Learning from User Interactions through Interventions

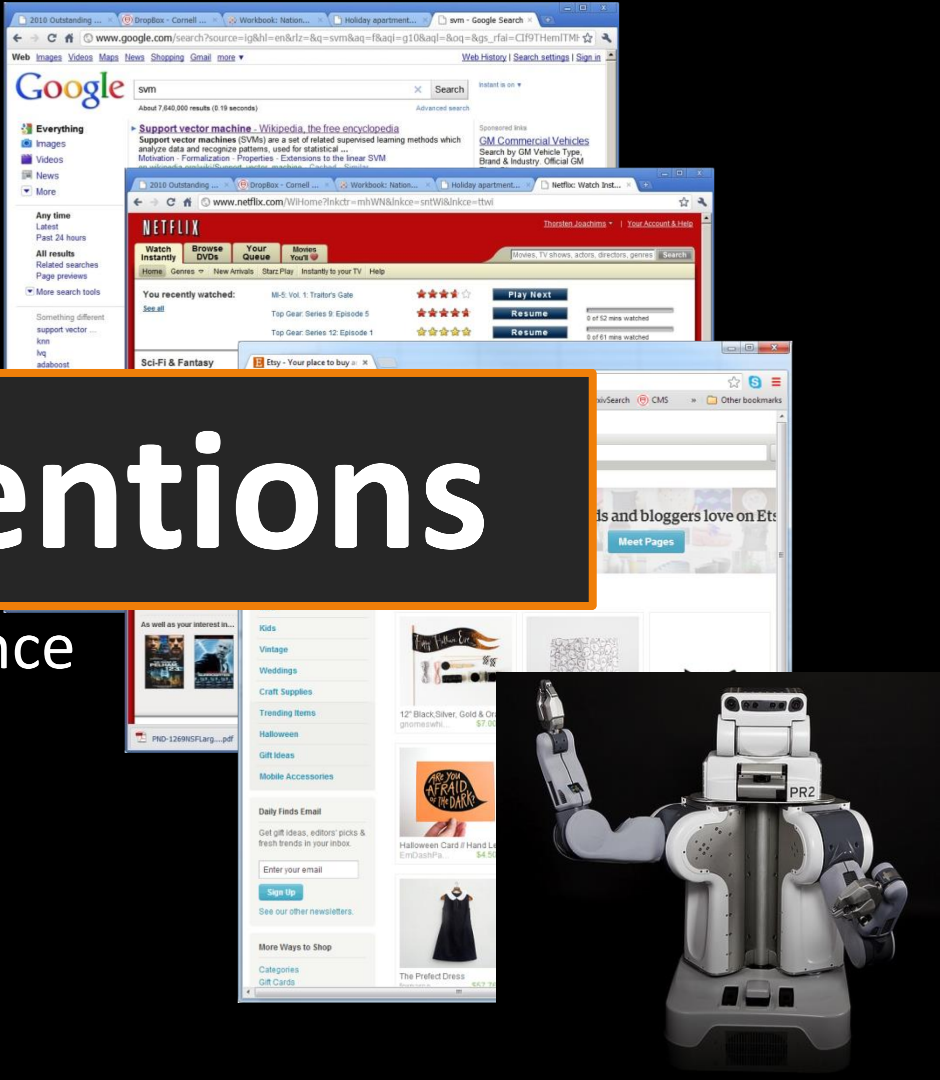
Josef Broder, Olivier Chapelle, Geri Gay, Arpita Ghosh, Laura Granka, Thorsten Joachims, Bobby Kleinberg, Madhu Kurup, Filip Radlinski, Karthik Raman, Tobias Schnabel, Pannaga Shivaswamy, Adith Swaminathan, Yisong Yue

Department of Computer Science
Department of Information Science
Cornell University

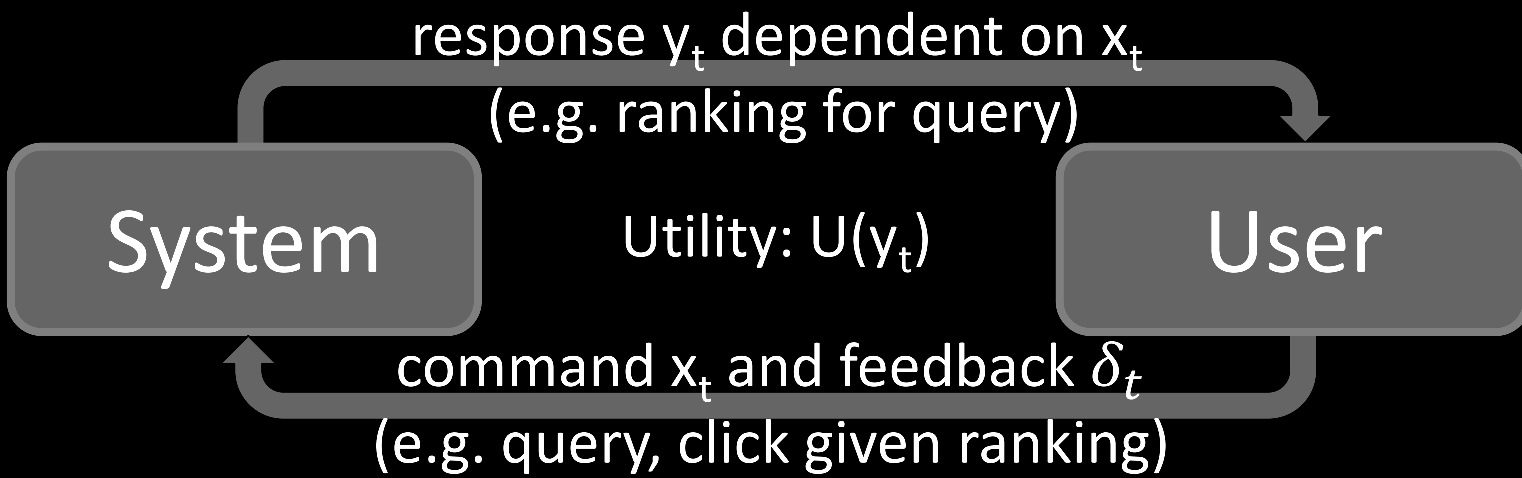
Interactive Learning Systems

- Examples
  - Search engines
  - Entertainment media
  - E-co
  - Sma
- Learni
  - Gathering and maintenance of knowledge
  - Measure and optimize performance
  - Personalization

Interventions



Interactive Learning System



- Designing Information Elicitation Interventions
- Online Learning with Interventions
- Offline Learning with Logged Intervention Data

Decide between two Ranking Functions

Distribution P(x) of x=(user, query)

⋮
(tj,"SVM")
⋮

Retrieval Function 1
$f_1(x) \rightarrow y_1$

Which one is better?

Retrieval Function 2
$f_2(x) \rightarrow y_2$

1. Kernel Machines
   http://svm.first.gmd.de/
2. SVM-Light Support Vector Machine
   http://svmlight.joachims.org/
3. School of Veterinary Medicine at UPenn
   http://www.vet.upenn.edu/
4. An Introduction to Support Vector Machines
   http://www.support-vector.net/
5. Service Master Company
   http://www.servicemaster.com/

⋮

$U(tj,\text{"SVM"},y_1)$

1. School of Veterinary Medicine at UPenn
   http://www.vet.upenn.edu/
2. Service Master Company
   http://www.servicemaster.com/
3. Support Vector Machine
   http://jbolivar.freeservers.com/
4. Archives of SUPPORT-VECTOR-MACHINES
   http://www.jiscmail.ac.uk/lists/SUPPORT...
5. SVM-Light Support Vector Machine
   http://ais.gmd.de/~thorsten/svm light/

⋮

$U(tj,\text{"SVM"},y_2)$

Measuring Utility

Name	Description	Aggre-gation	Hypothesized Change with Decreased Quality
Abandonment Rate	% of queries with no click	N/A	Increase
Reformulation Rate	% of queries that are followed by reformulation	N/A	Increase
Queries per Session	Session = no interruption of more than 30 minutes	Mean	Increase
Clicks per Query	Number of clicks	Mean	Decrease
Click@1	% of queries with clicks at position 1	N/A	Decrease
Max Reciprocal Rank*	1/rank for highest click	Mean	Decrease
Mean Reciprocal Rank*	Mean of 1/rank for all clicks	Mean	Decrease
Time to First Click*	Seconds before first click	Median	Increase
Time to Last Click*	Seconds before final click	Median	Decrease

