CWI
Centrum Wiskunde & Informatica
University of Glasgow
VIA VERITAS VITA
Universiteit van Amsterdam

Untangling Result List Refinement and Ranking quality:

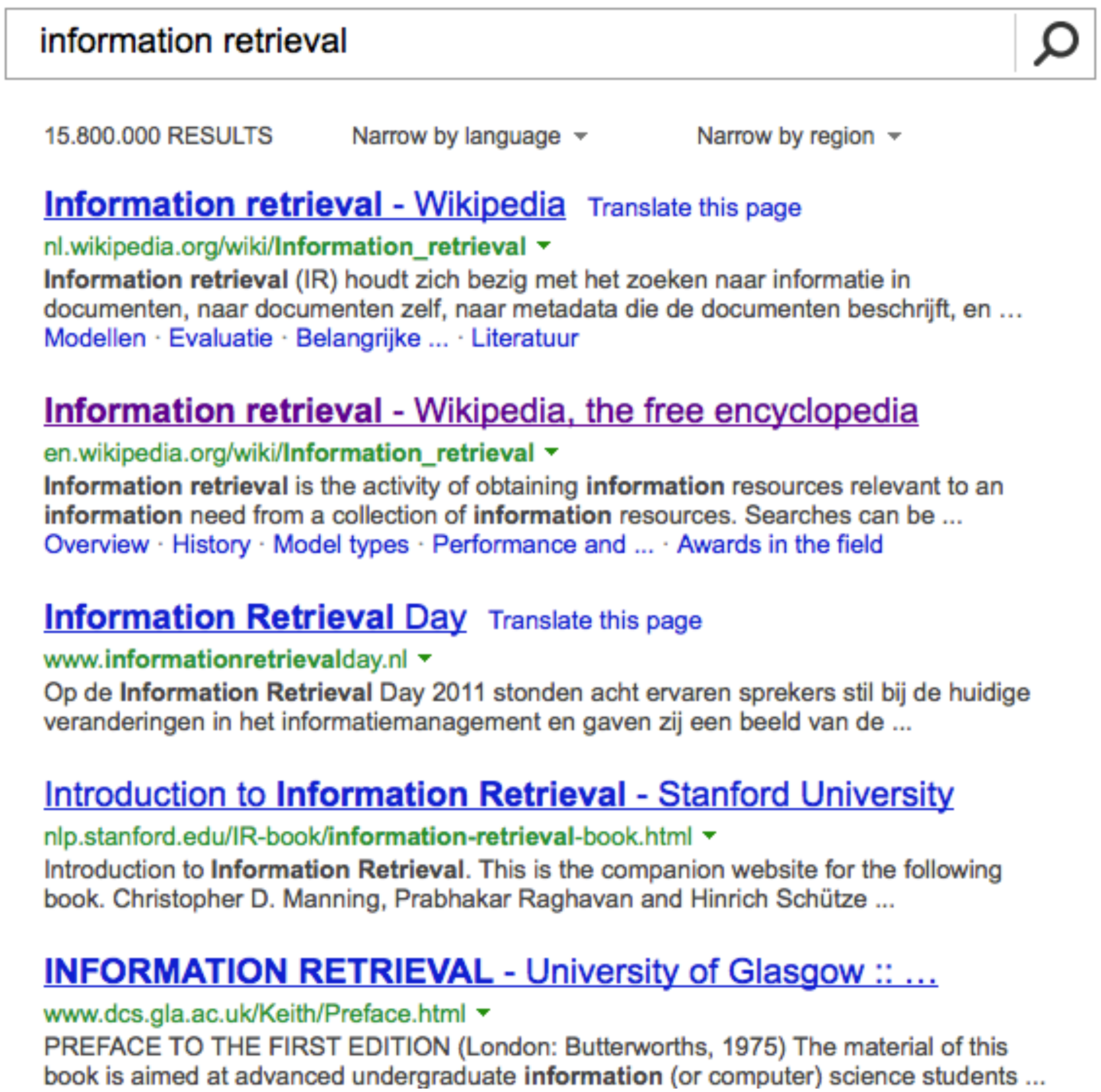
A Framework for Evaluation and Prediction

Jiyin He, Marc Bron, Arjen de Vries,
Leif Azzopardi, and Maarten de Rijke

Batch Evaluation

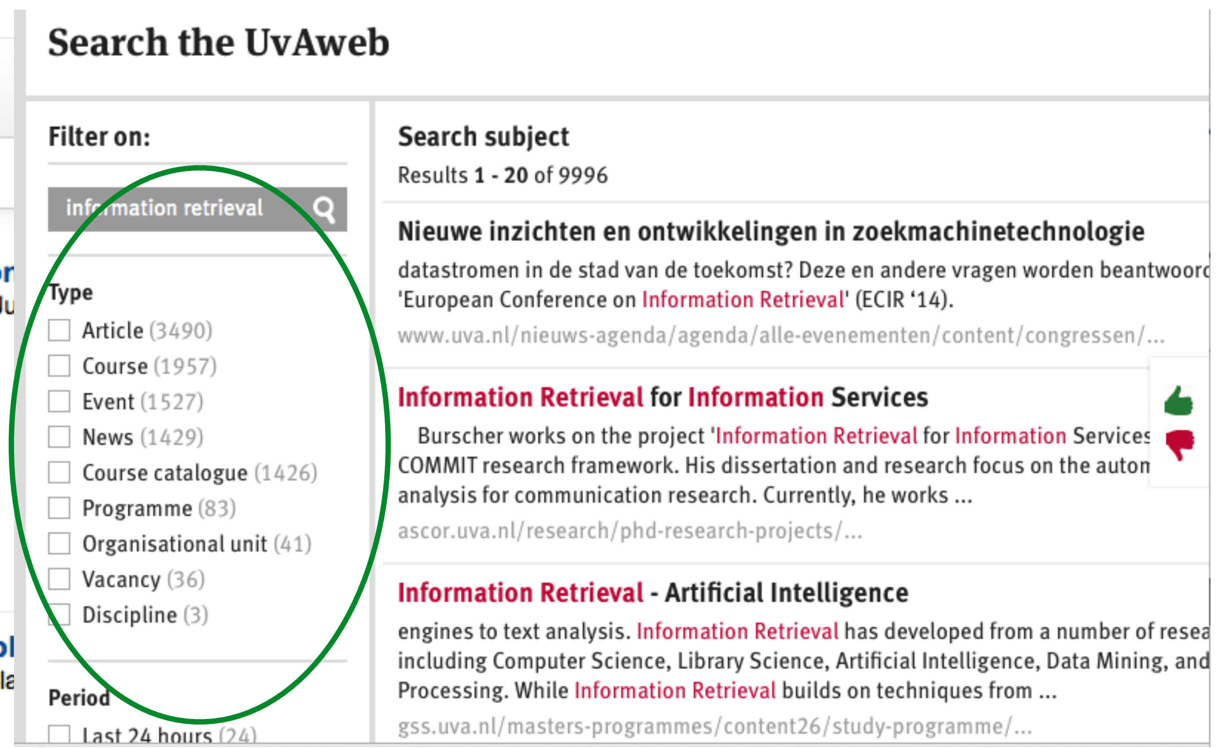
- Cost effective evaluation: prediction of search effectiveness based on a series of assumptions on how users use a search system
- Requirements:
  - A collection of documents
  - A set of test queries
  - Relevance judgements
  - An evaluation metric

Evaluation metrics and user interaction



- Evaluation metrics (Carterette, 2011)
  - A user interaction model: How users interact with a ranked list
  - Associating user interactions with effort or gain
  - Current batch evaluation metrics: boils down to the ranking quality of the results

