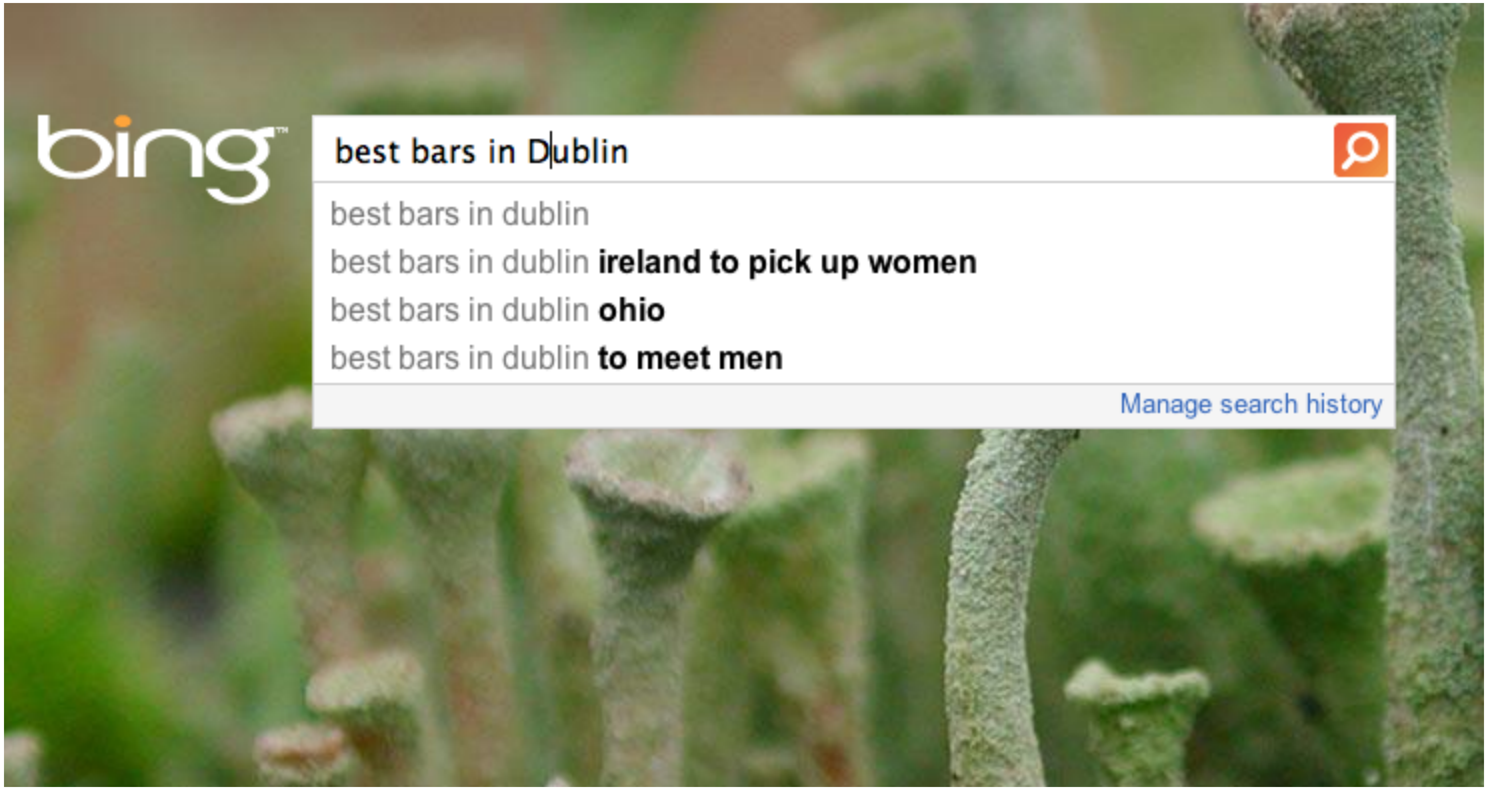
Search Engine Switching Detection Based on User Personal Preferences and Behavior Patterns

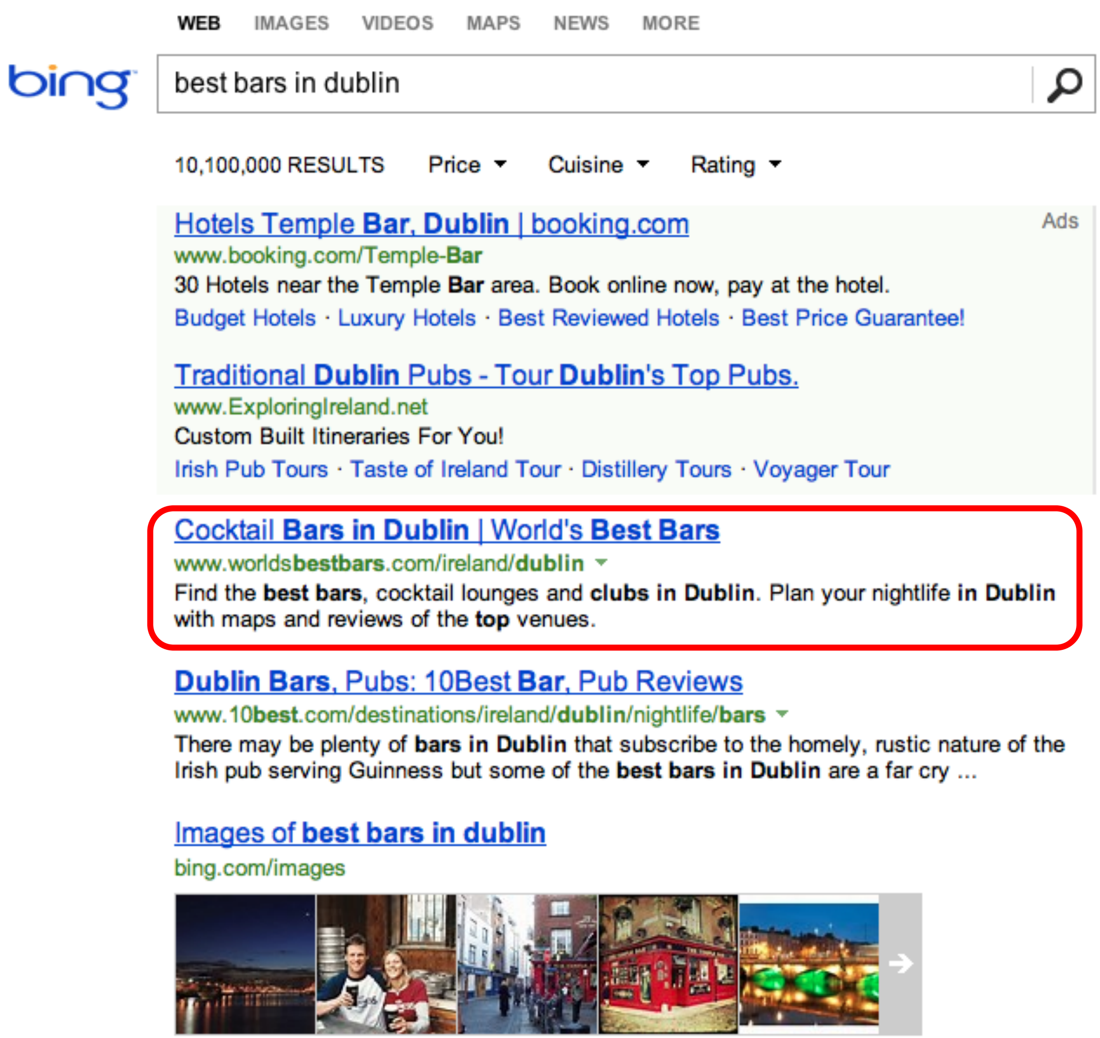
Denis Savenkov, Dmitry Lagun, Qiaoling Liu
Emory University, USA
{dsavenk,dlagun,qiaoling.liu}@emory.edu

