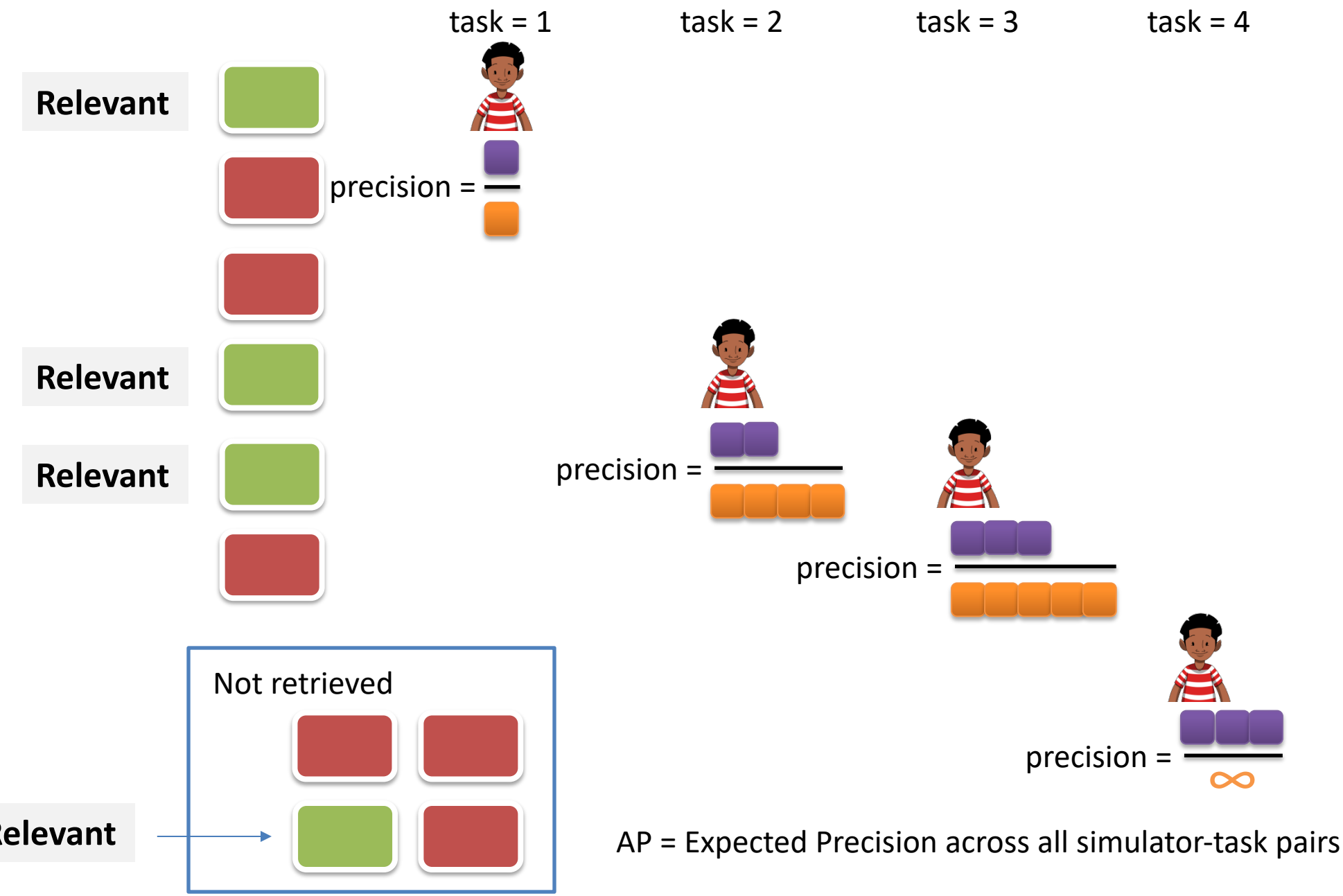
Relevant

Relevant

Not retrieved

Relevant


task = 1
task = 2
task = 3
task = 4
Relevant
Relevant
Relevant
precision =
precision =
precision =
precision =
Not retrieved
Relevant


task = 1
task = 2
task = 3
task = 4
Relevant
Relevant
Relevant
precision =
precision =
precision =
precision =
Not retrieved
Relevant
AP = Expected Precision across all simulator-task pairs