Beyond a ranked list



Categories

Facets

result list refinement (RLR) elements

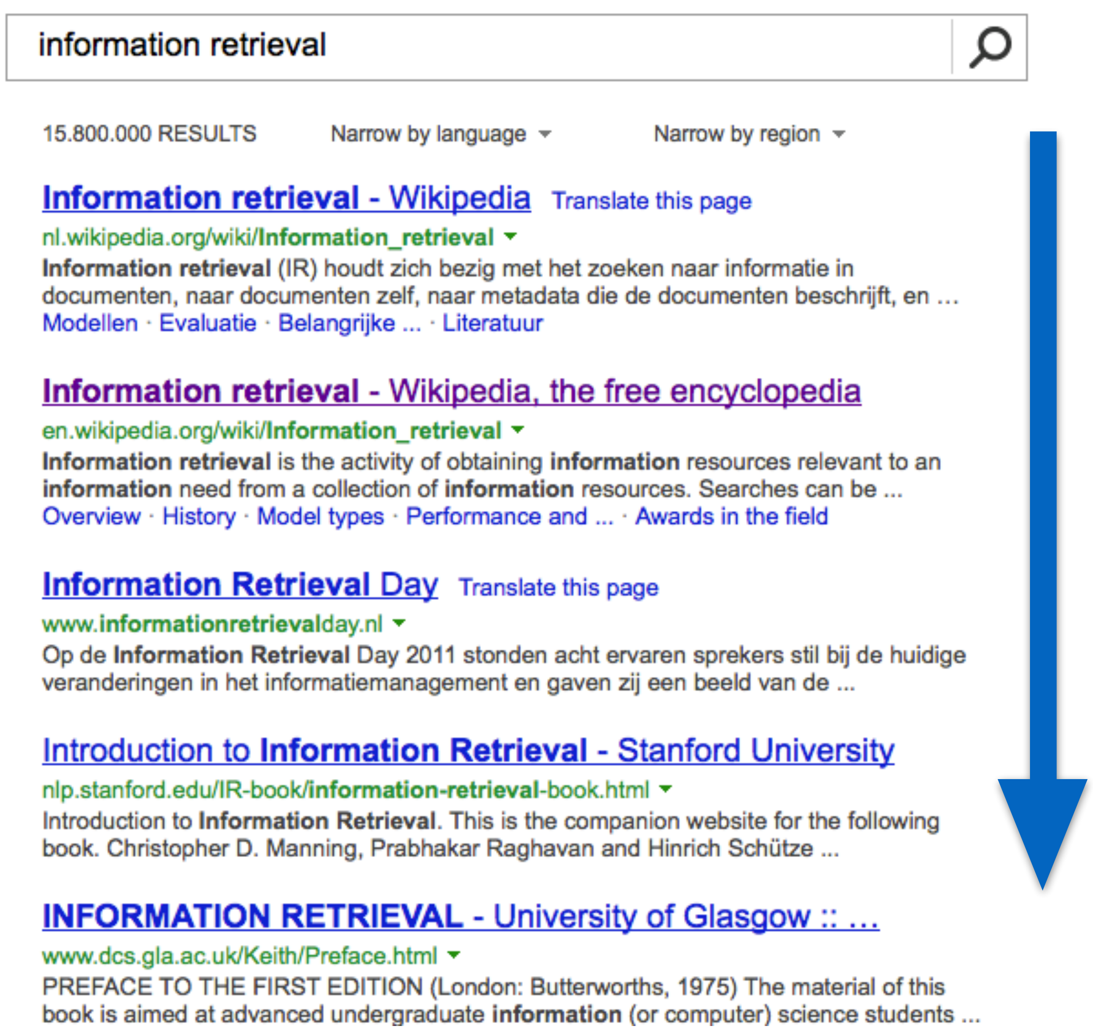
Q: how do we evaluate and compare systems under varying conditions of ranking quality, interface elements, as well as different user search behaviour?

Our solution

- An effort/gain-based user interaction model
  - How users interact with a ranked list and the RLR elements
  - Associating user interactions with effort and gain

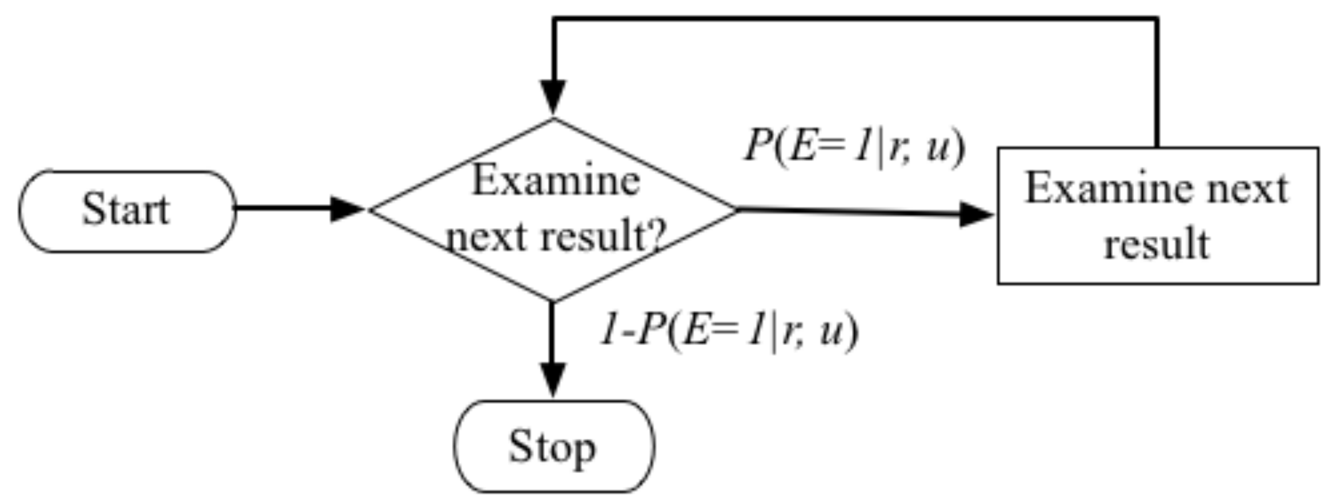
- Applications
  - Prediction: system performance w.r.t a particular application and user group
    - Model parameters derived from user studies
  - Simulation: whole system evaluation under varying conditions: ranking quality, interface elements, user types
    - Model parameters based on hypothesised values