(*) only queries with at least one click count

ArXiv.org: User Study

User Study in ArXiv.org

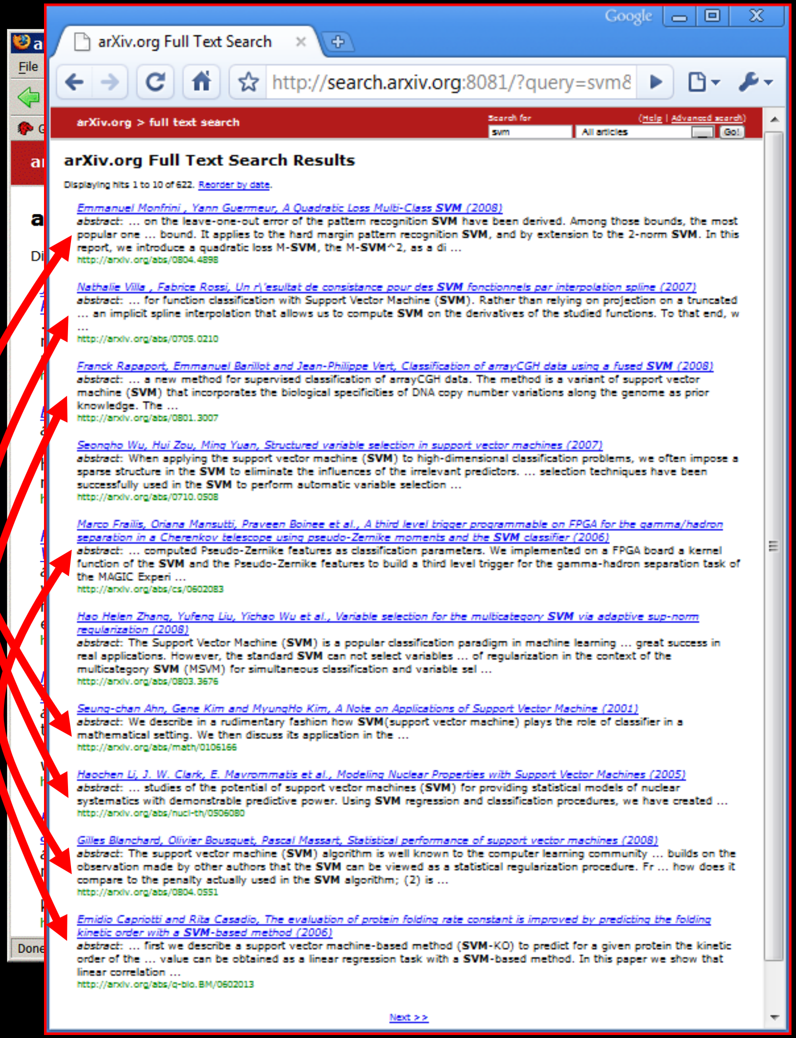
– Natural user and query population
– User in natural context, not lab
– Live and operational search engine
– Ground truth by construction

Orig $\succ$ Swap2 $\succ$ Swap4

- Orig: Hand-tuned fielded
- Swap2: Orig with 2 pairs swapped
- Swap4: Orig with 4 pairs swapped

Orig $\succ$ Flat $\succ$ Rand

- Orig: Hand-tuned fielded
- Flat: No field weights
- Rand : Top 10 of Flat shuffled

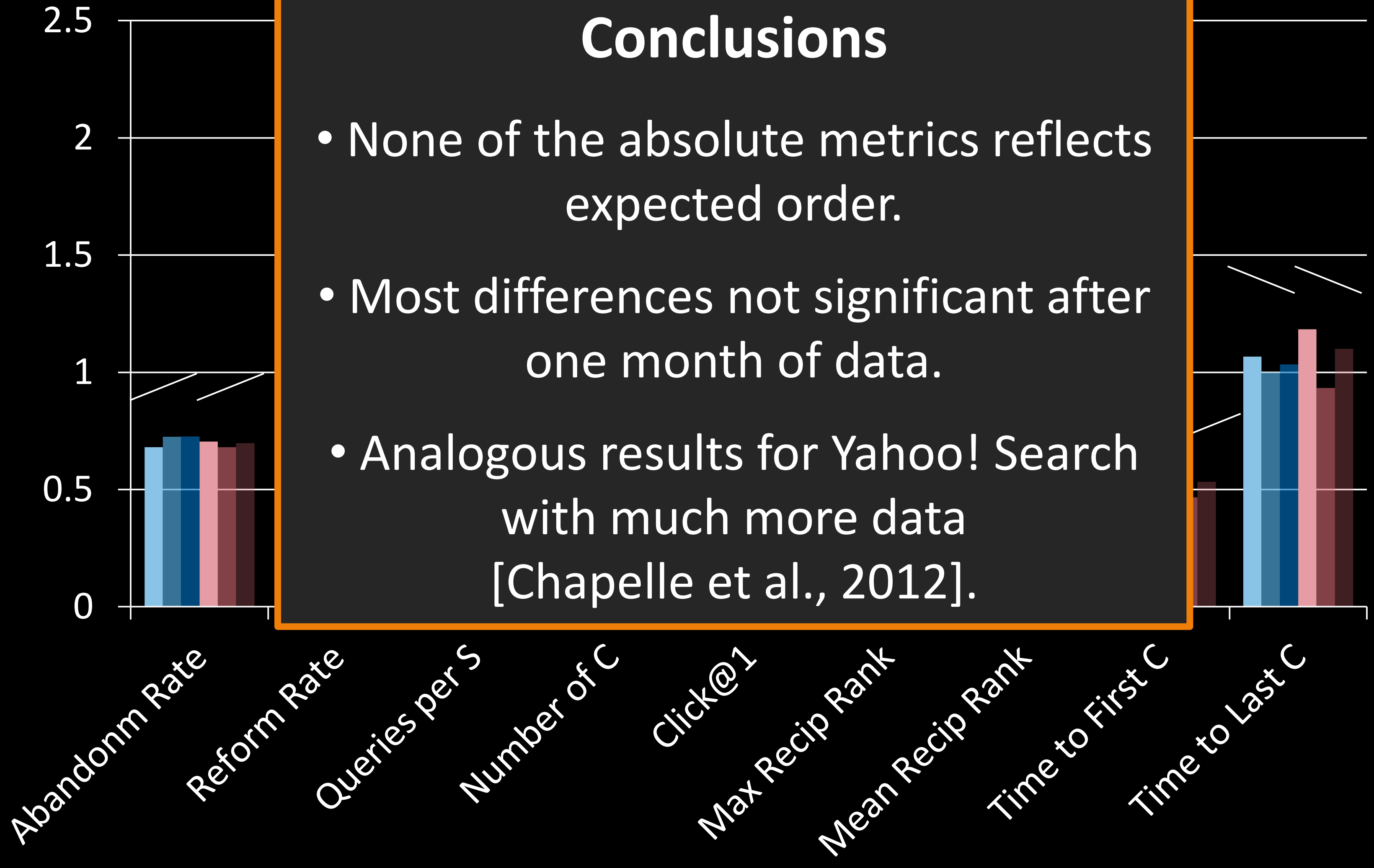


[Radlinski et al., 2008]

ArXiv.org: Experiment Setup

- Experiment Setup
  - Phase I: 36 days
    - Users randomly receive ranking from Orig, Flat, Rand
  - Phase II: 30 days
    - Users randomly receive ranking from Orig, Swap2, Swap4
  - User are permanently assigned to one experimental condition based on IP address and browser.
- Basic Statistics
  - ~700 queries per day / ~300 distinct users per day
- Quality Control and Data Cleaning
  - Test run for 32 days
  - Heuristics to identify bots and spammers
  - All evaluation code was written twice and cross-validated

Arxiv.org: Results



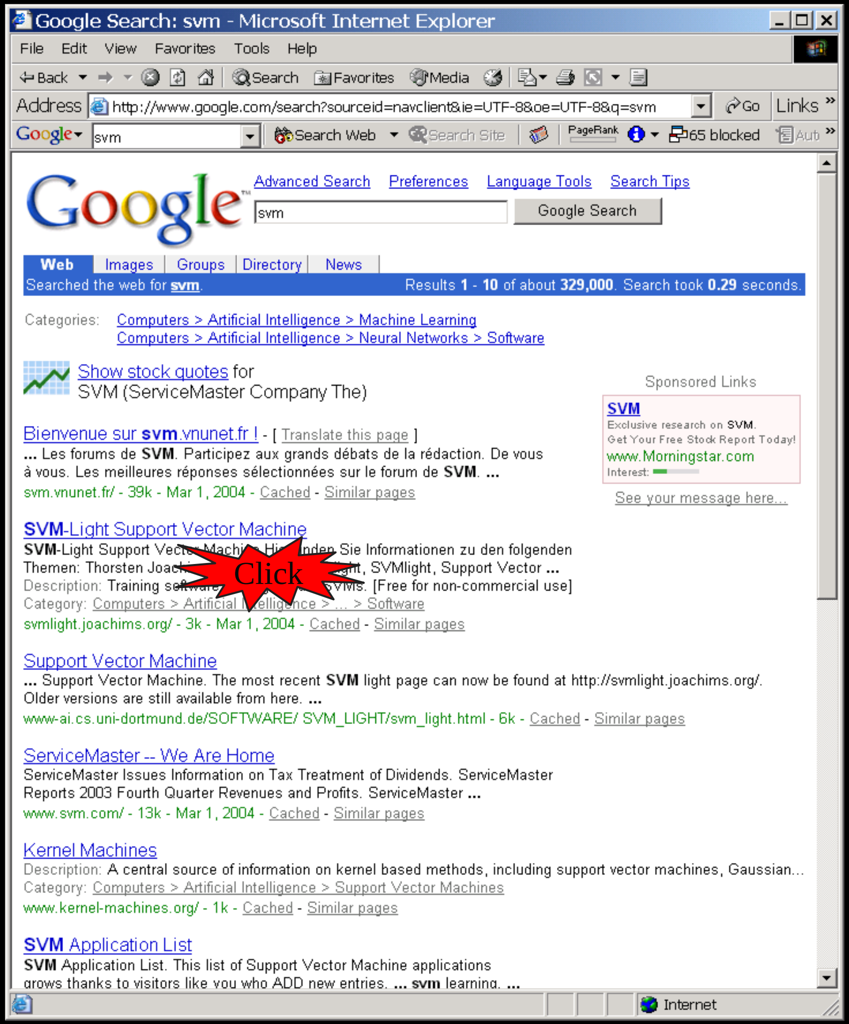
Conclusions

- None of the absolute metrics reflects expected order.
- Most differences not significant after one month of data.
- Analogous results for Yahoo! Search with much more data [Chapelle et al., 2012].

[Radlinski et al., 2008]

Economic Models of Decision Making

- Rational Choice
  - Alternatives $\mathcal{Y}$
  - Utility function U(y)
  - Decision $y^*=\mathrm{argmax}_{y\in\mathcal{Y}}\{U(y)\}$
- Bounded Rationality
  - Time constraints
  - Computation constraints
  - Approximate U(y)
- Behavioral Economics
  - Framing
  - Fairness
  - Loss aversion
  - Handling uncertainty



A Model of how Users Click in Search

- Model of clicking:
  - Users explore ranking to position k
  - Users click on most relevant (looking) links in top k
  - Users stop clicking when time budget up or other action more promising (e.g. reformulation)
  - Empirically supported by [Granka et al., 2004]

Decide between two Ranking Functions

Distribution P(x) of x=(user, query)

⋮
(tj,"SVM")
⋮

Retrieval Function 1
$f_1(x) \rightarrow y_1$

Which one is better?