presented by: Mikhail Ageev
Moscow State University, Russia
mageev@yandex.ru

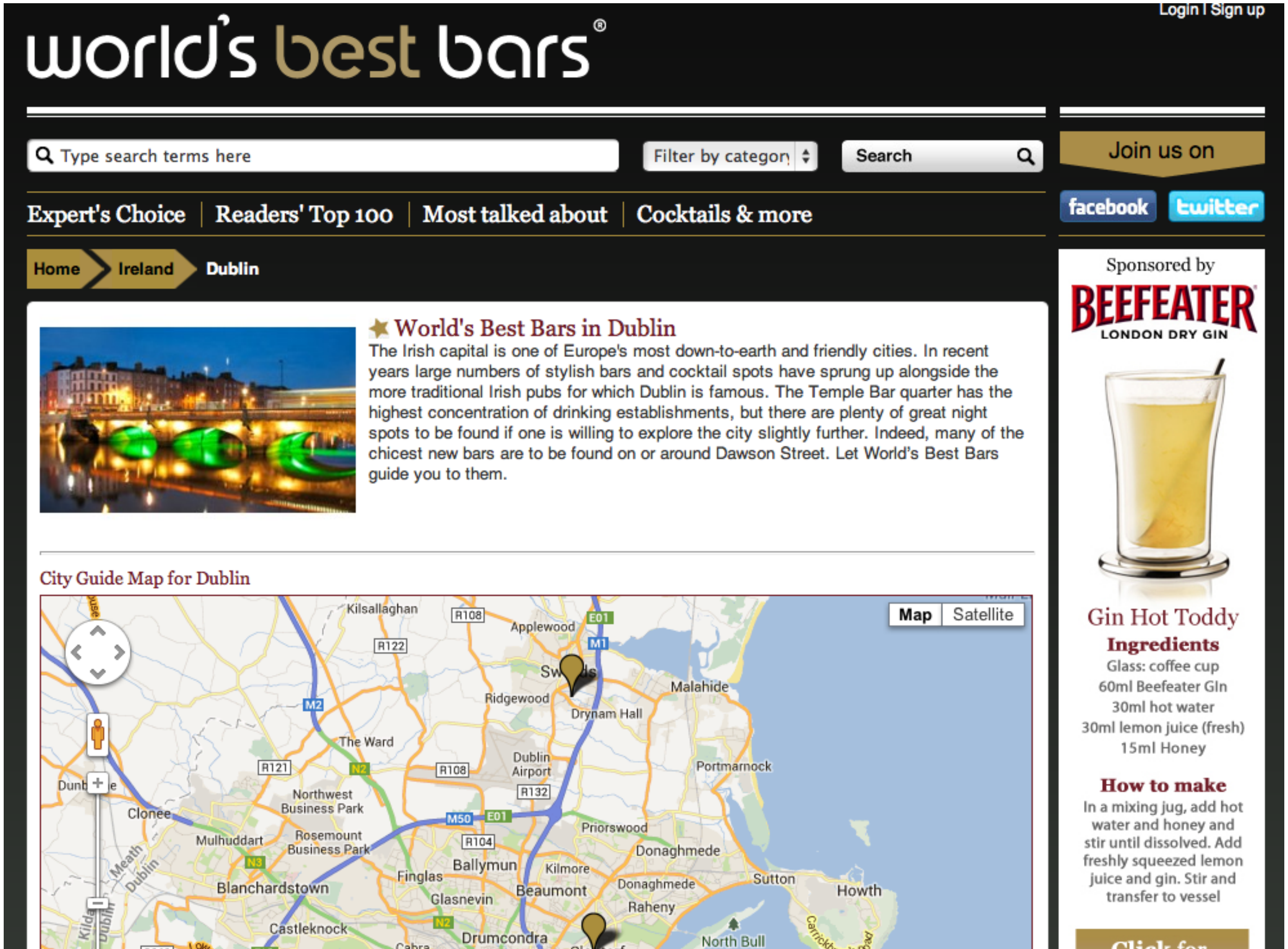
Search engine switching



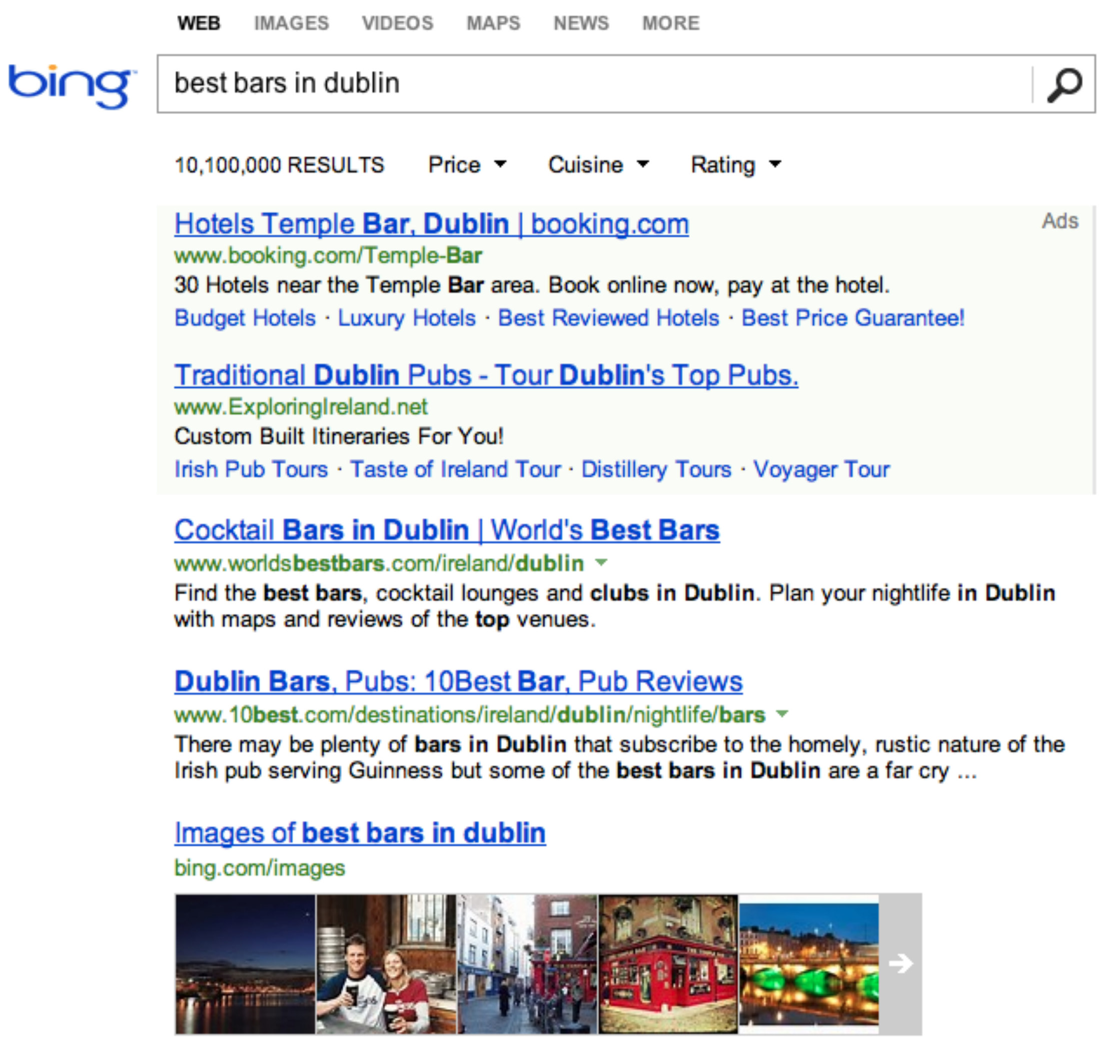
Search engine switching



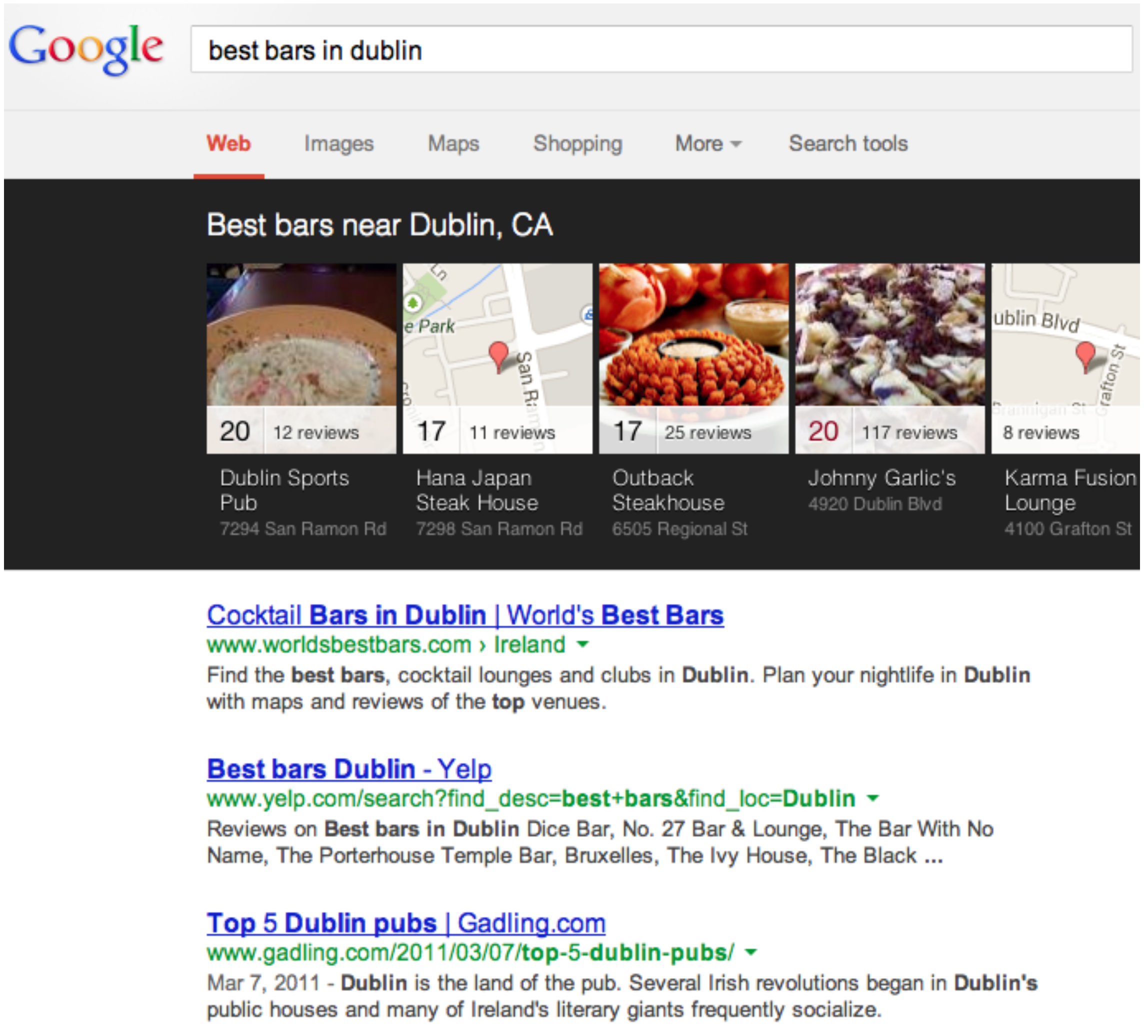
Search engine switching



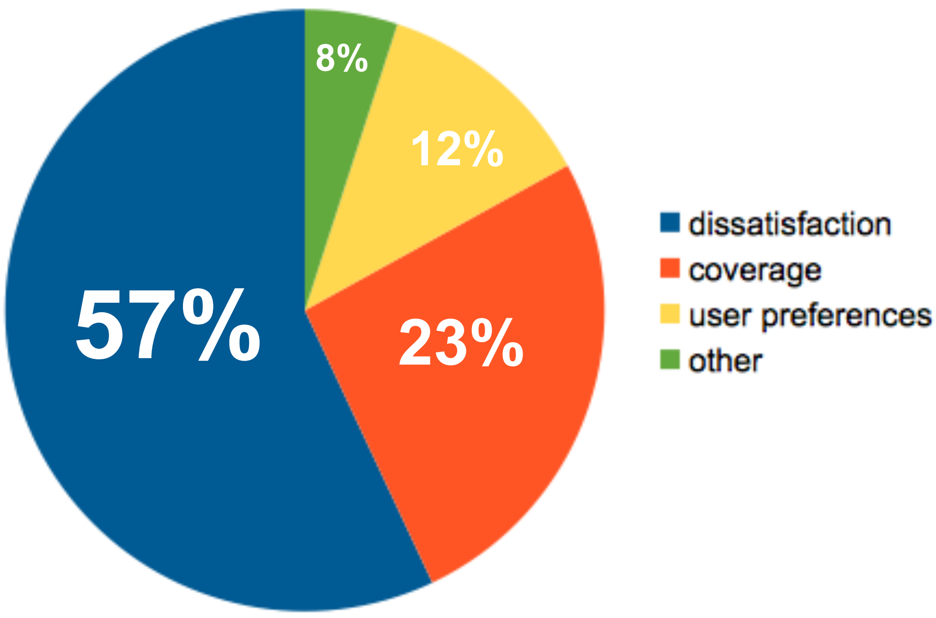
Search engine switching



Search engine switching



Why do people switch search engines?



Courtesy of Guo et al. SIGIR 2011

Motivation

- **57%** of switching cases is about user dis-satisfaction
  - can be used to improve search engine on problematic queries
- Caveat: not always possible to monitor directly
