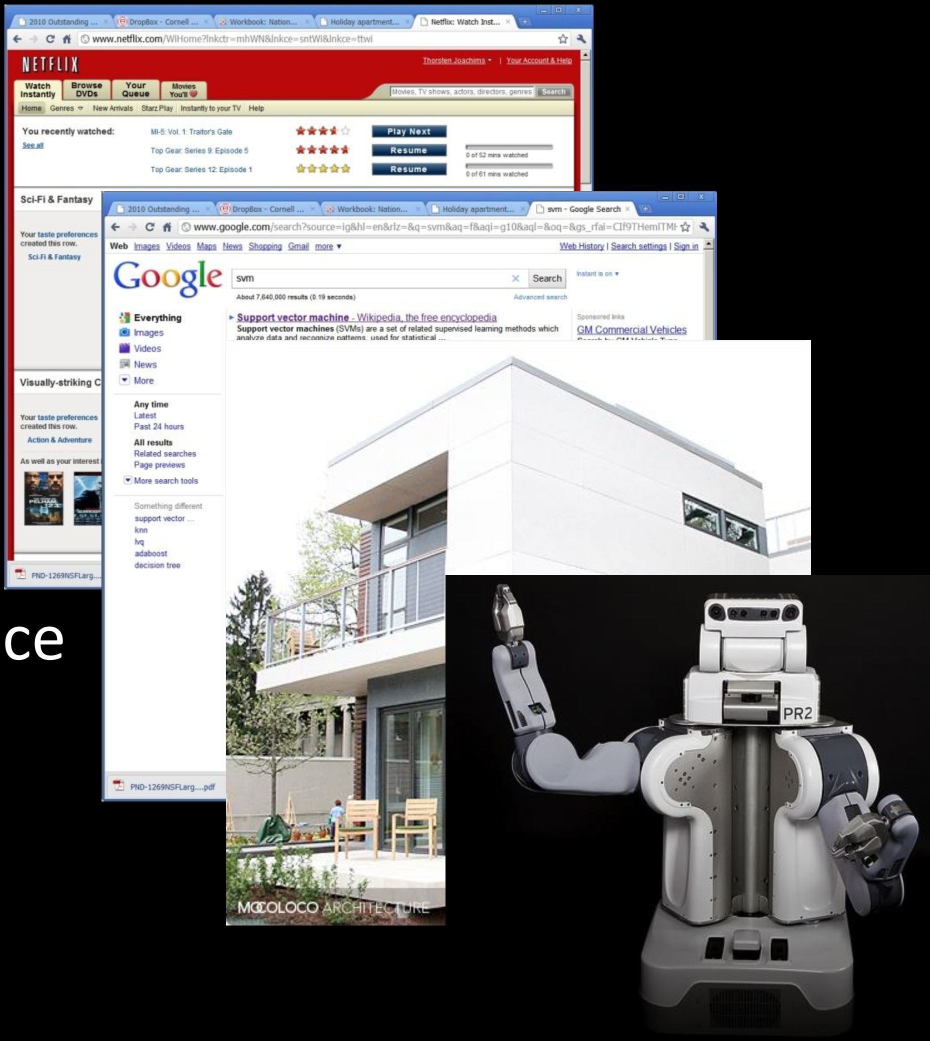
Online Structured Prediction via Coactive Learning