Retrieval Function 2
$f_2(x) \rightarrow y_2$

1. Kernel Machines
   http://svm.first.gmd.de/
2. SVM-Light Support Vector Machine
   http://svmlight.joachims.org/
3. School of Veterinary Medicine at UPenn
   http://www.vet.upenn.edu/
4. An Introduction to Support Vector Machines
   http://www.support-vector.net/
5. Service Master Company
   http://www.servicemaster.com/

⋮

U(tj,"SVM",y_1)

1. School of Veterinary Medicine at UPenn
   http://www.vet.upenn.edu/
2. Service Master Company
   http://www.servicemaster.com/
3. Support Vector Machine
   http://jbolivar.freeservers.com/
4. Archives of SUPPORT-VECTOR-MACHINES
   http://www.jiscmail.ac.uk/lists/SUPPORT...
5. SVM-Light Support Vector Machine
   http://ais.gmd.de/~thorsten/svm light/

⋮

U(tj,"SVM",y_2)

Balanced Interleaving

x=(u=tj, q="svm")

$f_1(x) \rightarrow y_1$

1. Kernel Machines
   http://svm.first.gmd.de/
2. Support Vector Machine
   http://jbolivar.freeservers.com/
3. An Introduction to Support Vector Machines
   http://www.support-vector.net/
4. Archives of SUPPORT-VECTOR-MACHINES ...
   http://www.jiscmail.ac.uk/lists/SUPPORT...
5. SVM-Light Support Vector Machine
   http://ais.gmd.de/~thorsten/svm light/

$f_2(x) \rightarrow y_2$

1. Kernel Machines
   http://svm.first.gmd.de/
2. SVM-Light Support Vector Machine
   http://ais.gmd.de/~thorsten/svm light/
3. Support Vector Machine and Kernel ... References
   *http://svm.*research.bell-labs.com/SVMrefs.html
4. Lucent Technologies: SVM demo applet
   *http://svm.*research.bell-labs.com/SVT/SVMsvt.html
5. Royal Holloway Support Vector Machine
   http://svm.dcs.rhbnc.ac.uk

Interleaving(y_1,y_2)

1.	Kernel Machines *http://svm.first.gmd.de/*	**1**
2.	Support Vector Machine *http://jbolivar.freeservers.com/*	**2**
3.	SVM-Light Support Vector Machine *http://ais.gmd.de/~thorsten/svm light/*	**2**
4.	An Introduction to Support Vector Machines *http://www.support-vector.net/*	**3**
5.	Support Vector Machine and Kernel ... References *http://svm.*research.bell-labs.com/SVMrefs.html	**3**
6.	Archives of SUPPORT-VECTOR-MACHINES ... *http://www.jiscmail.ac.uk/lists/SUPPORT...*	**4**
7.	Lucent Technologies: SVM demo applet *http://svm.*research.bell-labs.com/SVT/SVMsvt.html	**4**

Model of User:
Better retrieval functions is more likely to get more clicks.

Invariant:
For all k, top k of balanced interleaving is union of top k_1 of r_1 and top k_2 of r_2 with $k_1=k_2 \pm 1$.

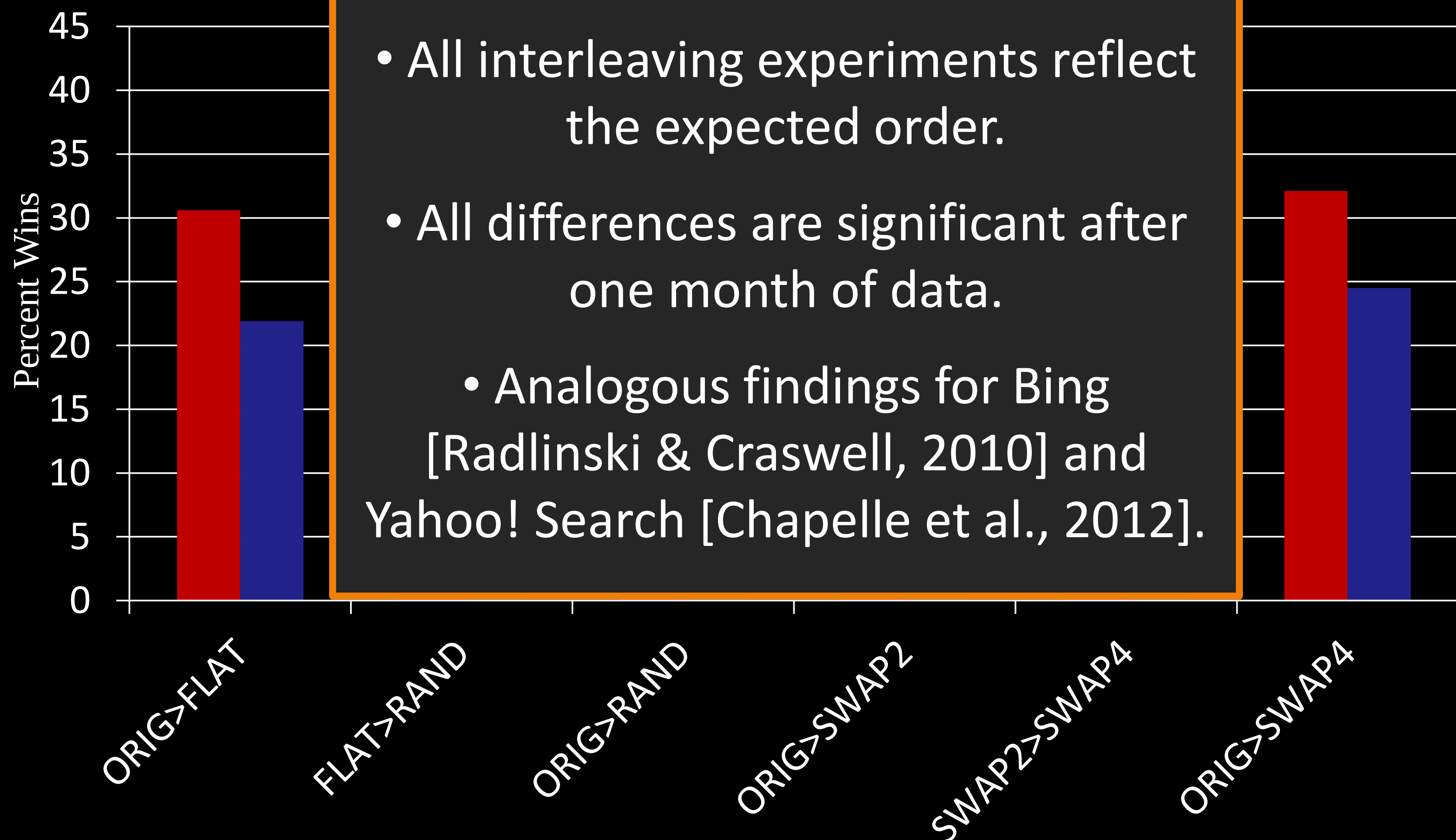
Interpretation: $(y_1 \succ y_2) \leftrightarrow$ clicks(topk(y_1)) > clicks(topk(y_2))
→ see also [Radlinski, Craswell, 2012] [Hofmann, 2012]

[Joachims, 2001] [Radlinski et al., 2008]

Arxiv.org: Interleaving Experiment

- Experiment Setup
  - Phase I: 36 days
    - Balanced Interleaving of (Orig,Flat) (Flat,Rand) (Orig,Rand)
  - Phase II: 30 days
    - Balanced Interleaving of (Orig,Swap2) (Swap2,Swap4) (Orig,Swap4)
- Quality Control and Data Cleaning
  - Same as for absolute metrics