Other metrics can also be studied in the framework

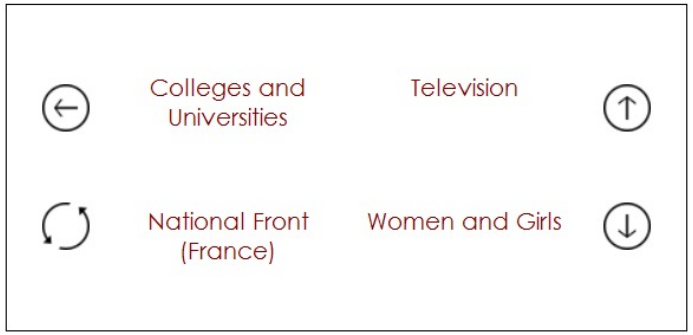
- Classical IR
  - Mean Reciprocal Rank (MRR)
  - Normalized Discounted Cumulative Gain (NDCG)
  - Ranked-Biased Precision (RBP) [Moffat & Zobel 08]
  - Time-based gain [Smucker & Clarke 12]
- Session IR
  - Session NDCG
  - U-measure based on trail-text

Evaluation of tag-based search interfaces [Zhang et al. 17]

- Example of search interface beyond ranking
  - Traditional interface: static layout
    - Medium screen: tag list alongside document list
    - Small screen: only tag list or document list at a time, and user needs to click "switch" to switch between the two lists
  - ICM interface: dynamic layout
- Evaluation based on simulators
  - Task: find target document(s)
  - Simulator never stops until task is complete
  - Metrics: interaction cost

Tag-based search interfaces: simulator action model

- If a target document is shown, user always clicks it
- Otherwise, if a tag related to a target document is shown, user always clicks it
- Otherwise:
  - On ICM: User always goes to “next page”
  - On medium static interface: user scrolls document list with probability τ, and scrolls tag list with probability (1- τ)
  - On small static interface:
    - If user is on document list, user scrolls list with probability τ1 and switches list with probability (1- τ1)
    - If user is on tag list, user scrolls list with probability τ2 and switches list with probability (1- τ2)



Simulator scrolls list with probability τ_2 and switches list with probability $(1- \tau_2)$

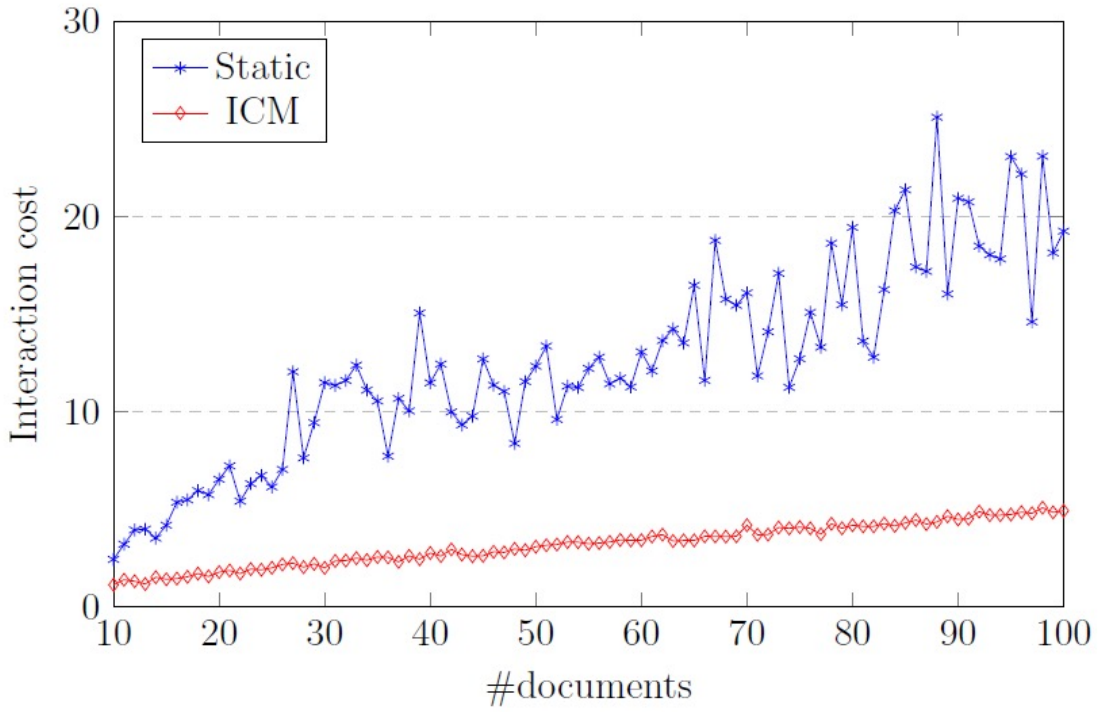
Simulator scrolls list with probability τ_1 and switches list with probability $(1- \tau_1)$