Modelling user interaction: with a ranked list



E.g., following assumptions of user behaviour as in RBP

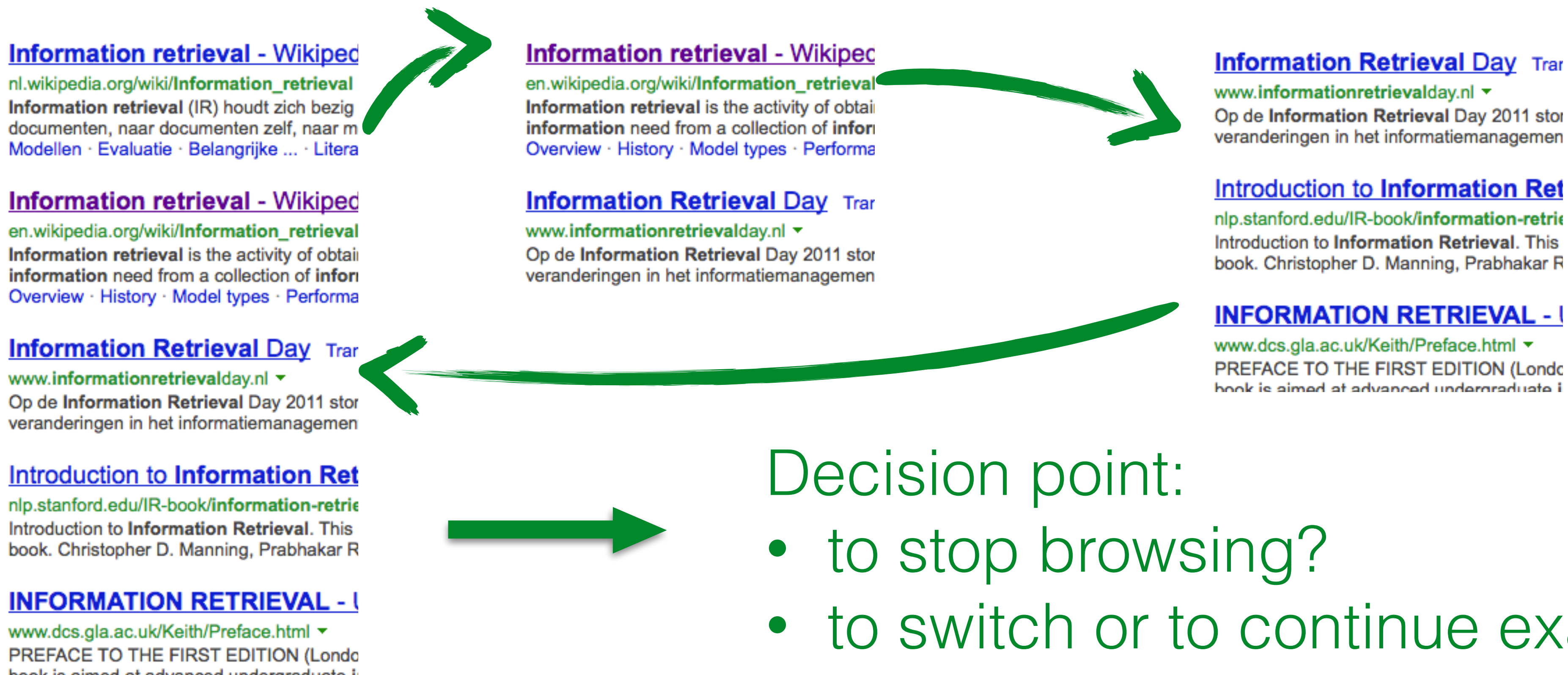
Parameter: continuation



Decision point: when to stop?

Modelling user interaction: with result list refinement

- Result refinement = switching between different filtered versions of the ranked list (sublists)



Decision point:
- to stop browsing?
- to switch or to continue examining?
- which one to select next?

Combinatory number of possible user paths:

- A Monte-Carlo solution

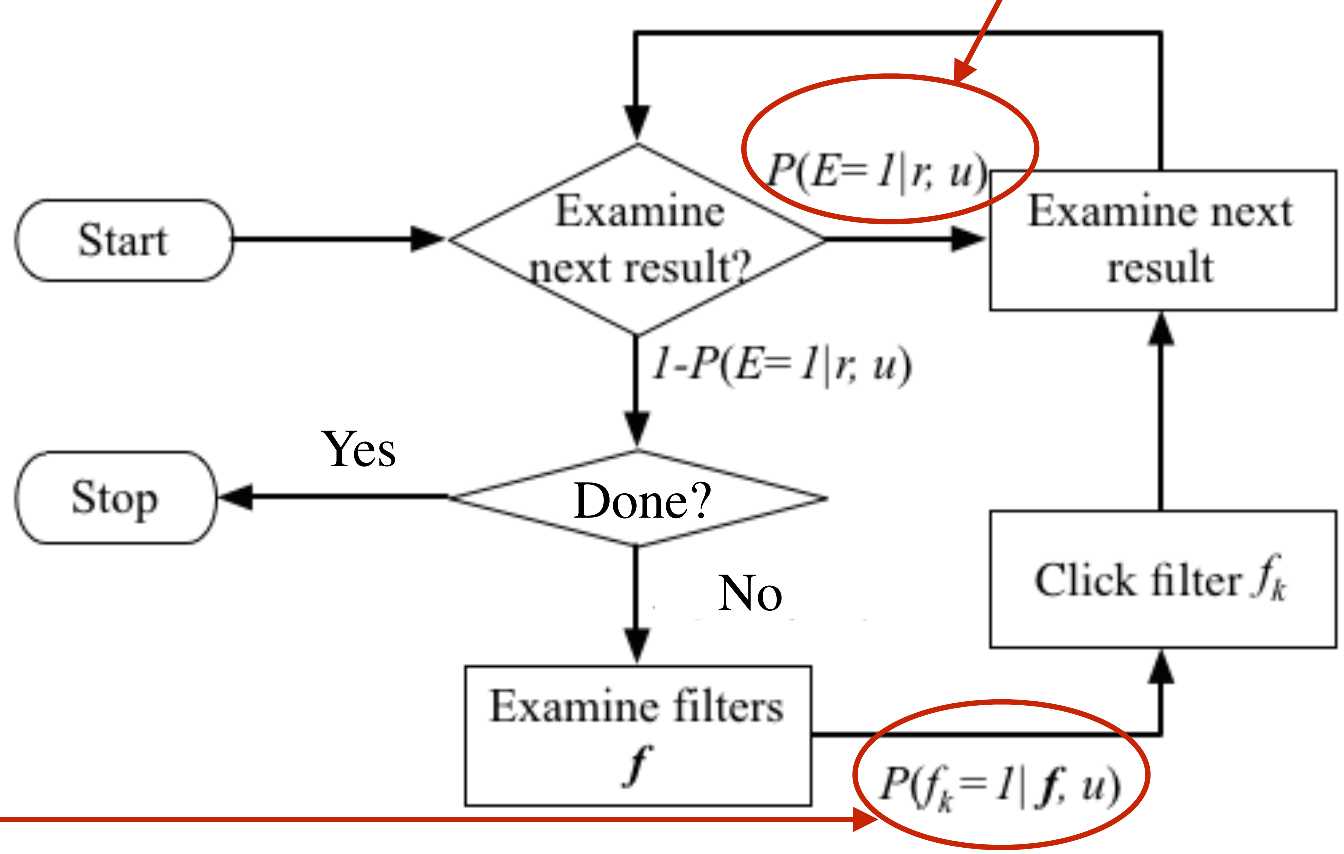
Modelling user interaction: with result list refinement

- Action path constraints
  - In each sublist, users browse top-down
    - common assumption; reducing possible paths from n! to constant
  - Users skip and only skip documents already seen
    - preventing inflated relevance and infinite switching
  - Deterministic quitting point
    - gain based: quit when certain amount of effort is spent
    - effort based: quit when certain amount of gain is achieved

Modelling user interaction: with result list refinement

- Action path constraints
  - In each list, users browse top-down
  - Users skip and only skip documents already seen
  - Deterministic quitting point
    - gain based: quit when certain amount of effort is spent
    - effort based: quit when certain amount of gain is achieved

Parameter 1: continuation



Parameter 2: List selection

User actions, efforts, and gain

- From user action paths to user efforts and gain
  - Each action is associated with an effort
  - Each action may or may not result in a gain, i.e., finding relevant document
- User actions
  - Examine result, refine a list, pagination
- Simple assumption about effort and gain
  - Equal unit effort for all actions → Total effort = # actions
  - Equal unit gain for all relevant documents → Total gain = # relevant docs found