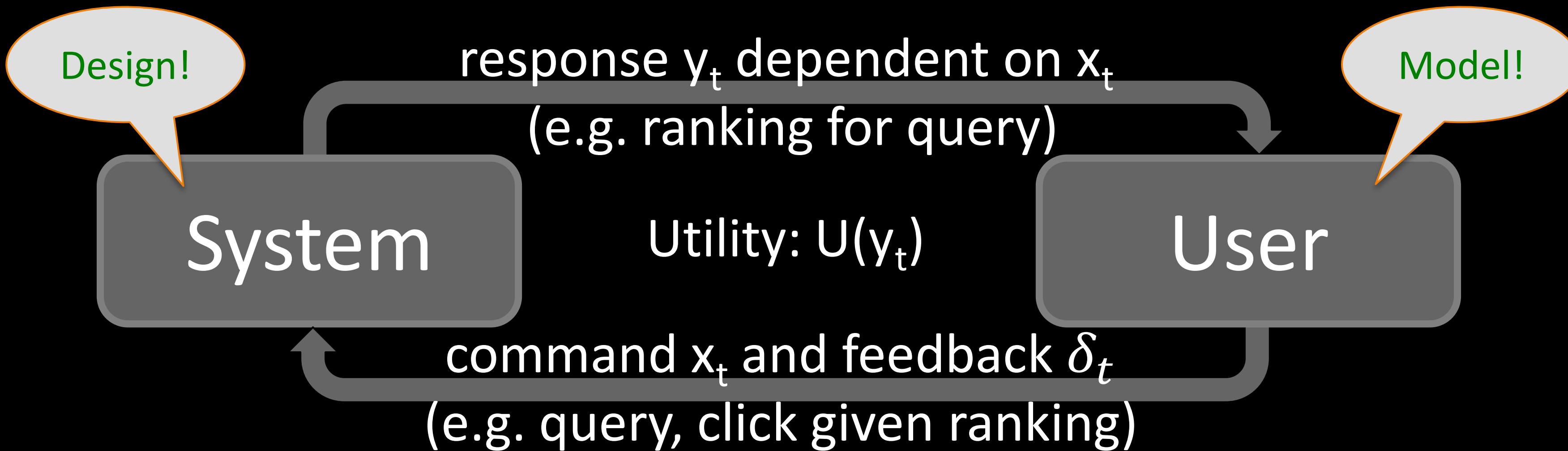
Arxiv.org: Interleaving Results



Conclusions

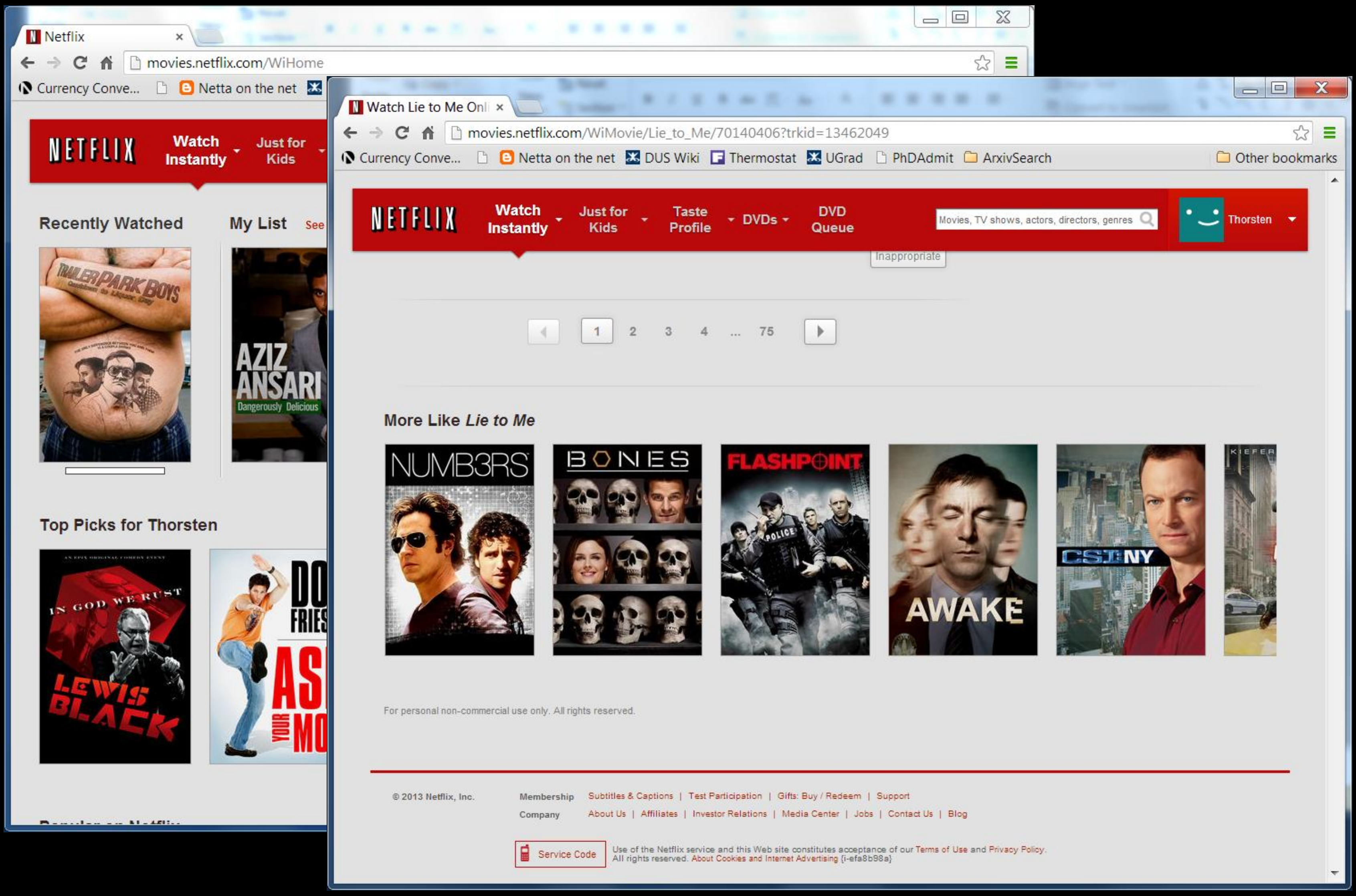
- All interleaving experiments reflect the expected order.
- All differences are significant after one month of data.
- Analogous findings for Bing [Radlinski & Craswell, 2010] and Yahoo! Search [Chapelle et al., 2012].

Interactive Learning System



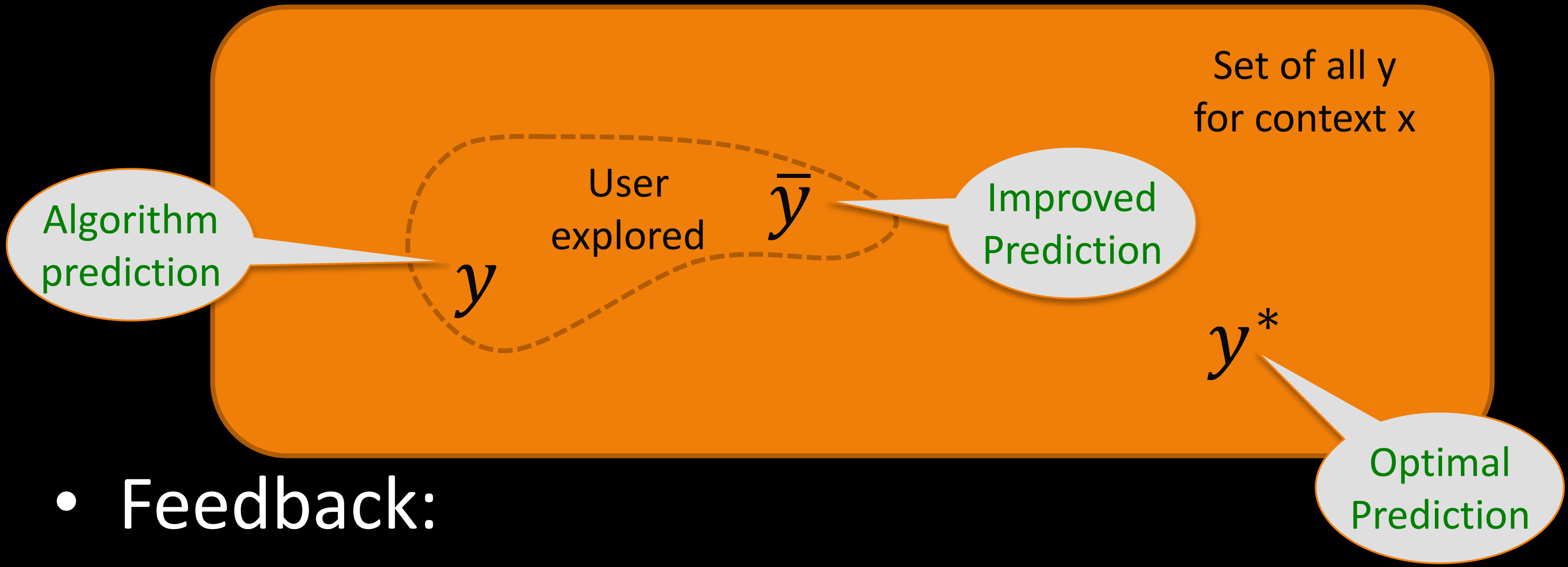
- Designing Information Elicitation Interventions
  - Model user's decision process → derive intervention design
- Online Learning with Interventions
- Offline Learning with Logged Intervention Data

Coactive Exploration Example 1



Coactive Feedback Model

- Intervention: prediction y and browsing network



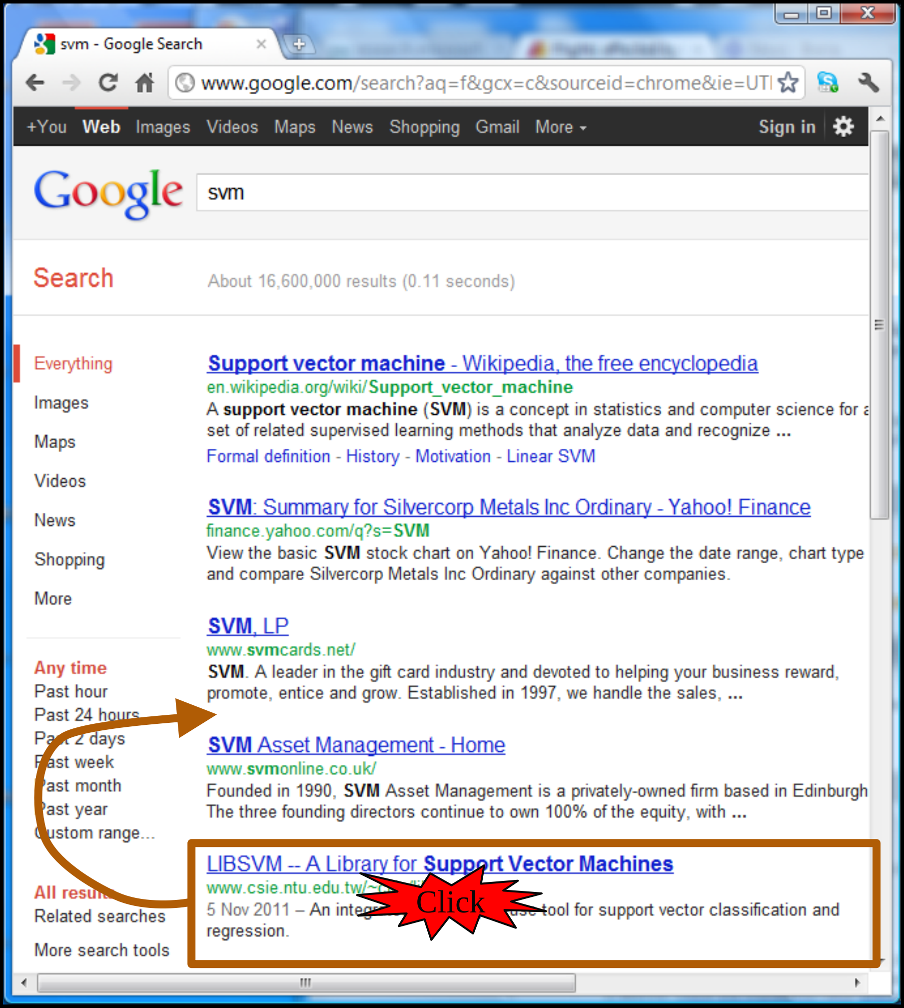
- Feedback:
  - Improved prediction $\bar{y}_t$