Simulator scrolls document list with probability τ, and scrolls tag list with probability $(1- \tau)$

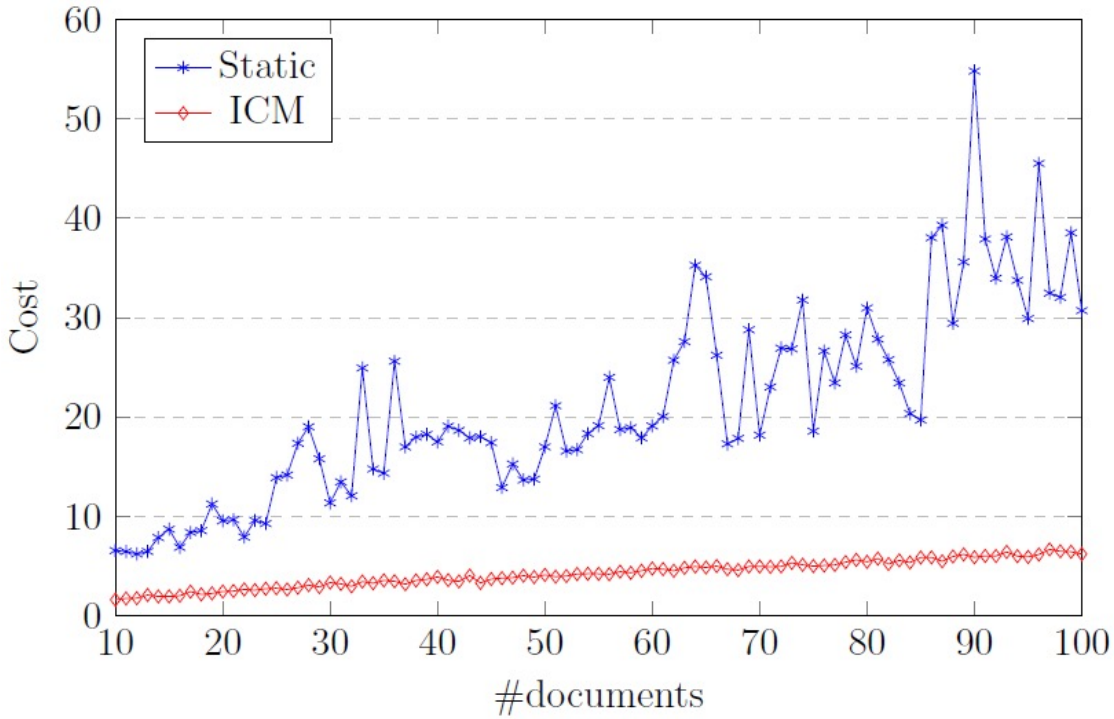
Results of simulation-based evaluation

Interface Card Model has consistently lower interaction cost than the static interface