Validation of prediction

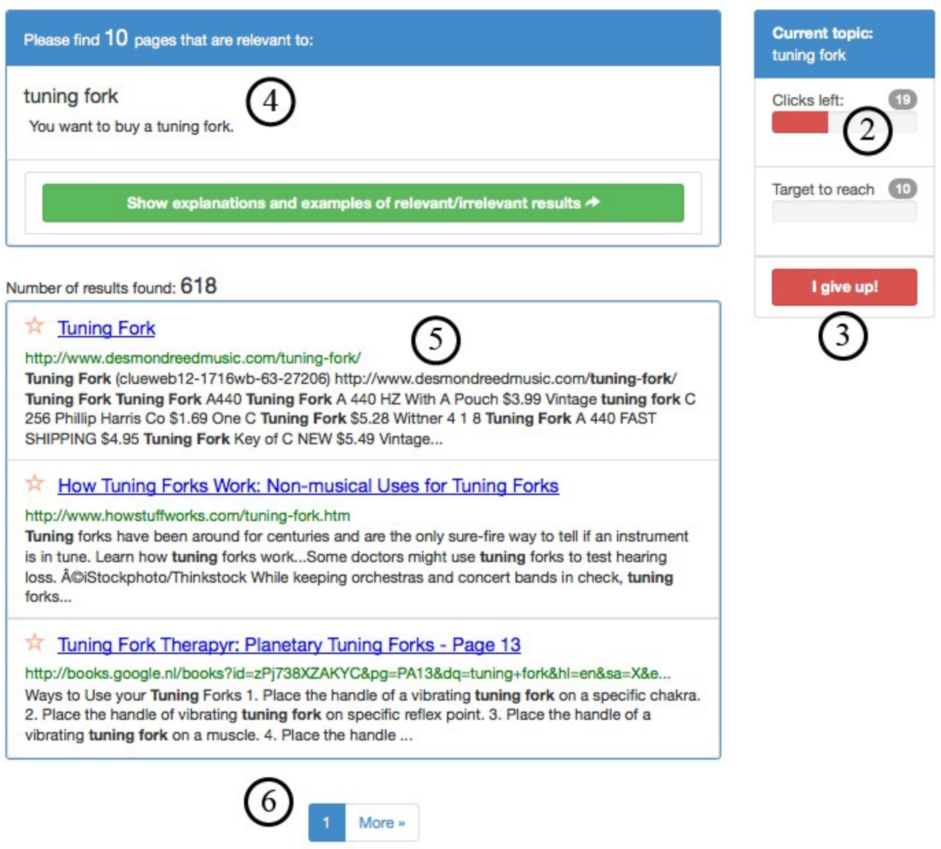
- RQs
  - Does the predicted effort correlate to user effort derived from usage data?
  - Can we accurately predict when a RLR interface is beneficial, compared to a basic interface?
- 3 Steps
  - Obtaining usage data from user study
  - Measuring (real) user effort
  - Predicting user performance by calibrated user interaction model
- Data
  - TREC 2013 Federated Search track
  - 50 topics with retrieved web pages and snippets, all judged
  - Results from 108 verticals, each associated with one or more categories

Obtaining usage data: study design

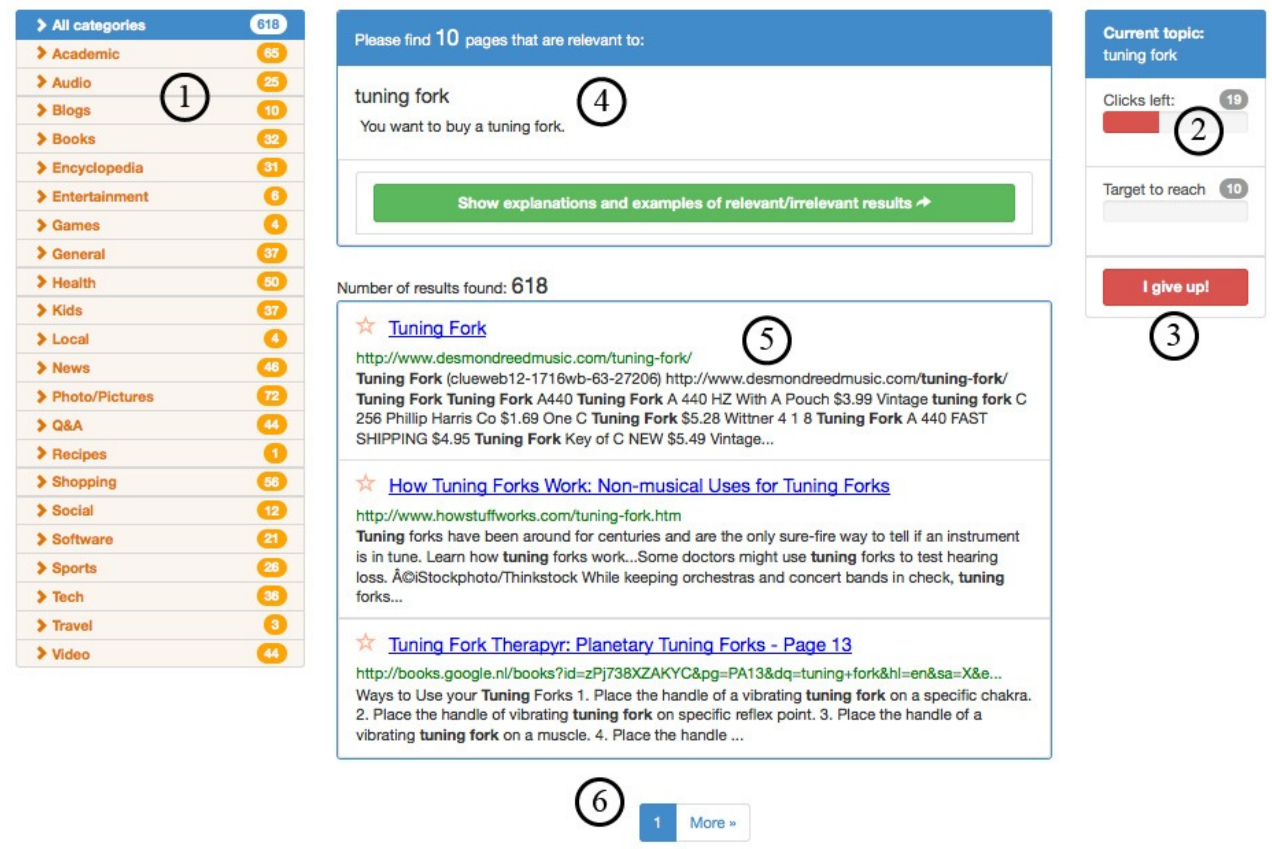
- User task (He et al., 2014)
  - Finding 10 relevant documents
    - Manageable effort, potential for considerable effort save
  - within 50 clicks
    - Preventing randomly clicking all results
  - Snippet based relevance judgement with user feedback
    - Reduced user variability in relevance judgement
- Experiment design
  - Between subject
  - Randomised topic and interface assignment