$$U(\bar{y}_t|x_t) > U(y_t|x_t)$$

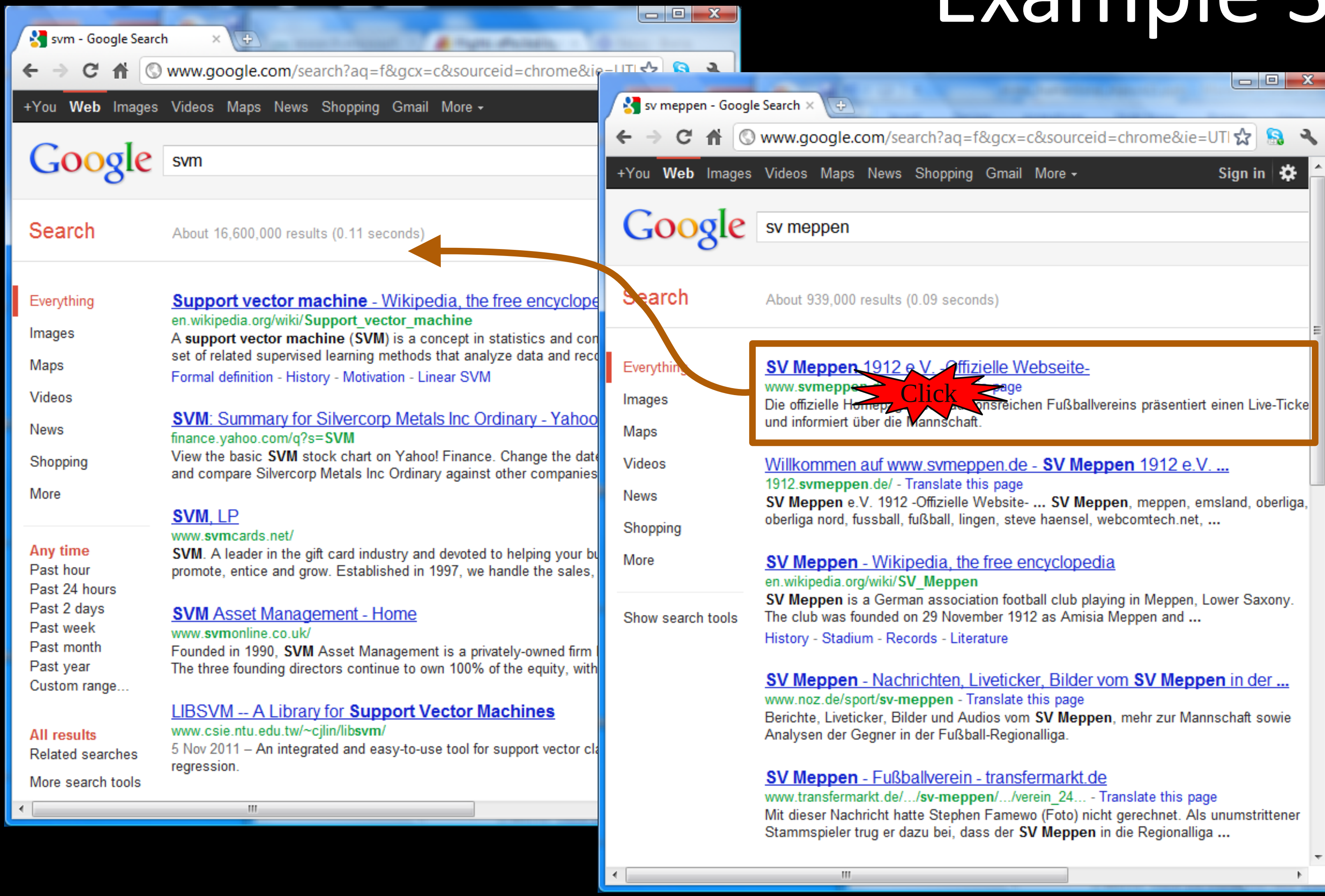
  - Supervised learning: optimal prediction y_t^*

$$y_t^* = \mathrm{argmax}_y\, U(y|x_t)$$

Coactive Exploration Example 2



Coactive Exploration Example 3



Coactive Exploration
Machine Translation

x_t

We propose Coactive Learning as a model of interaction between a learning system and a human user, where both have the common goal of providing results of maximum utility to the user.

y_t

Wir schlagen vor, koaktive Learning als ein Modell der Wechselwirkung zwischen einem Lernsystem und menschlichen Benutzer, wobei sowohl die gemeinsame Ziel, die Ergebnisse der maximalen Nutzen für den Benutzer.

$\prec$

$\bar{y}_t$

Wir schlagen ~~vor,~~ koaktive Learning als ein Modell ~~der Wechselwirkung~~ des Dialogs zwischen einem Lernsystem und menschlichen Benutzer, wobei ~~sowohl die~~ beide das gemeinsame Ziel haben, die Ergebnisse der maximalen Nutzen für den Benutzer zu liefern.

Coactive Preference Perceptron

- Model
  - Linear model of user utility: $U(y|x) = w^T \phi(x,y)$
- Algorithm
  - FOR t = 1 TO T DO
    - Observe x_t
    - Present $y_t = \text{argmax}_y \{ w_t^T \phi(x_t,y) \}$
    - Obtain feedback $\bar{y}_{\mathbf{t}}$ from user
    - Update $w_{t+1} = w_t + \phi(x_t,\bar{y}_{\mathbf{t}}) - \phi(x_{\mathbf{t}},y_{\mathbf{t}})$
- This may look similar to a multi-class Perceptron, but
  - Feedback $\bar{y}_t$ is different (not get the correct class label)
  - Regret is different (misclassifications vs. utility difference)

$$\mathrm{R(A)} = \frac{1}{T}\sum_{t=1}^{T}[U(y_t^*|x) - U(y_t|x)]$$

Never revealed:
- cardinal feedback
- optimal y^*

[Shivaswamy, Joachims, 2012]

Coactive Perceptron: Regret Bound

- Model

  U(**y**|**x**) = $\mathbf{w}^T$ ϕ(**x**,**y**), where w is unknown

- Feedback: ξ-Approximately α-Informative

$$E[U(x_t,\bar{y}_t)] \geq U(x_t,y_t) + \alpha\big(U(x_t,y_t^*) - U(x_t,y_t)\big) - \xi_t$$

user feedback

system prediction

gap to optimal

model error

- Theorem

  For user feedback **ȳ** that is α-informative in expectation, the expected average regret of the Preference Perceptron is bounded by

$$E\left[\frac{1}{T}\sum_{t=1}^{T} U(y_t^*|x) \; - U(y_t|x)\right] \leq \frac{1}{\alpha T}\sum_{t=1}^{T}\xi_t + \frac{2R||w||}{\alpha\sqrt{T}}$$

→ zero

model error

[Shivaswamy, Joachims, 2012]

Preference Perceptron: Experiment

Experiment:

- Automatically optimize Arxiv.org Fulltext Search

Model

- Utility of ranking y for query x: $U_t(y|x) = \sum_i \gamma_i \, w_t^T \phi(x,y^{(i)})$ [~1000 features]

  →Computing argmax ranking: sort by $w_t^T \phi(x,y^{(i)})$

Analogous to DCG

Feedback

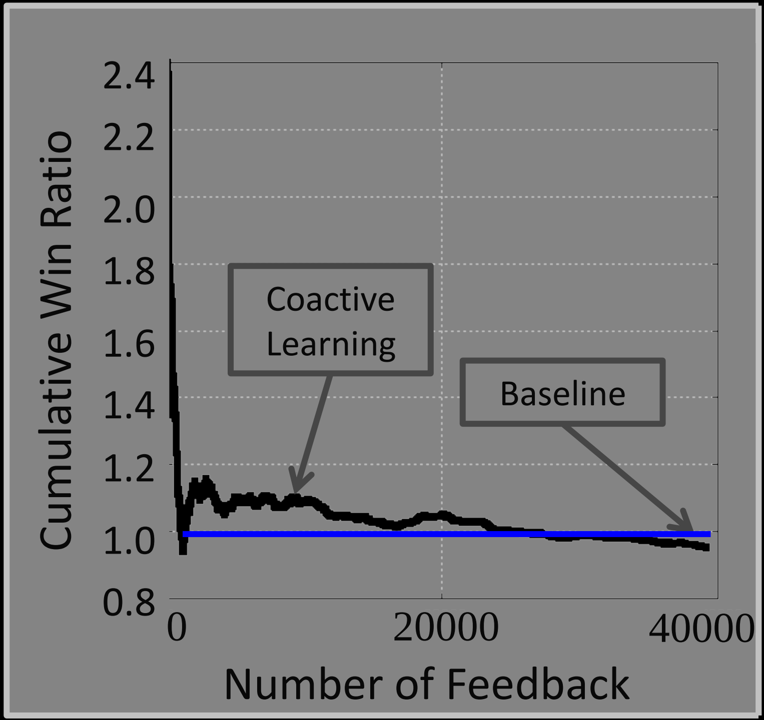
- Construct $\bar{y}_t$ from y_t by moving clicked links one position higher.