Medium Screen



Small Screen



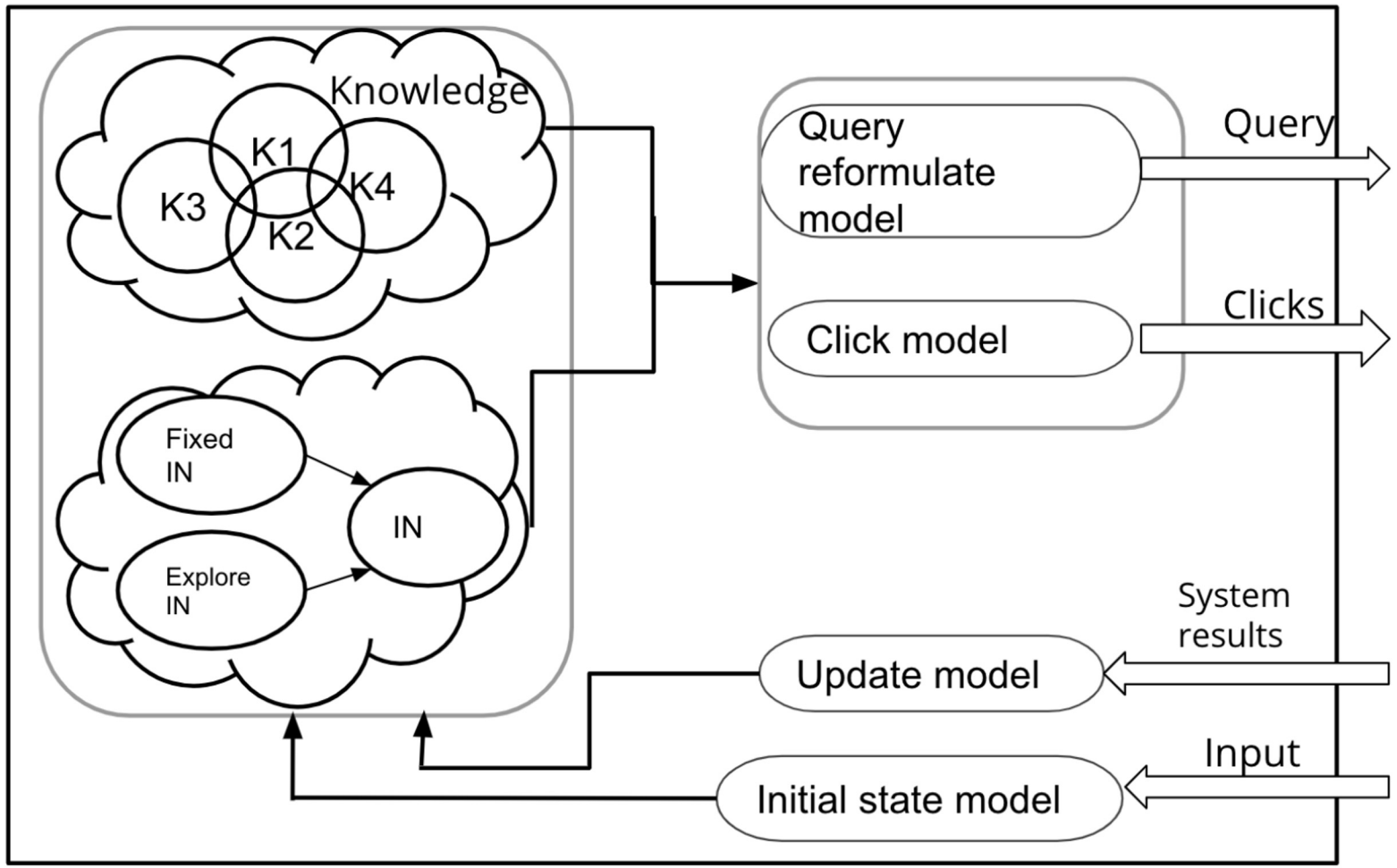
Validation from real user experiment

- Real user experiment
  - ICM is more efficient than static interface
  - The difference is higher on small screen than on medium screen
- Insights about real user behavior
  - Users can well utilize the tag list on the medium screen, but cannot make full use of the tag list on the small screen