  - could be monitored using web browser (or toolbar)
  - could be monitored from search logs
    for navigational queries switching to another search engine
- Can we reliably detect switching? [our work]
  - e.g. can be used to improve search experience in such cases

Motivation

- High switching rate may indicate user dissatisfaction with the search engine
- Switching rate can be used for automatic search quality evalution
- Search engines could focus on improving user experience for searches followed by switching

Yandex Switching Detection Challenge

- Data: 30 days of <u>anonymized</u> search logs
  - 8,595,731 sessions (1,457,533 switching sessions)
  - 10,139,547 unique queries
- Task
  - detect search engine switching from user actions recorded in the search engine log
- Evalution
  - area under the ROC curve (AUC)

Related work

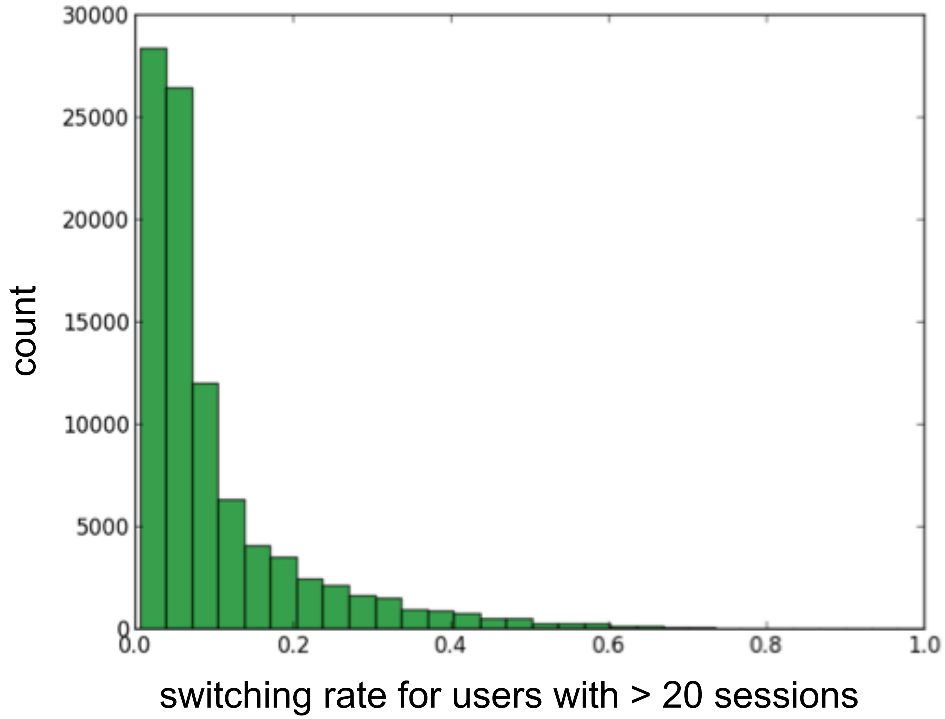
- Characterization of user actions specific to search engine switching
  [A. Heath and R. White, WWW 08]

- Prediction of search engine swithcing in online settings
  [R.White and S.Dumais, CIKM 09]

- Understanding and predicting switching rationales
  [Q.Guo et al., SIGIR 11]

- Personalized switching prediction and extensive experimentation
  [Our work]