Obtaining usage data: interfaces

Basic interface



RLR interface



Obtained usage data

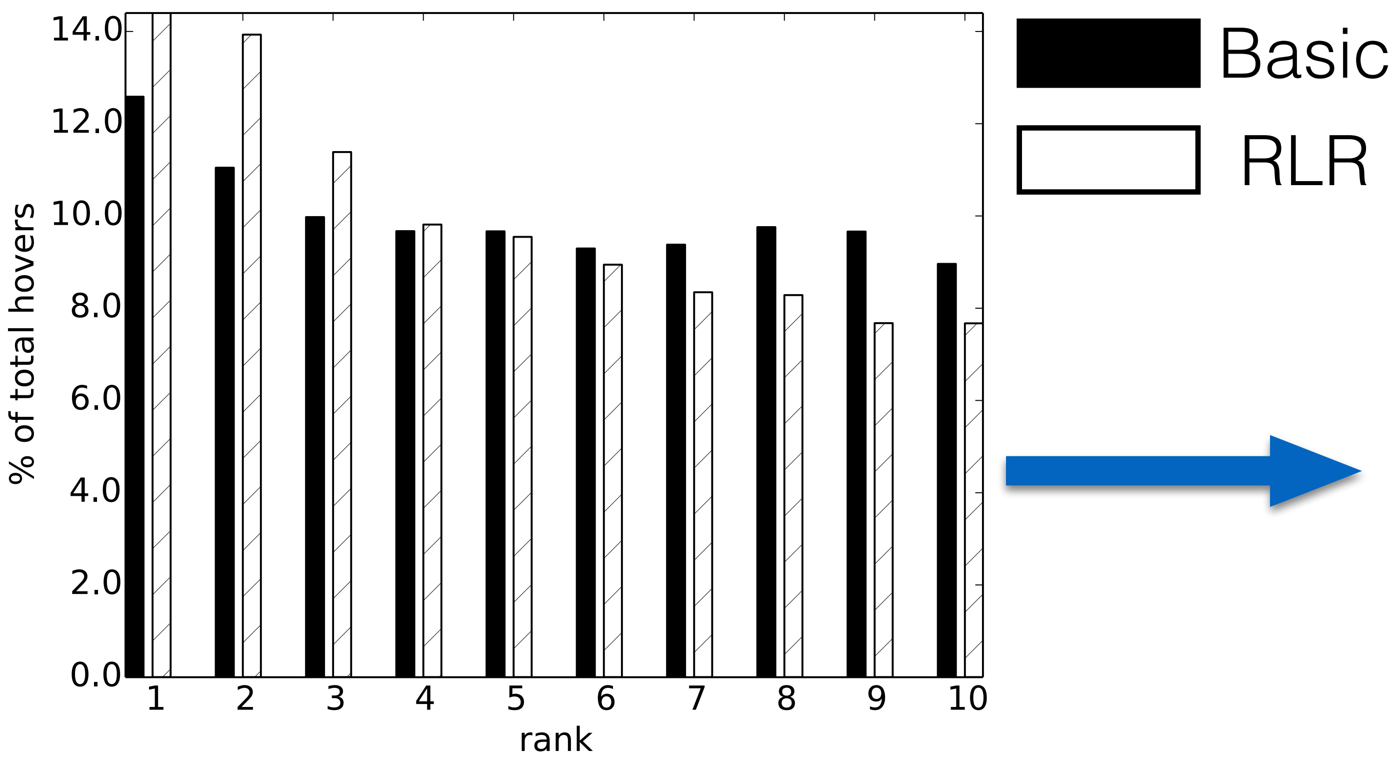
	Basic	RLR
Completed task instances	145 (Median p. task: 2)	255 (Median p. task: 3)
#Participants	49	48
#Uncompleted task instances	35	28

Measuring (real) user effort

- Examine result: mouse hover over a result snippet
  - # results visited on a SERP =

    all results in a page before a “pagination” action +

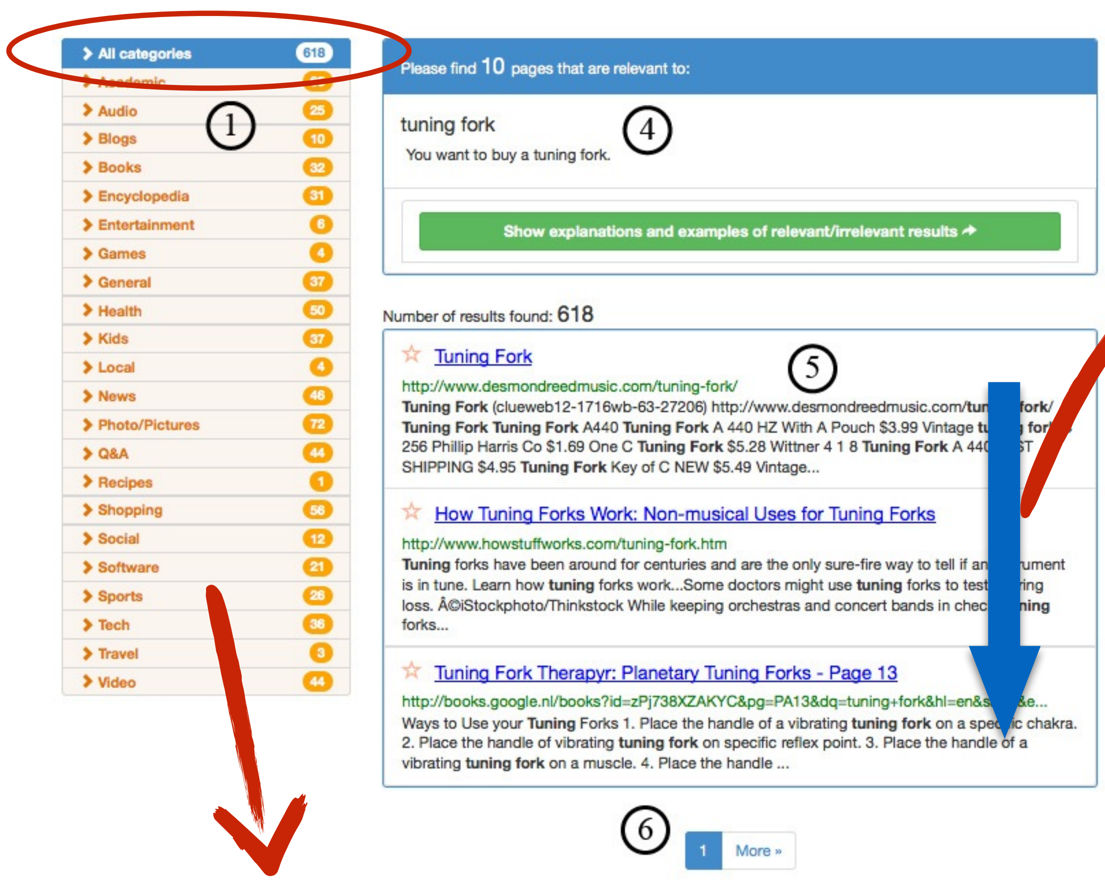
    up to the last clicked result on the last visited page