Screen size	Sample size	Workers' average
Small	42	$\hat{\tau}_1 = 0.845$, $\hat{\tau}_2 = 0.370$
Medium	38	$\hat{\tau} = 0.211$

Table 6.2: Real user action averages

A Cognitive State User Model [Labhishetty et al. 20]



Interpretation of the model parameters

Modeling behaviour of doing more exploration in the session or fixing/focussing on one item quickly.

- λ_1 - indicates to what extent user prefers to explore even after observing many products.
- λ_2 - indicates to what extent user prefers to explore even after longer session.

Modeling behaviour in choosing what type of knowledge source is used for making queries

- α_{K1} - the background knowledge is preferred to use.
- α_{K2} - knowledge learnt during the session is given more preference or likely to learn more during session
- α_{K3} - has a lot of prior knowledge about the product domain, the words used to describe.
- α_{K4} - similar words are preferred/likely to use in queries, maybe because of lack of knowledge in other sources

Learning from search logs

- E-comm Search log sessions are used to estimate IN and K.
- Experiments: CSUM is trained and fitted to the search sessions to learn parameters. It is evaluated on test search sessions for performance in prediction.
- CSUM is used to analyse user behaviours in search log.

Analysis of user behaviour using the model

We use the estimated parameters of the cognitive state of the model on the search log to reveal and understand any user behaviour patterns.

As we have six parameters: $\{\lambda_1, \lambda_2, \alpha_{K1}, \alpha_{K2}, \alpha_{K3}, \alpha_{K4}\}$, each session can be represented as a data point in 6 dimension space where this representation is from user behaviour perspective.

From Table1,2, average variance of all parameters (var_u , var_p) is high for same user session relative to same product session. Variance of λ_1, λ_2 is generally higher indicating users differ more in fixed/exploration behaviour.

UserId	$\lvert S_u \rvert$	var_u	λ_1	λ_2	$\alpha_{K_1})$	α_{K_2}	α_{K_3}	α_{K_4}
Mean	-	0.311	0.059	0.064	0.042	0.052	0.047	0.046
1	12	0.426	0.076	0.076	0.059	0.067	0.088	0.06
2	7	0.446	0.088	0.073	0.065	0.069	0.062	0.088
3	7	0.439	0.088	0.088	0.051	0.068	0.088	0.055

ProductId	$\lvert S_p \rvert$	var_p	λ_1	λ_2	α_{K_1}	α_{K_2}	α_{K_3}	α_{K_4}
Mean	-	0.239	0.045	0.035	0.046	0.039	0.037	0.039
1	25	0.443	0.089	0.053	0.089	0.053	0.079	0.078
2	19	0.353	0.089	0.089	0.059	0.041	0.018	0.054
3	18	0.410	0.049	0.061	0.085	0.061	0.075	0.078

Table1: Variance of the parameters for same user sessions, Table2: variance for same product sessions

Common user behaviour patterns through clustering

Clustering of user sessions using 6-dimensional data points. Clustering is done with different number of clusters: 2, 3, 32.

1.With just 2 clusters, only λ1, λ2 varies indicating most distinguishable user behaviours patterns are exploration and fixed IN behaviours.