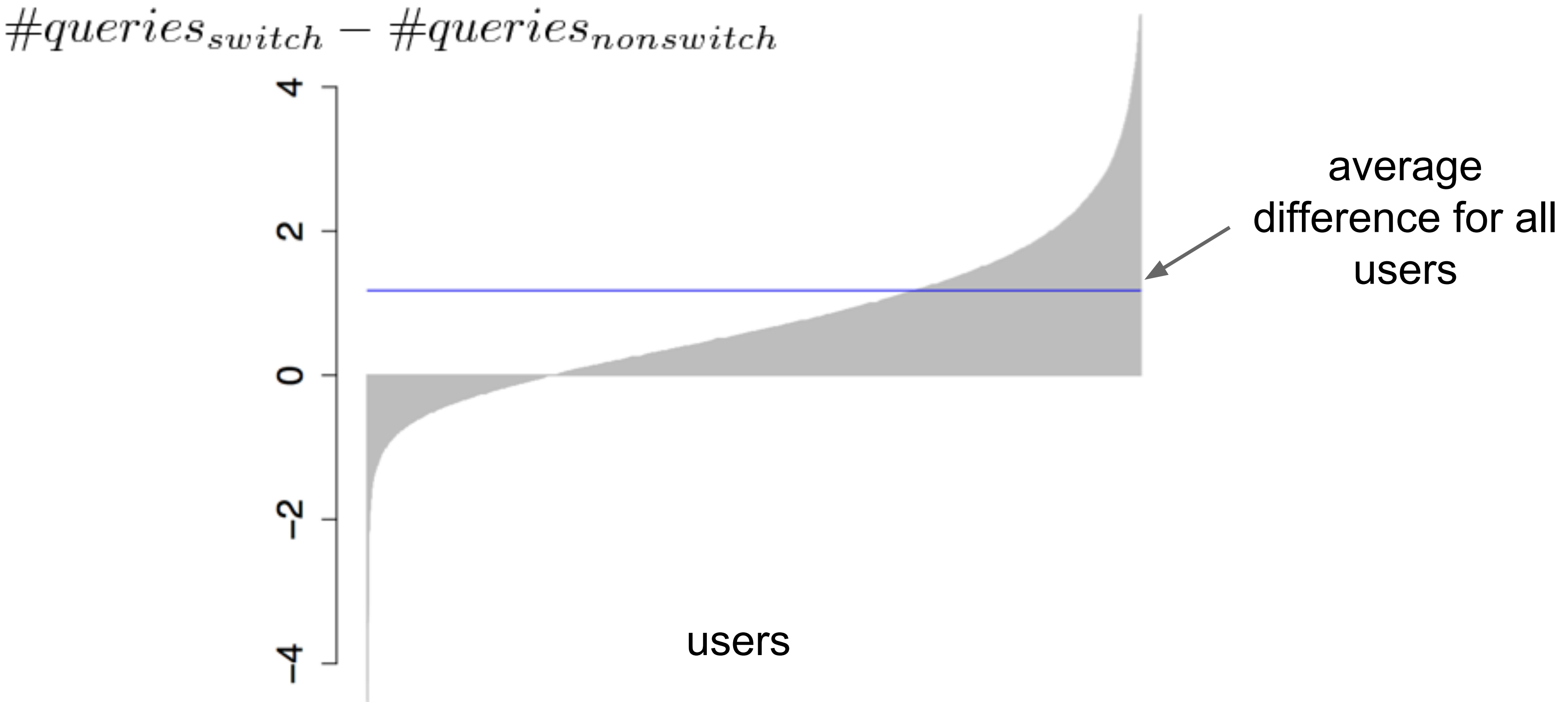
Insight: some users switch more frequently than others



possible reasons:

- user search experience varies
- switch depends on a search task

Insight: switching is more likely in longer sessions, but varies for users



Caveats:

- the effect is different for different users
- for some users the opposite is true

Switching detection: Main Idea

- switching is a *personal choice* of a user

- users are different
  - some users don't switch at all
  - some users are more persistent and could spend more time studying search results

- **Main Idea:** <u>build personalized model</u> that will learn user's personal habits and behavior patterns and use it for switching detection

Evaluation setup

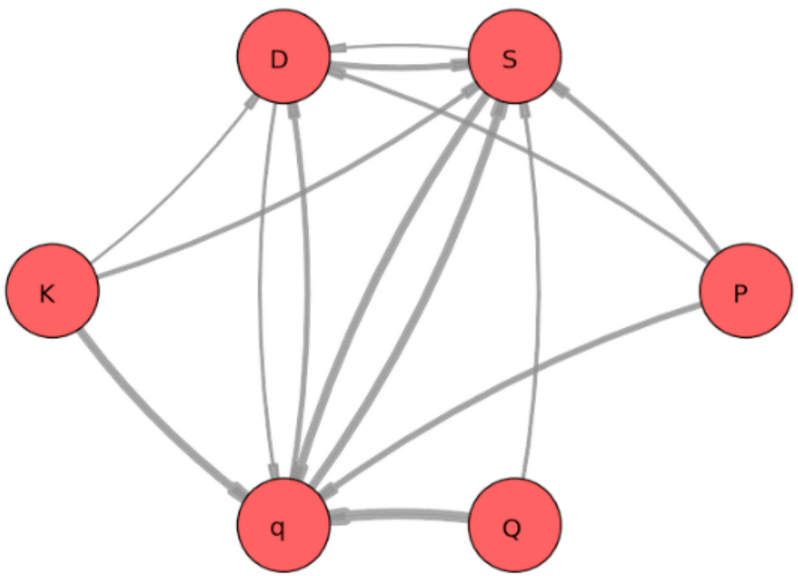
- Data
  - 24 days of search log data for training
    - 1-21 days used to calculate features
    - 22-24 days for machine learning
    - 25-27 days for validation
- Evaluation Metric
  - Area under the ROC curve (AUC)

Search Trails

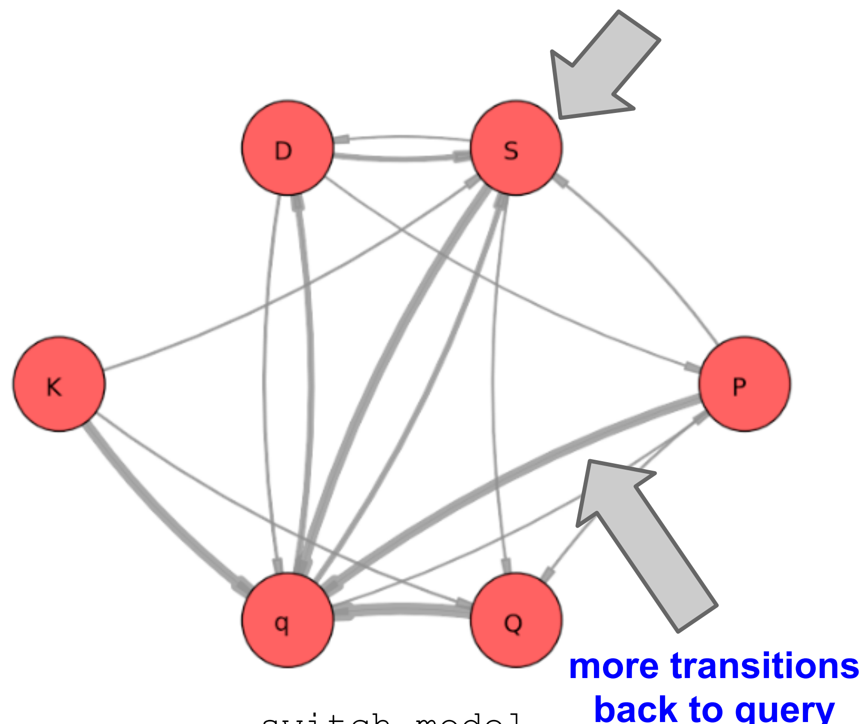
- Sequence of user's action in a session
  - type-I: **Q**=query; **C**=click; **E**=end of session
  - type-II:
    - **q/K/Q**=query with short/medium/long pause before next action;
    - **D/P/S**=click with short/medium/long dwell time;
    - **E**=end of session
- Markov model for switching detection