Baseline

- Handtuned w_{base} for $U_{base}(y|x)$

Evaluation

- Interleaving of ranking from $U_t(y|x)$ and $U_{base}(y|x)$



[Raman et al., 2013]

Why did it fail?

- Assume

  $$U_t(y|x) = U(y|x)$$

- Prediction

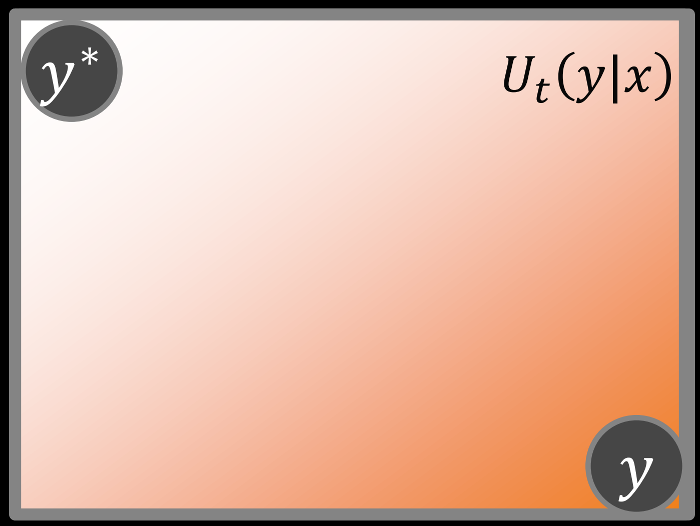
  $$y = y^* = \text{argmax}_y\, U_t(y|x)$$

- Feedback quality

  $$E[U(x,\bar{y})] \geq U(x,y) + \alpha\big(U(x,y^*) - U(x,y)\big) - \xi$$
  $$= U(x,y^*) - \xi$$

  → any presence of click noise implies $\xi > 0$

  → biased gradient



Optimizing the User Feedback

- Assume

  $U_t(y|x) = U(y|x)$

- Prediction

  $y = y^* = \text{argmax}_y\, U_t(y|x)$

- Intervention

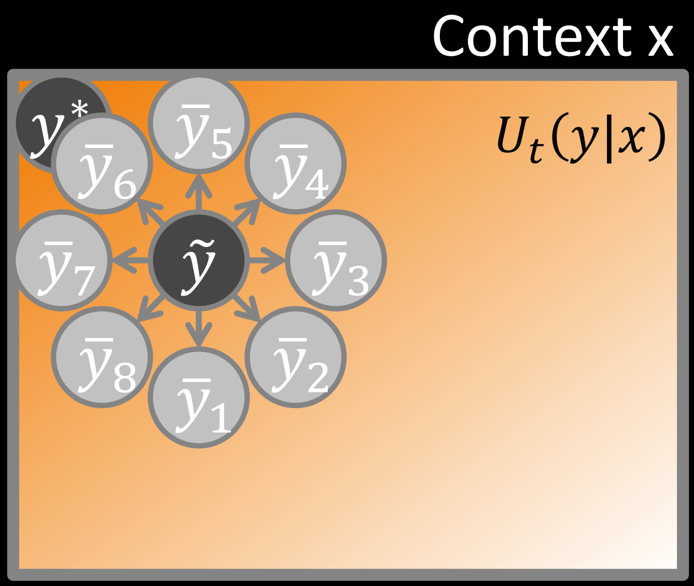
  Present $\tilde{y} = Perturb(y)$

- Feedback quality

  $E[U(x,\bar{y})] \geq U(x,\tilde{y}) + \alpha\big(U(x,y^*) - U(x,\tilde{y})\big) - \xi$

  → $\xi = 0$ (or small)

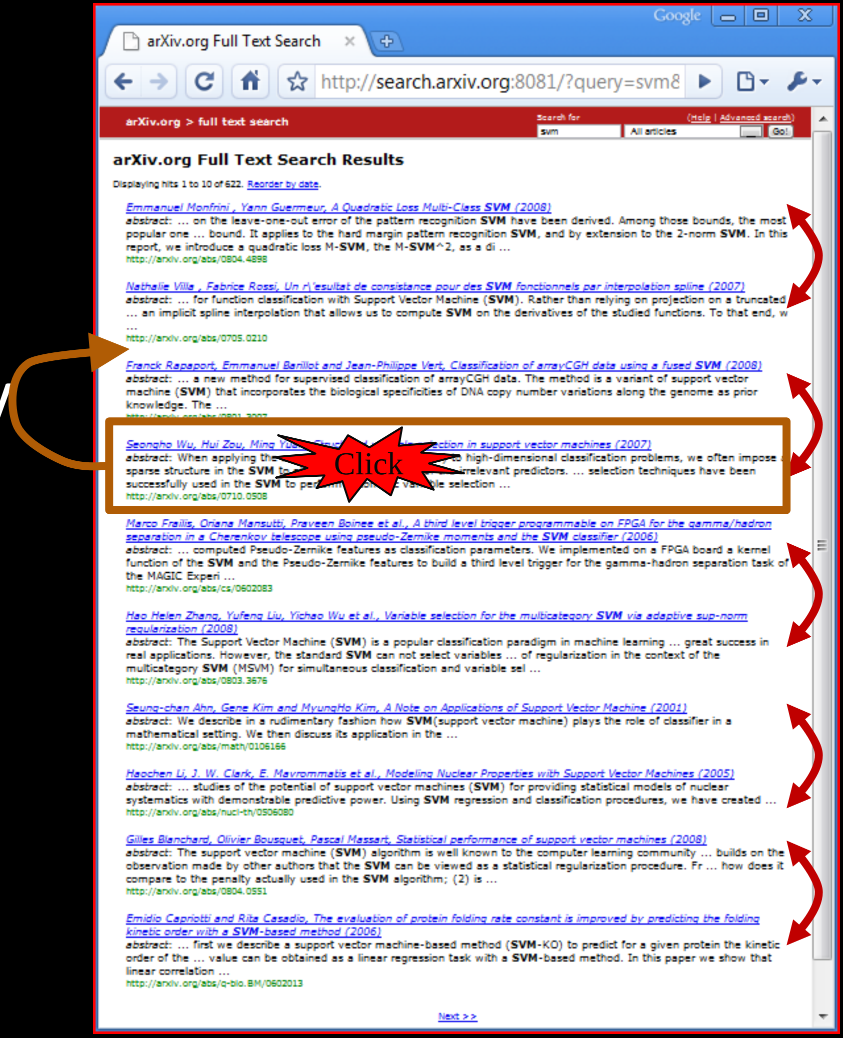
  → unbiased gradient at cost $U(x,y^*) - U(x,\tilde{y})$



FairPair Perturbation

- Idea
  – Perturb by swapping adjacent pairs
  – Generate preferences only within pair

→ Randomizes out bias from presentation and feedback generation



[Radlinski & Joachims, 2006] [Raman et al., 2013]

Preference Perceptron: Experiment

Experiment:

- Automatically optimize Arxiv.org Fulltext Search

Model

- Utility of ranking y for query x: $U_t(y|x) = \sum_i \gamma_i \, w_t^T \phi(x,y^{(i)})$ [~1000 features]
  $\rightarrow$Computing argmax ranking: sort by $w_t^T \phi(x,y^{(i)})$

Feedback

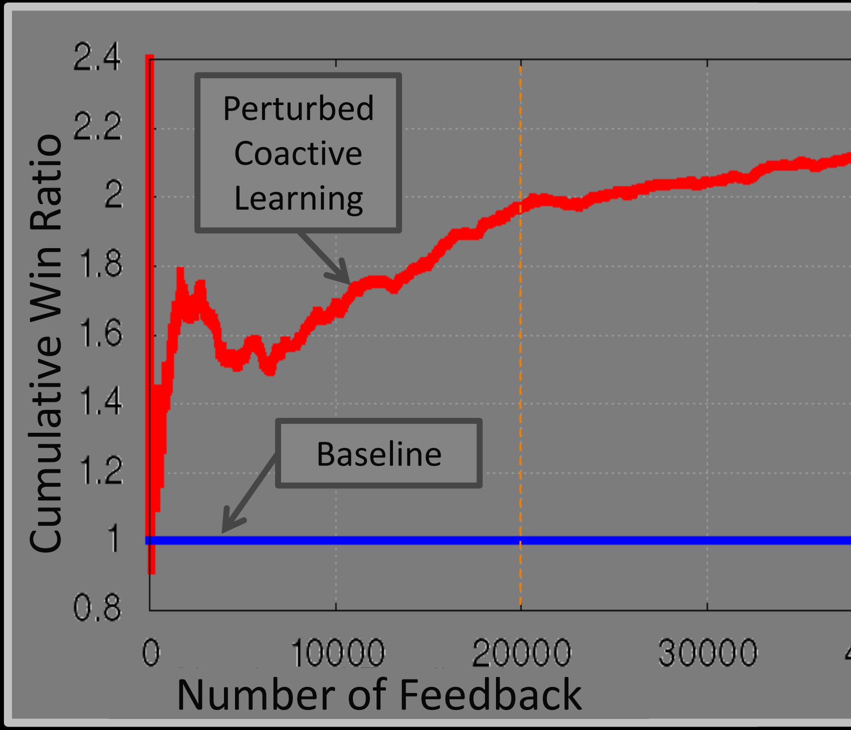
- **FairPair Perturbation**
- Construct $\bar{y}_t$ from y_t by moving clicked links one position higher.