Mild position bias: as a result of snippet-based result examination

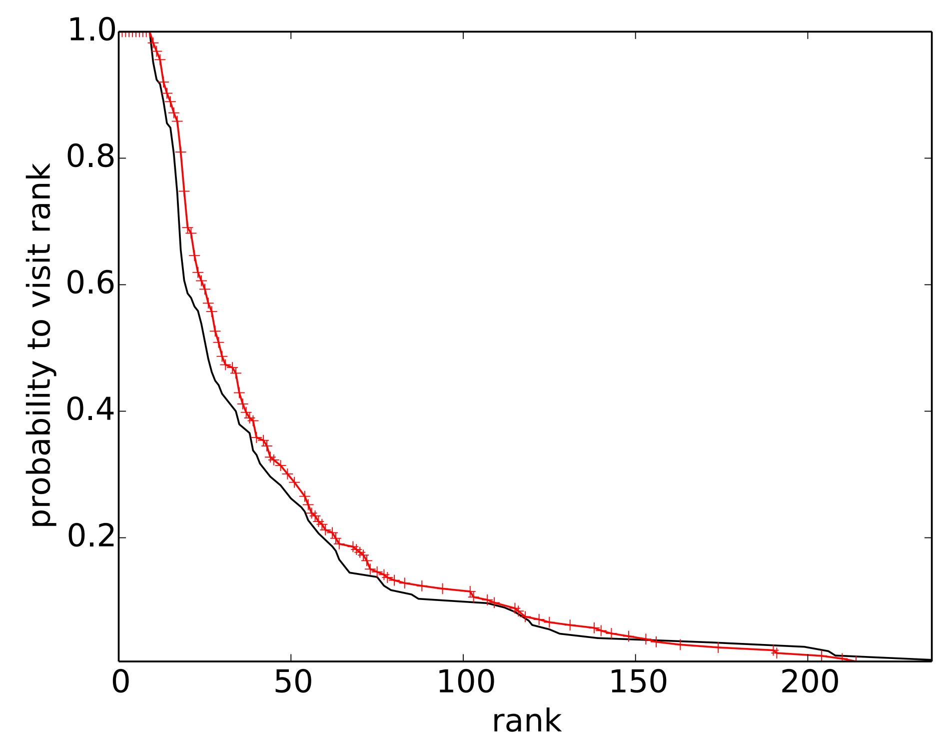
Predicting user effort with calibrated interaction model

Default selection



Parameter 1: continuation

Probability a result is visited @ rank *K*

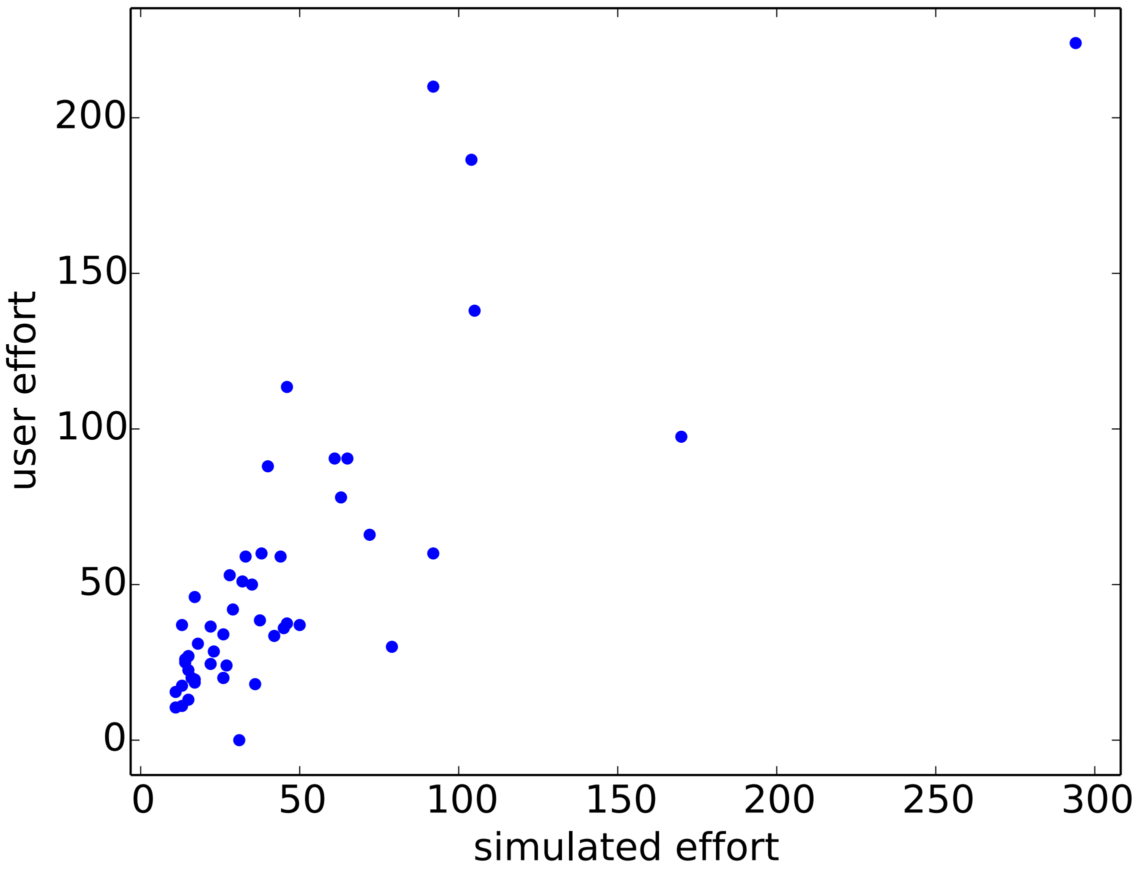


Parameter 2: List selection

- Per topic, the relative frequency that a filter is chosen
- Default selection: “All categories”

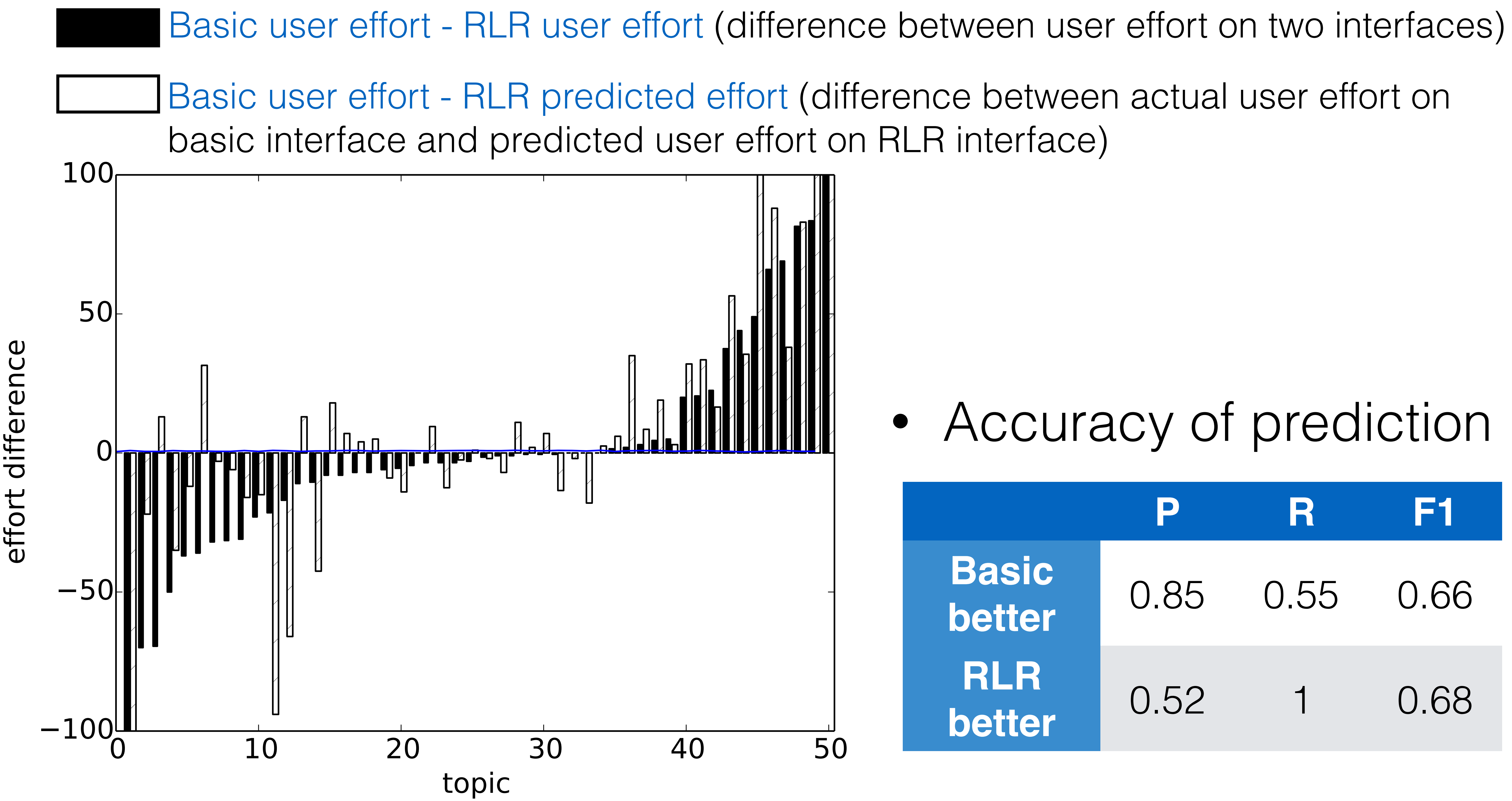
Q1: Does the predicted effort correlate to user effort?