  [A.Hassan et al, SIGIR 2012]

Search trails Markov model



non-switch model



switch model

Session with switchings

- contain less transitions to SAT click state
- more transitions back to query

General VS. Individual Markov Model

Table 2: Markov model transition probabilities

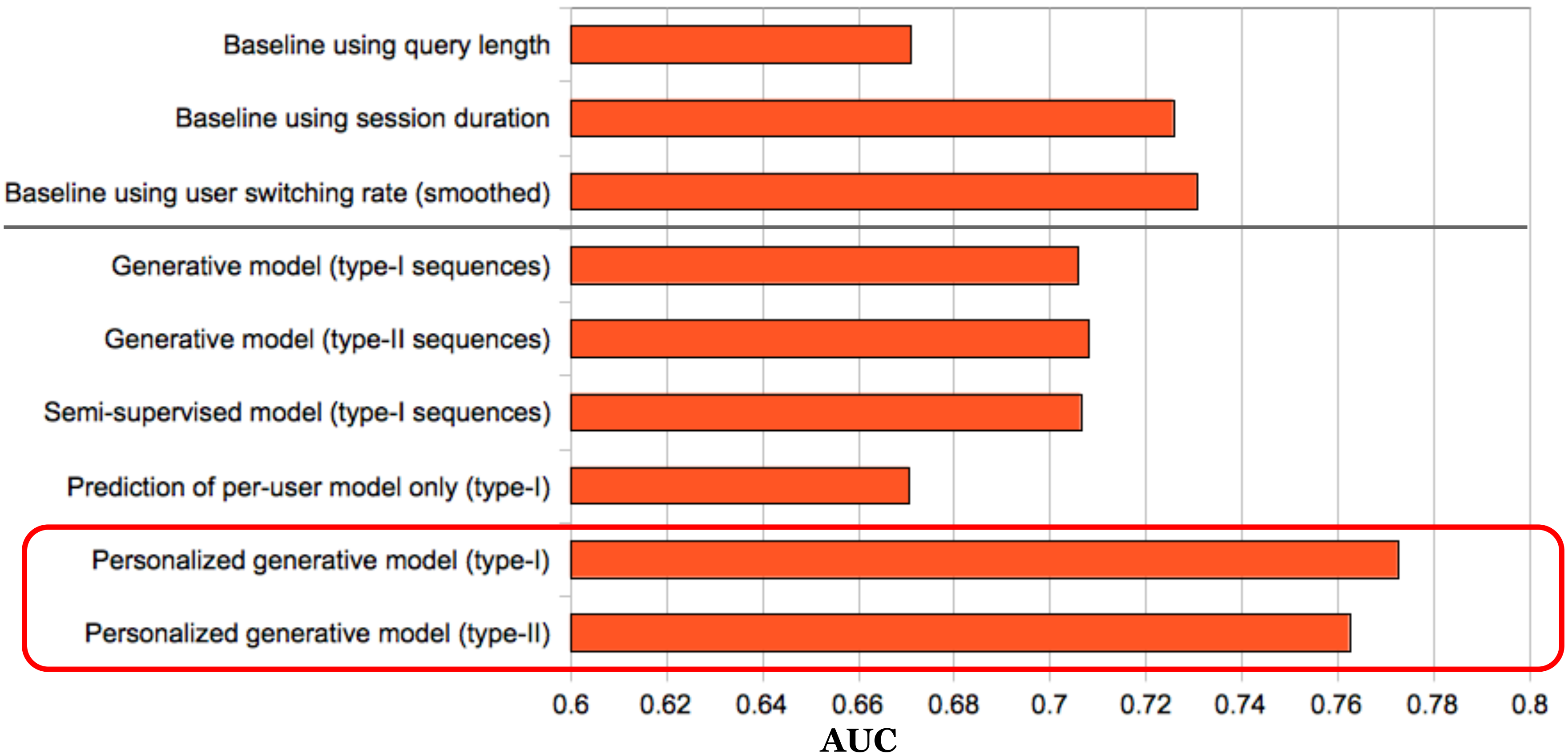
	Non-switch sessions				Switch sessions		
	Q	C	E		Q	C	E
Q	0.217	$\boxed{0.750}$	0.033	Q	0.316	$\boxed{0.621}$	0.063
C	0.207	0.470	$\boxed{0.323}$	C	0.288	0.521	$\boxed{0.191}$

Table 3: Markov model transition probabilities for a particular user

	Non-switch sessions				Switch sessions		
	Q	C	E		Q	C	E
Q	0.417	0.527	0.056	Q	0.506	0.430	0.065
C	0.219	0.352	0.428	C	0.397	0.354	0.249

- Model built for particular user can differ from aggregated model
- But: Most users have little or no history
- We use combination between general and personalized model

Performance of Personalized Markov Model



Personalized markov models significantly improves performance of the generative model for switching detection.

Machine Learning Approach to Switching Detection

- Machine learning approach was shown to be useful for switching detection

- We tried 3 personalization approaches:
  a. build a model for each user and use personalized model prediction as a feature

  b. add user ids to the feature set

  c. add personalized user statistics as a feature set

Types of features

1. <u>Session features</u>
   a. session duration, number of queries, number of clicks, average dwell time of click, last action, maximum pause between actions, etc.