Baseline

- Handtuned w_{base} for $U_{base}(y|x)$

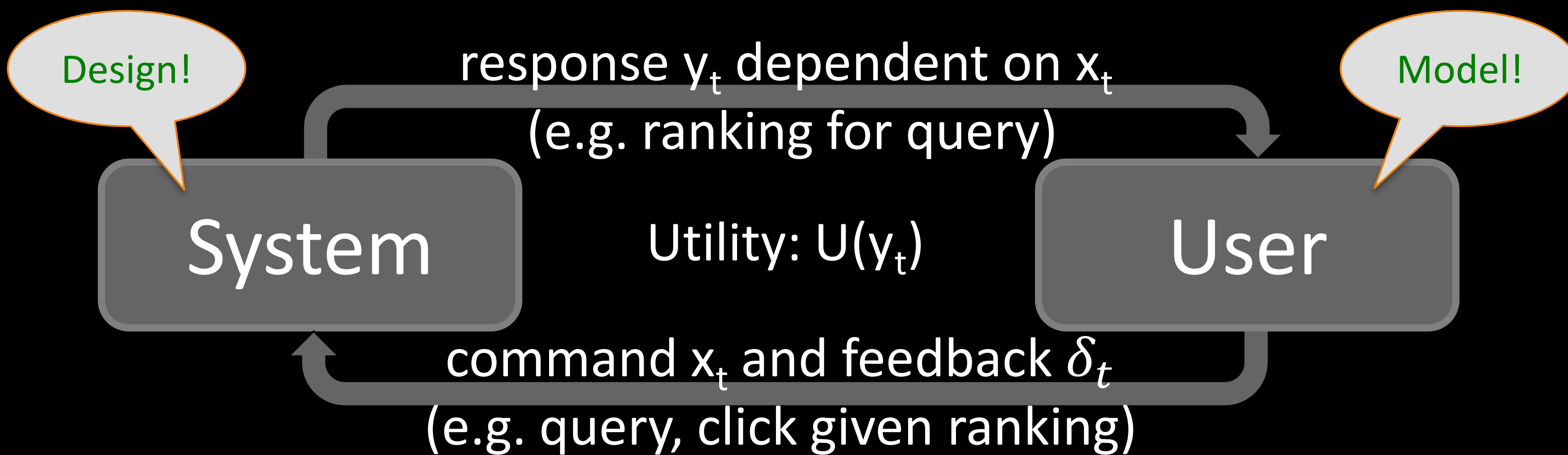
Evaluation

- Interleaving of ranking from $U_t(y|x)$ and $U_{base}(y|x)$



[Raman et al., 2013]

Interactive Learning System



- Designing Information Elicitation Interventions
  - Model user's decision process → derive intervention design
- Online Learning with Interventions
  - Design space: LearningSystem = { Algorithm } x { Interventions }
- Offline Learning with Logged Intervention Data

Information in Interaction Logs

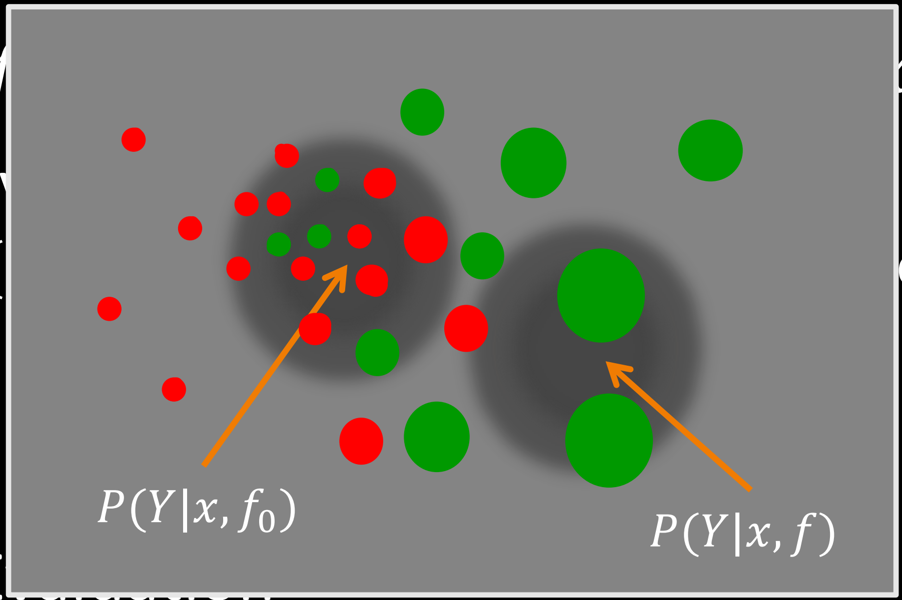
- Partial Information (aka “Bandit”) Feedback
  - Search engine f_0 interleaves ranking y for query x with baseline ranker and observes win/loss δ
  - News recommender f_0 presents set y of articles for user x and observes that user reads δ minutes
  - Ad system f_0 presents ad y to user x and observe click/no-click δ
  - MT system f_0 predicts translation y for x and receives rating δ

→ Data: $S = ((\underbrace{x_1}_{\text{context}}, \underbrace{y_1}_{f_0\text{ action}}, \underbrace{\delta_1}_{\text{reward}}), \ldots, (x_n, y_n, \delta_n))$

Changing History

- Expected Performance of Stochastic Policy f: $P(y|x, f)$

$R(f$ … $x)$



- On-Policy Ev…

  Given $S = ($ … ed under f_0,

- Off-Policy E…

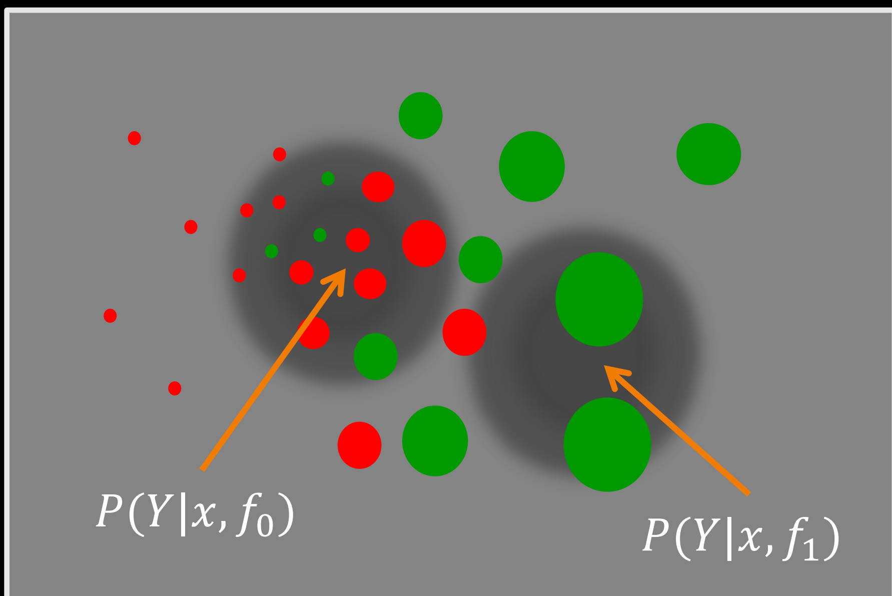
  Given $S = ((x_1, y_1, \delta_1), \ldots, (x_n, y_n, \delta_n))$ collected under f_0,

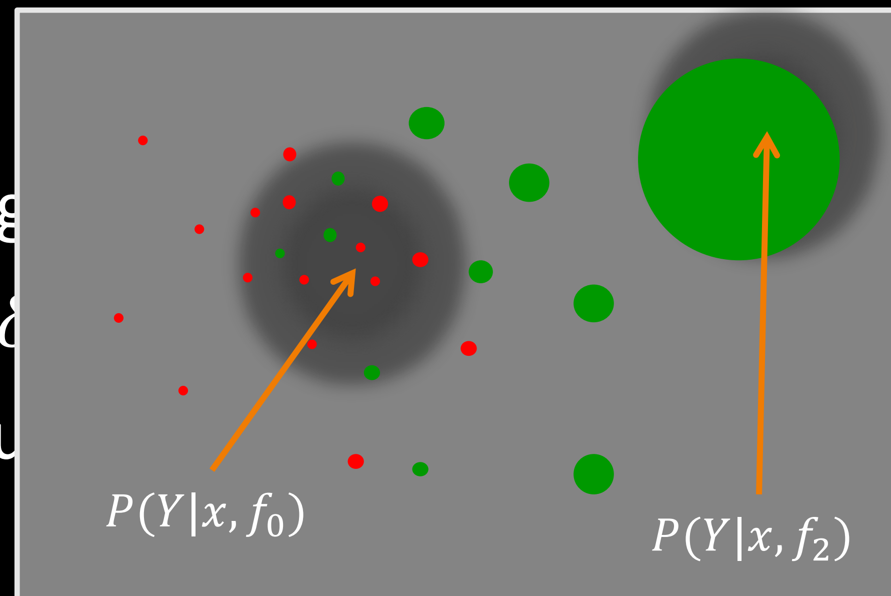
$$\hat{R}(f) = \frac{1}{n} \sum_{i=1}^{n} \delta_i \frac{P(y_i|x_i, f)}{P(y_i|x_i, f_0)}$$

Propensity weight

[Rubin, 1983] [Langford, Li, 2009.]