- Predicted effort: an approximation of real user effort
  - Correlation as a measure for the accuracy of approximation



Pearson correlation between the predicted effort and user effort: **0.79** (p-value < 0.01)

Q2: Can we accurately predict when a RLR interface is beneficial?



- Accuracy of prediction

	P	R	F1
Basic better	0.85	0.55	0.66
RLR better	0.52	1	0.68

Validation of prediction: conclusions

- Our RLR user interaction model is able to accurately predict user effort

- Different interfaces are suitable for different queries (i.e., of different ranking quality)

- Model allows prediction of which interface is most suitable

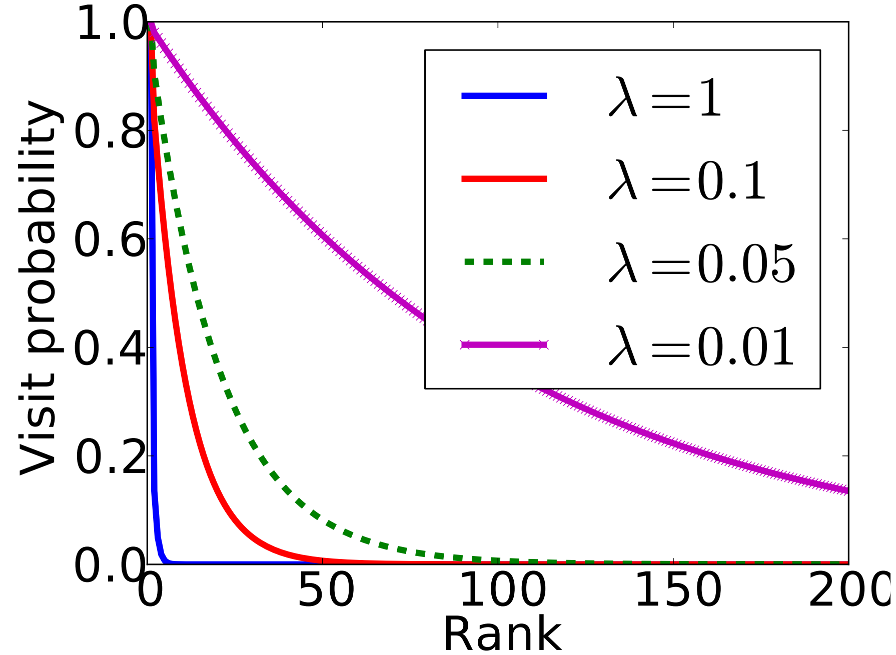
Whole system evaluation: hypothesised users

- RQs
  - When does an RLR interface help to save user effort compared to a basic interface?
- Study whole system performance under varying conditions:
  - Ranking quality
  - Sublist characteristics
  - User behaviour

Hypothesised user parameter setting

- Intuition: some users are more patient than others
- Parameter 1: continuation
  - at each rank *r*, draw a decision as a bernoulli trial
  - Bernoulli parameterised by a exponential decay function to approximate the empirical distribution of rank biased visit

$$P(E = 1|r, u) = e^{-\lambda r}$$



1: impatient users
0.01: patient users

Hypothesised user parameter setting

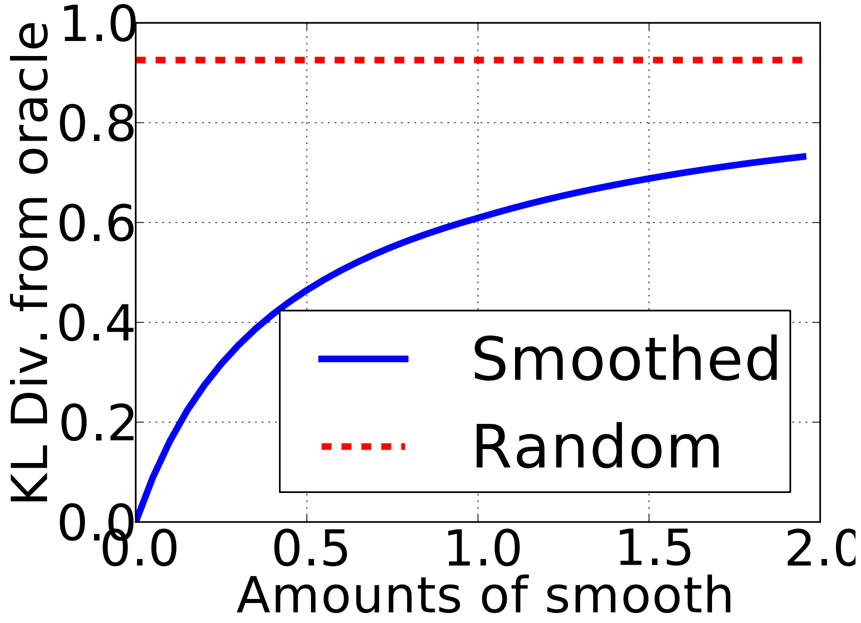
- Intuition: some users make better selection of sublists than others

- Parameter 2: list selection
  - draw a decision vector from a categorical distribution

$$f_1, ..., f_k \sim Cat(K, \vec{c})$$

  - setting user prior knowledge of the candidate lists with its conjugate prior

$$\vec{c} \sim Dir(K, \vec{\alpha})$$



Uniform: no idea what to select
NDCG: informed selection

Factors influencing RLR effectiveness

- Query difficulty for the basic interface (Dq)
  - Efforts needed to accomplish a task with basic interface
- Sublist relevance (Rq)
  - Averaged NDCG score over sublists of a query
- Sublist entropy (Hq)
  - Entropy of relevant documents distributed among sublists
- User accuracy (U)
  - Controlled by the amount of smooth added to the prior of list selection
  - Level 1 (oracle based on NDCG);
  - 15% (level 2), 50% (level 3), 67% (level 4) less accurate compared to level 1
- User task
  - to find 1, 10, or “all” relevant documents

Method

- Fit a generalised linear model (logistic regression)
- DV: whether a RLR interface outperforms (i.e., save efforts) a basic interface
- IVs: factors outlined above
- Model selection: forward and backward selection with Bayesian information criterion (BIC)
- Explain the relation between DV and IVs and their interactions