2.With 3 clusters, α_{K3} ,α_{K4} differ in cluster centers - thus the next major difference in behaviour is different type of knowledge source used for making queries using only product space words(K3) or using only similar words outside product space (K4)

ClusterId	Cluster size	λ_1	λ_2	α_{K_1}	α_{K_2}	α_{K_3}	α_{K_4}
0	799	0.659	0.133	0.392	0.38	0.404	0.340
1	779	0.139	0.635	0.413	0.407	0.269	0.400
0	513	0.607	0.159	0.385	0.433	0.151	0.475
1	687	0.111	0.686	0.425	0.408	0.286	0.385
2	378	0.628	0.156	0.386	0.315	0.679	0.202
0	86	0.105	0.669	0.651	0.672	0.128	0.130
8	85	0.102	0.693	0.643	0.159	0.148	0.662
5	76	0.695	0.103	0.179	0.142	0.695	0.147
3	75	0.105	0.695	0.161	0.159	0.129	0.657

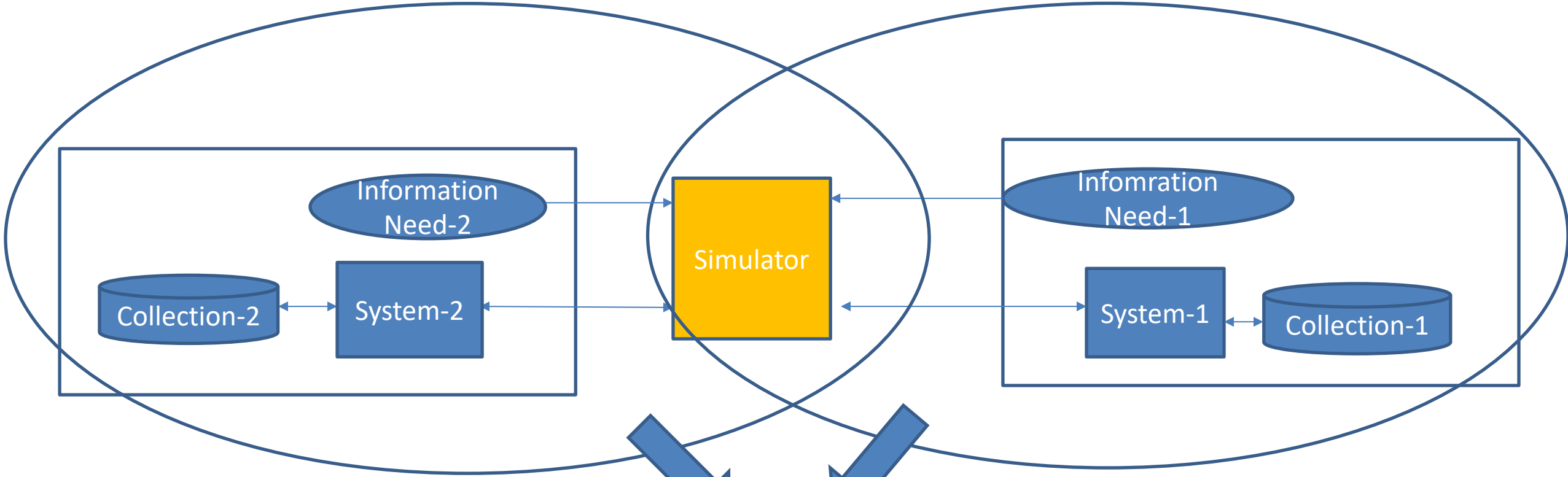
Table 3. Cluster centers

Summary

- A cognitive user model for E-commerce search (CSUM), modeling all major user actions:
  - Query formulation
  - Query reformulation
  - User clicks
- Novelty of the model
  - Modelling of user cognition - Information need and knowledge and update of the knowledge
  - Interpretable model, with parameters that meaningfully correlate with different user behaviors.
- Interpretable simulator as a tool to mine search E-comm search log to identify interesting user behavior patterns.

A New Approach: Tester- based Evaluation of Simulators [Labhishetty & Zhai 21]

Tester: A set of IR systems where we have a certain expected performance pattern about the order of the performance.



Can come from real users, but don't have to

Expected Pattern

Tester

Performance Measure of Simulator (Success/Failure Rate, …)

Evaluation: Can the testers distinguish different simulators in a meaningful way?