Partial Information Empirical Risk Minimization





$$\hat{f} := \operatorname{argmax}_{f \in H} \sum_{i}^{n} \frac{P(y_i|x_i, h)}{p_i} \delta_i$$

[Langford & Li], [Bottou, et al., 2014]

Counterfactual Risk Minimization

- Theorem [Generalization Error Bound]
  - For any bounded capacity H, for all $f \in H$ with probability $1-\eta$

$$\mathrm{U}(f) \geq \widehat{Mean}\left(\frac{P(y_i|x_i,f)}{p_i}\delta_i\right) - O\left(\sqrt{\widehat{Var}\left(\frac{P(y_i|x_i,f)}{p_i}\delta_i\right)}\right)$$

Unbiased Estimator

Variance Control

- Intuition
  - De-bias estimator through propensity weighting
  - Correct for differences in variance of estimator for $f \in \mathrm{H}$

→ Constructive principle for designing learning algorithms: Counterfactual Risk Minimization (CRM)

[Swaminathan & Joachims, 2015]

CoStA Learning Algorithm

- Counterfactual Stochastic Approximator (CoStA)
  - Hypothesis space
    - $\mathrm{P}(y|x,w) = \exp\big(w \cdot \phi(x,y)\big)/Z(x)$
  - Training objective

$$w = \underset{w \in \Re^N}{\operatorname{argmax}} \left[\widehat{Mean}\left(\frac{P(y_i|x_i,w)}{p_i}\delta_i\right) - \lambda_1 \sqrt{\widehat{Var}\left(\frac{P(y_i|x_i,w)}{p_i}\delta_i\right)} \right]$$

Unbiased Estimator

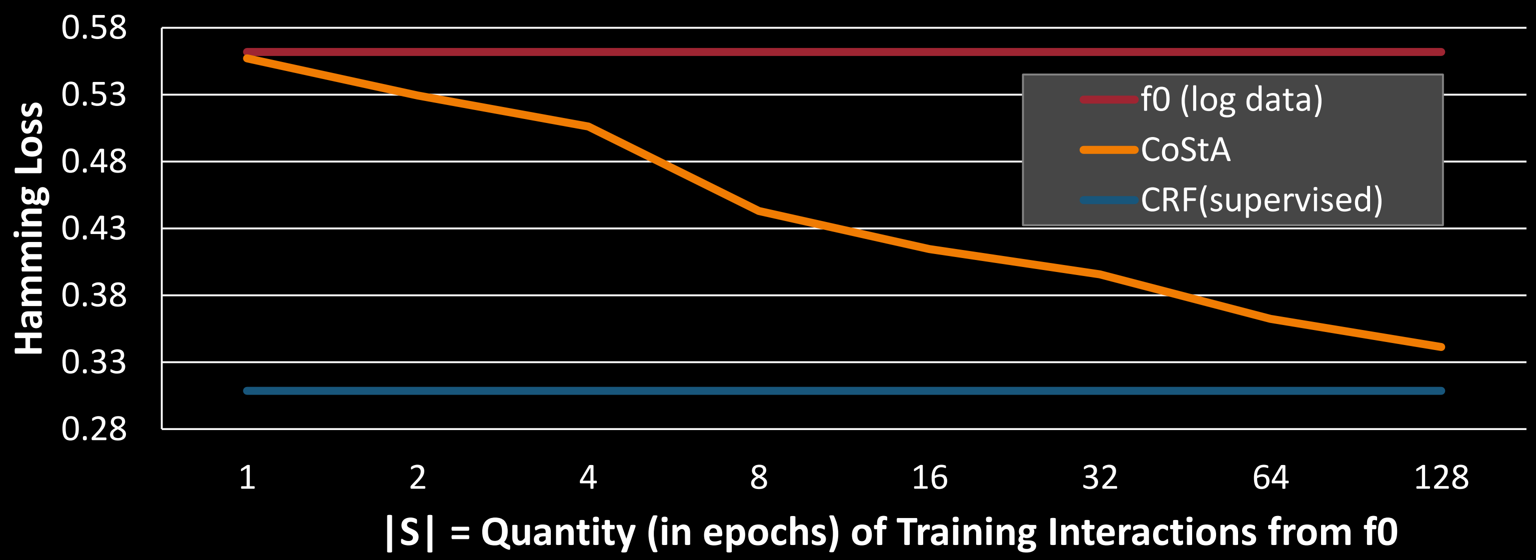
Variance Control

  - Optimization
    - successive Taylor majorization → stochastic gradient

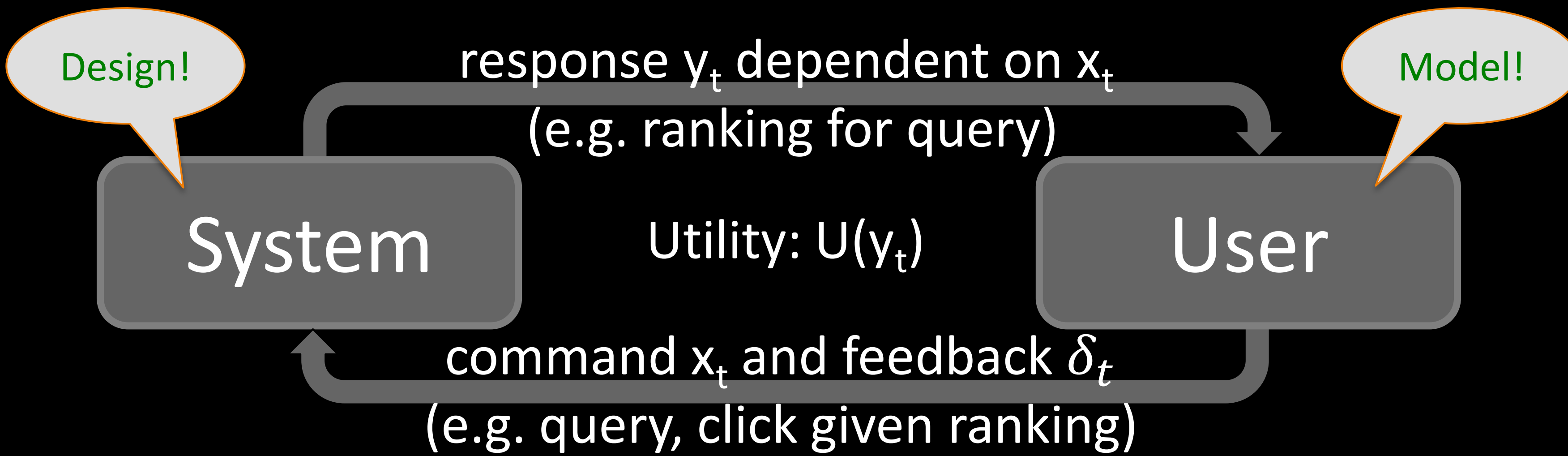
[Swaminathan & Joachims, 2015]

CoStA Experiment

- Experiment Setup
  - x: Reuters RCV1 text document
  - y: label vector with 4 binary labels
  - δ: number of incorrect labels
  - H: Isomorphic to CRF with one weight vector per label
- Results
  - Use f_0 to collect logs and train CoStA



Learning from User Interactions Conclusions



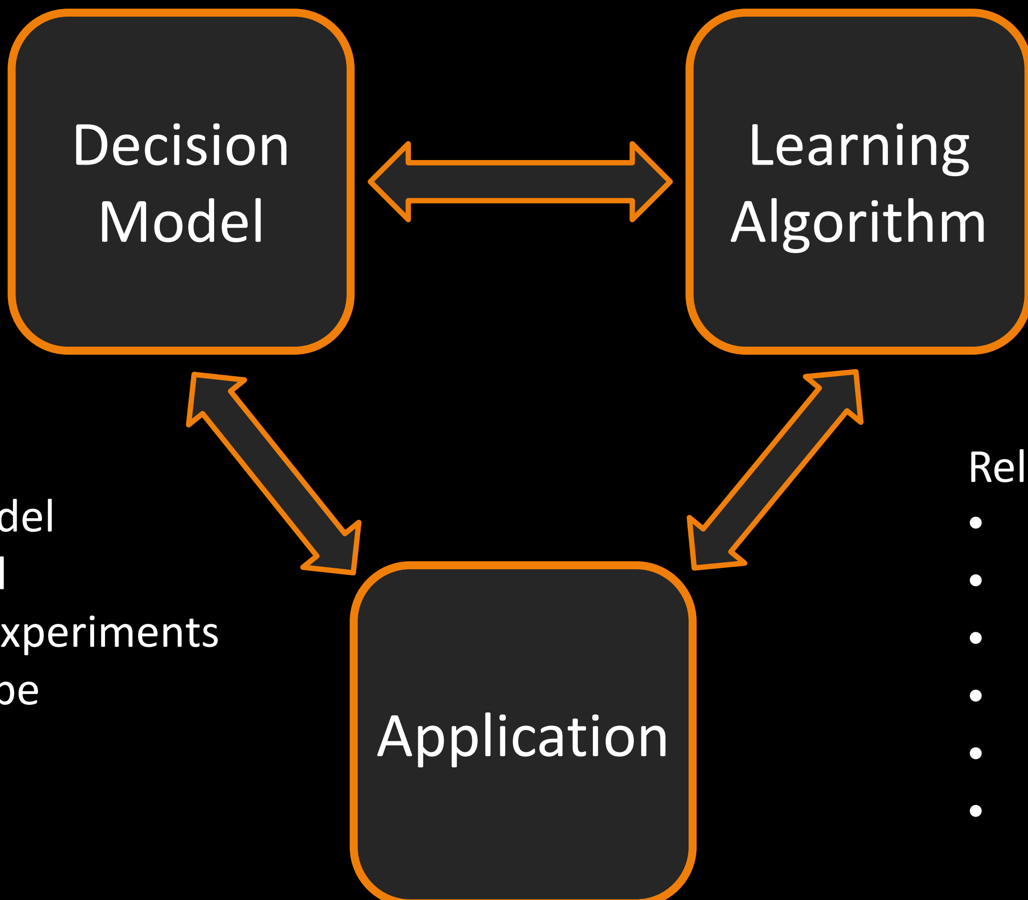
- Designing Information Elicitation Interventions

ML =

Algorithm + Interventions

- – Counterfactual Risk Minimization and CoStA Algorithm

Learning from Human Decisions



Design Space:
- Decision Model
- Utility Model
- Interaction Experiments
- Feedback Type
- Regret
- Applications

Related Fields:
- Micro Economics
- Decision Theory
- Econometrics
- Psychology
- Communications
- Cognitive Science

Contact: tj@cs.cornell.edu
Software + Papers: www.joachims.org