2. <u>Statistics-based features</u>
   a. average values of all features described above in switch and non-switch sessions separately
   b. use these averages for normalization
   c. session duration divided by the average duration of switch sessions

3. <u>Personalized statistics-based features</u>
   a. average values of session features for each user in switch and non-switch sessions
   b. use them separately as well as for normalization

Results: Personalized Statistics Improves Prediction Performance

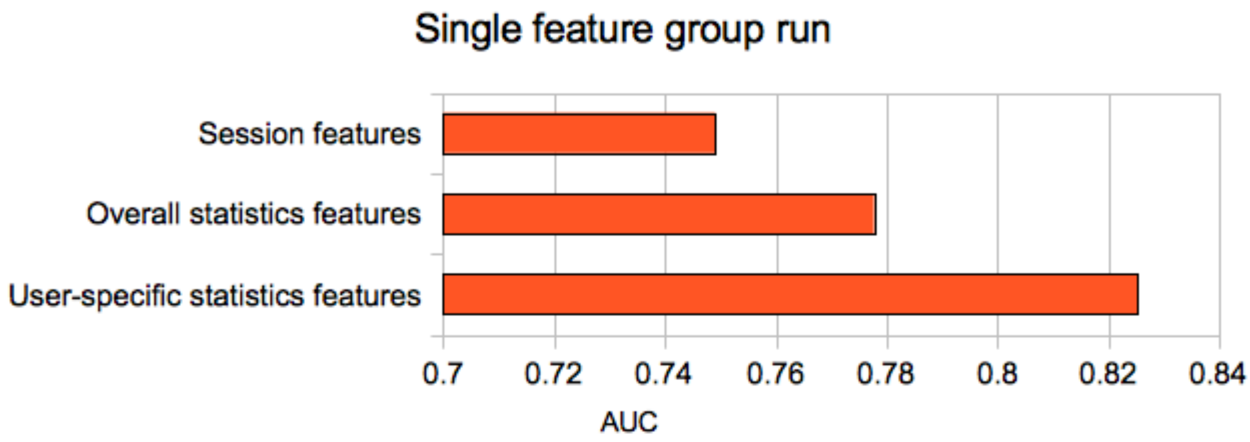
- Per-user models and model with user-ids as features are prone to <u>overfitting</u>
- Using <u>per-user</u> aggregated <u>statistics</u> significantly <u>improves detection performance</u>

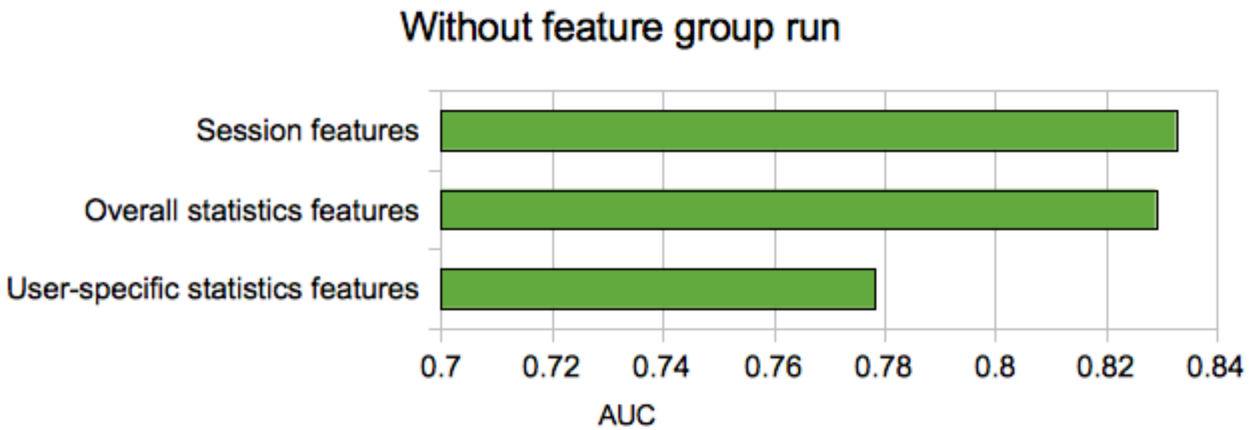
Best Performing Features (Gini index)

Rank	Feature
1	probability of switch under 3-gram model
2	total number of switches for a given user
3	average click position
4	user switching rate (smoothed)
13	time to first click in a session