Testers: 1.Query History (QH) Tester - S1: BM25, S2: BM25+previous queries data

2.Click History (CH) Tester - S1: BM25, S2: BM25+previous clicks data

3.BM25 Ablation - S1:BM25, S2:BM25 \ TF,IDF

4 Representative Simulators:

Simulators	Description
SIT - Smartest (Ideal)	Smart query, Ideal clicks, ``*Disgust*'' point stopping[1] (sequentially non-relevant documents).
SST - Smart & Stochastic	Smart query, Stochastic clicks, ``*Disgust*'' point stopping.
STST - Single-term & Stochastic	Single term query, Stochastic clicks, ``*Disgust*'' point stopping.
RU - Random	Random query, Random clicks, Random stopping.

Table2: Success rate (Sr), Failure rate (Fr) based on sDCG/q

Random User is the worst in all cases (as expected)

Testers	"Smartest (Ideal)"		" Smart & Stochastic"		"Single-term & Stochastic"		"Random"	
	Sr	Fr	Sr	Fr	Sr	Fr	Sr	Fr
M+QH (1, 0.5)	0.766	0.149	0.745	0.192	0.809	0.085	0.021	0.0
M+QH (1, 0.01)	0.723	0.192	0.723	0.213	0.872	0.021	0.021	0.0
M+CH (1, 0.8)	0.787	0.043	0.723	0.192	0.638	0.149	0.021	0.0
M+CH (1, 0.5)	0.809	0.021	0.638	0.277	0.575	0.192	0.0	0.0
(M\TF, M)	0.340	0.638	0.383	0.617	0.362	0.468	0.0	0.043
(M\IDF, M)	0.532	0.404	0.426	0.532	0.0	0.0	0.0	0.043
Avg	0.635	0.252	0.599	0.334	0.517	0.182	0.009	0.027
Avg w/o TF, IDF testers	0.771	0.101	0.708	0.218	0.723	0.112	0.016	0.0

Overall, the "smartest" user is the best (as expected)

BM25 Ablation Tester didn't behave as expected

Summary

- The Proposed Tester based approach is effective and feasible approach to evaluate reliability of user simulators; there are multiple challenges that should be further studied. E.g.,BM25 ablation tester.
- Testers themselves should be first reliable to use in the evaluation.
- Open Tester & Simulator Benchmarking
  - Turn all TREC collections into simulators & create a comprehensive set of Testers to evaluate all of them.

Demo: http://timan102.cs.illinois.edu/tesir

Future Directions

- How should we evaluate a user simulator?
  - As a whole vs. component evaluation
  - Similarity to one user vs. a group of users
  - Application-oriented evaluation (e.g., Tester-based evaluation)
  - Non-interpretable (black box) vs. interpretable (white box) simulators
- How can we create a sustainable ecosystem to "publish" user simulators and improve them over time?
  - As a web site (e.g., Living Lab [Jagerman et al. 18], Evaluation-as-a-Service [Hopfgartner et al. 18], TESIR [Labhishetty & Zhai 21])
  - As a toolkit (e.g., SimIIR: https://github.com/leifos/simiir)

Future Directions (cont.)

- How can we develop realistic and interpretable user simulators?
  - Can we develop a general formal model of a user (e.g., POMDP)?
  - How can we define the tasks (formally)?
  - How can we model the cognitive state of a user and its transition during the interaction?
  - How can we deal with the dependency of user simulation on a retrieval system? Standardization of an interactive system (e.g., the Interface Card Model)?
  - How can we develop simulation models for EVERY user action?
  - How can we incorporate knowledge about users from many studies in related areas such as Information Science, HCI, Economics, and Psychology?
- Other uses of user simulation: analysis of user behavior, augmentation of training data, training reinforcement learning algorithms, ….

Immediate Action:

Turn all static TREC collections into collections of **user simulators** [Carterette et al. ICTIR 15]

Carterette, Ben, Ashraf Bah, and Mustafa Zengin. "**Dynamic test collections for retrieval evaluation**." In *Proceedings of the 2015 international conference on the theory of information retrieval*, pp. 91-100. 2015.

Thank You!

Questions/Comments?

czhai@illinois.edu

http://czhai.cs.illinois.edu/