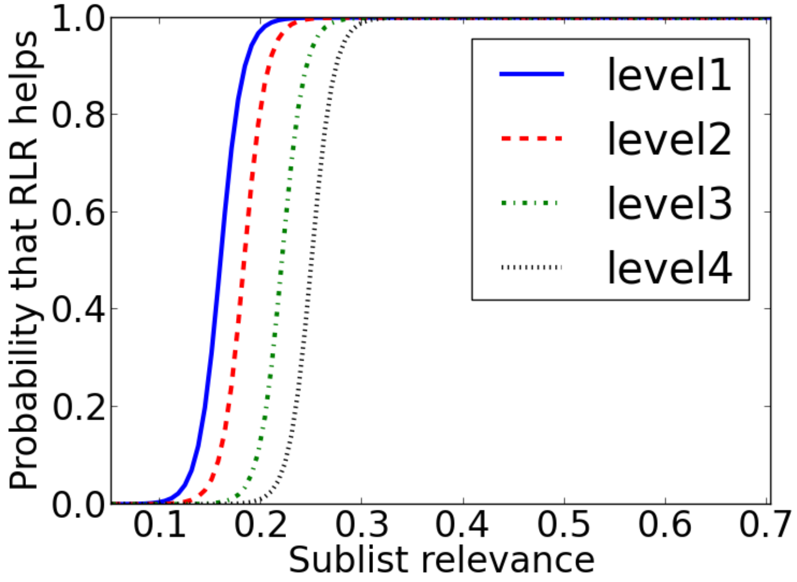
Main effects

Coefficients	Find-1	Find-10	Find-all
intercept	-7.340	-10.437	-0.534
Dq	0.106	-0.069	0.002
U-level2	3.223	**-2.131**	**-5.106**
U-level3	1.559	**-5.528**	**-8.014**
U-level4	-2.319	**-8.194**	**-8.014**
Hq	-1.044	3.635	-1.649
Rq	-	-49.792	**114.940**
Dq : U-level2	-1.655	-	-
Dq : U-level3	-2.004	-	-
Dq : U-level4	-2.068	-	-
Dq : Hq	**1.310**	-0.097	-
Dq : Rq	-	**3.263**	**0.091**
Hq : Rq	-	13.968	**-57.277**
Dq : Hq : Rq	-	**-0.842**	-

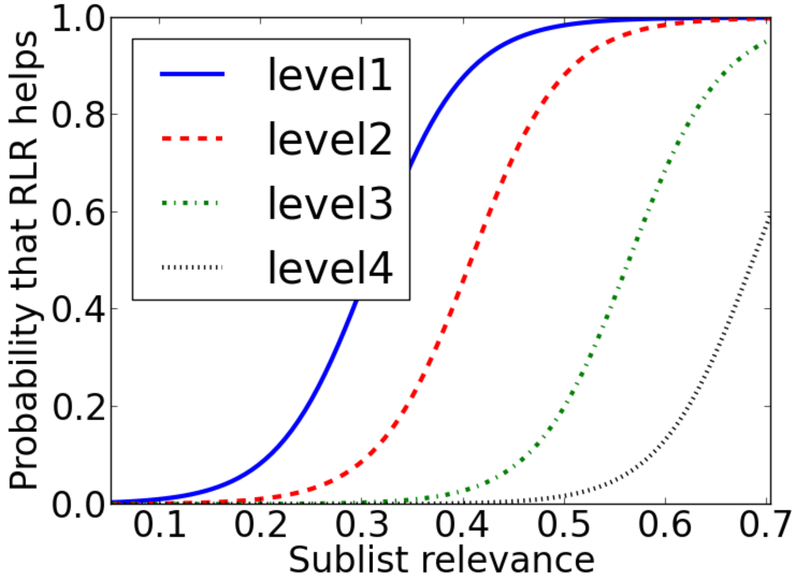
- Find - 1: none of the main effects are significant
- Find - 10 /all: users need to know which sublists to pick
- Find - all: having sublists with relevant documents ranked high is useful.

Interaction effects

Dq:Rq for Find -10



Dq:high; Hq:median

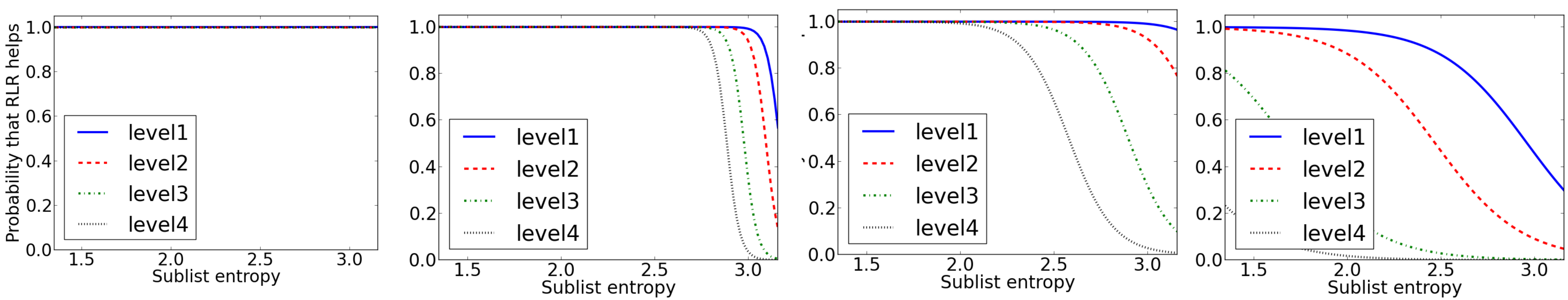


Dq:low; Hq:median

- When query is difficult for basic interface, sublists and users do not need to be very accurate for RLR to be more effective
- When query is easy for basic interface, higher quality of sublists and user accuracy are necessary

Interaction effects

Dq:Rq:Hq for Find -10



(a) Dq:high; Rq:high (b) Dq:high; Rq: low (c) Dq:low; Rq: high (d) Dq:low; Rq: low

- When query is difficult for basic interface, RLR is likely to be beneficial especially when few sublists contain most of the relevant documents
- When query is easy for basic interface, very specific conditions with respect to user accuracy, sublist relevance, and sublist entropy need to be met for RLR to be beneficial.

Relation to traditional metrics

	user effort	predicted effort	△ user effort
nDCG@10	-0.21	-0.19	0.02
nDCG@all	-0.42*	-0.34	0.00
NRBP	-0.41*	-0.33	0.08
AP	-0.63**	-0.54**	0.02
binary nDCG@10	-0.54**	-0.44**	0.02
binary nDCG@all	**-0.72****	**-0.59****	0.04
Our model	**0.79****	—	**-0.49****

Pearson's linear correlation; p-value < 0.01 (*); <0.001 (**)

- Query difficulty alone is not sufficient to predict whether a RLR interface will be beneficial

Whole system evaluation: conclusions

- When ranking quality is low, sub-lists and user sublist selection do not have to be of high quality for RLR to be more effective than basic;

- When ranking quality is high, only under specific conditions RLR may be beneficial, i.e., quality sub-lists, recall oriented task, accurate users;

- Implication for HCIR experiments: are your queries, user tasks, and ranking algorithms appropriate to study the properties of the interface?

Conclusions

- A user interaction model for evaluating search systems with result refinement elements
- By instantiate the model with parameter values derived from real usage data, we have validated the predictive power of our model
- By simulating users with hypothesised parameter values, we can investigate whole system performance under varying conditions concerning ranking quality, interface differences, user types, and task types