Takeway: Features based on users statistics are among the top by importance

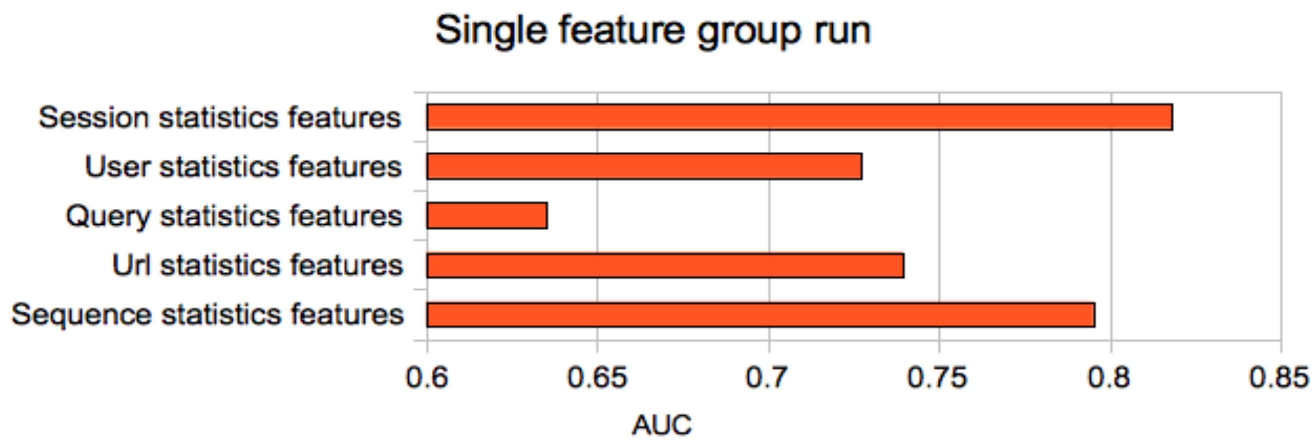
Feature Ablation Experiments

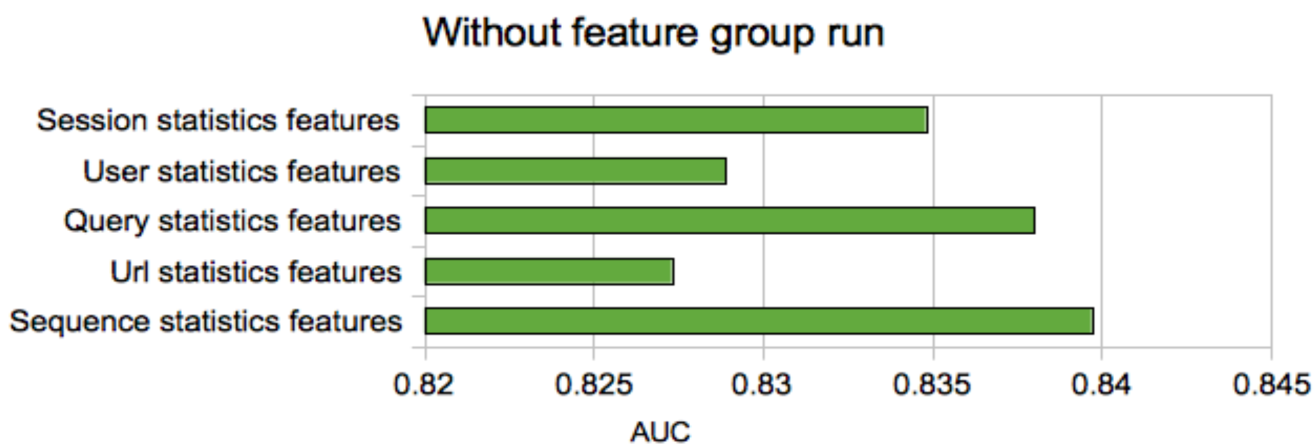




<u>Takeway</u>: User statistics-based features are the most important.

Feature importance: another perspective





- Session statistics and search trails features are 2 most useful groups
- url statistics are more useful than query statistics (urls triggering switching behavior?)

Performance boosted by personalization

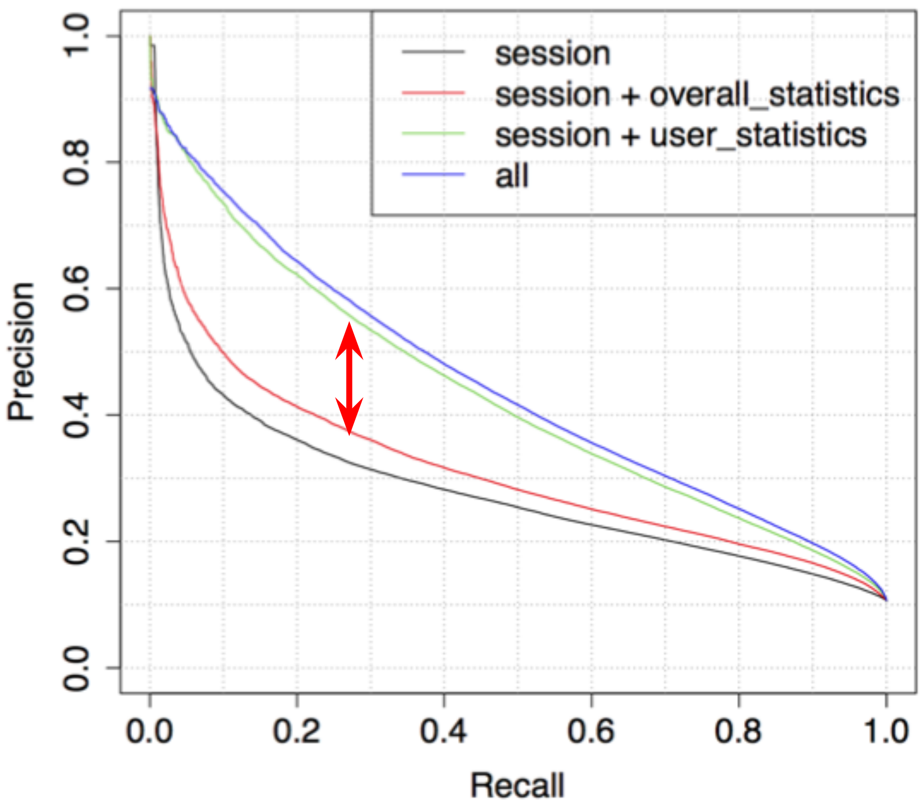


Figure 5: Precision-Recall curve for the positive class (switch sessions)

How much is enough?

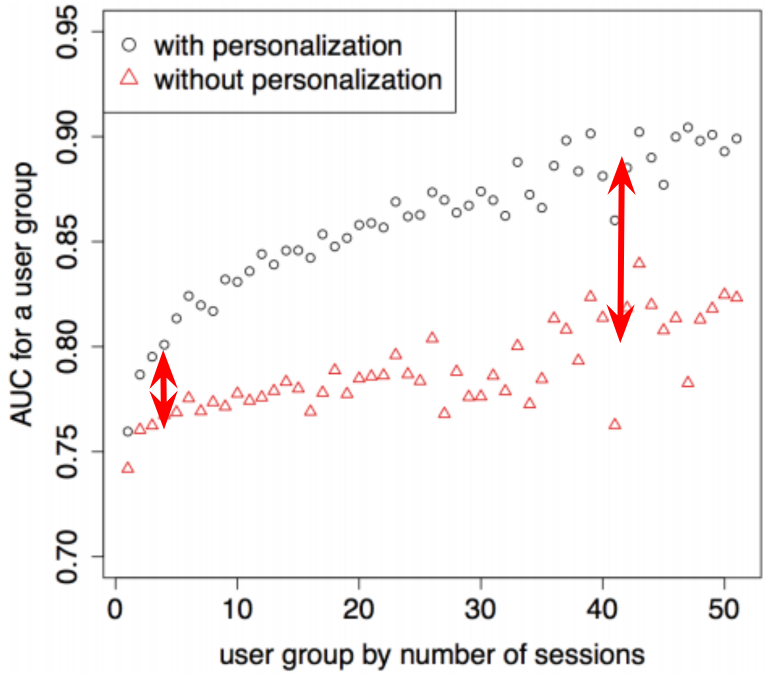


Figure 4: AUC for users with different size of search history (number of sessions)

Even for user with history as small as ~5 sessions user statistics based features improves switching detection performance.

Model comparison

Model	AUC
Baseline: # queries	0.6710
Baseline: session duration	0.7257
<u>Baseline: user switching rate</u>	<u>0.7306</u>
Semi-supervised model from [A.Hassan at al, 2012]	0.7081
Personalized generative model	0.7725
Online prediction model trained on subset of features from [R.White et al. 09]	0.7206
<u>Our model</u>	**<u>0.8450</u>**

Conclusion

- We showed that utilizing individual user behavior models drastically improves switching detection performance

- Described personalized model won 1st place in Yandex Switching Detection Challenge

  code: `http://mathcs.emory.edu/~dsavenk/switch_detect`

- We believe the same strategy has potential to be useful for other log analysis tasks, such as relevance prediction, satisfaction prediction, etc.

Thank You!

Happy Switching

Questions?