P. Shivaswamy, T. Joachims

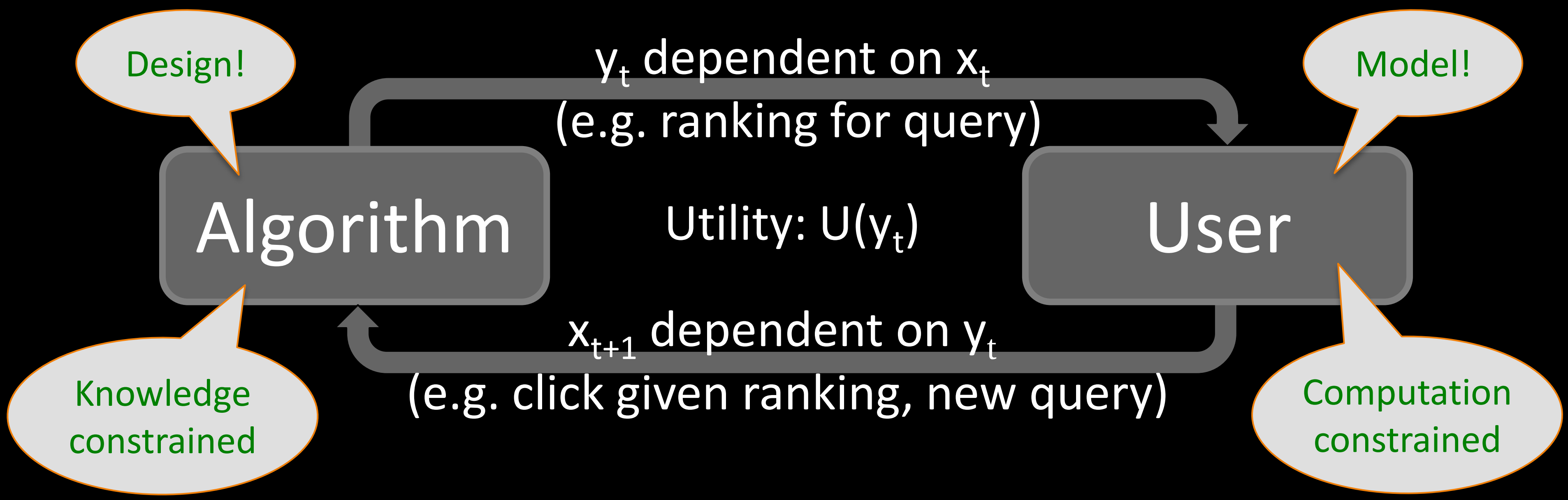
Department of Computer Science
Cornell University

User-Facing Machine Learning

- Examples
  - Search Engines
  - Netflix
  - Smart Home
  - Robot Assistant
- Learning
  - Gathering and maintenance of knowledge
  - Measure and optimize performance
  - Personalization

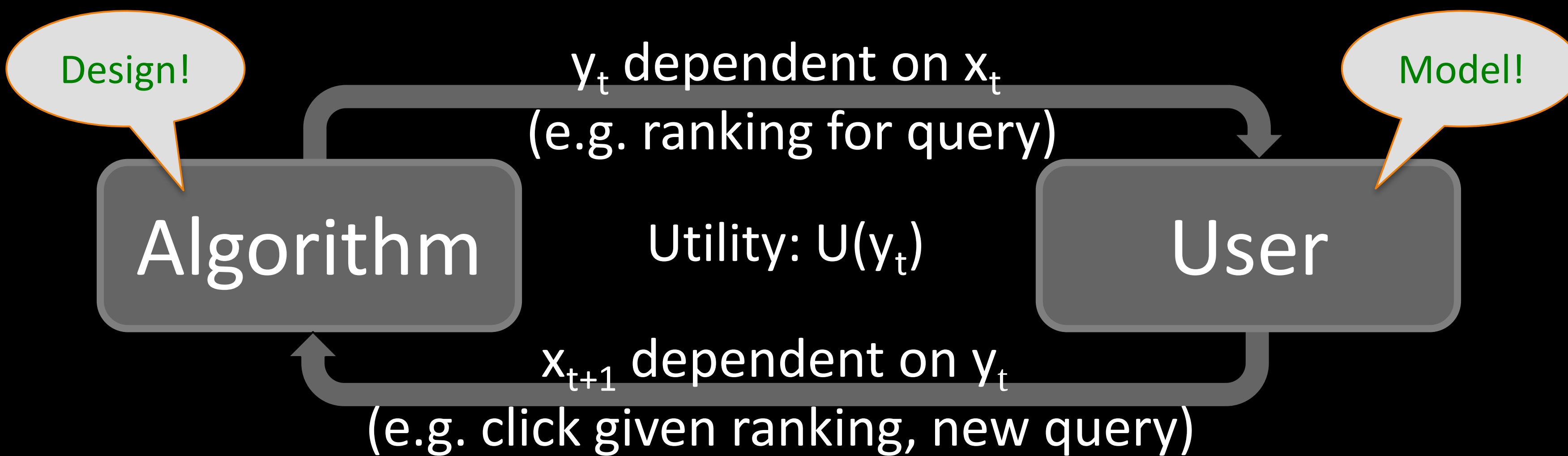


Interactive Learning System



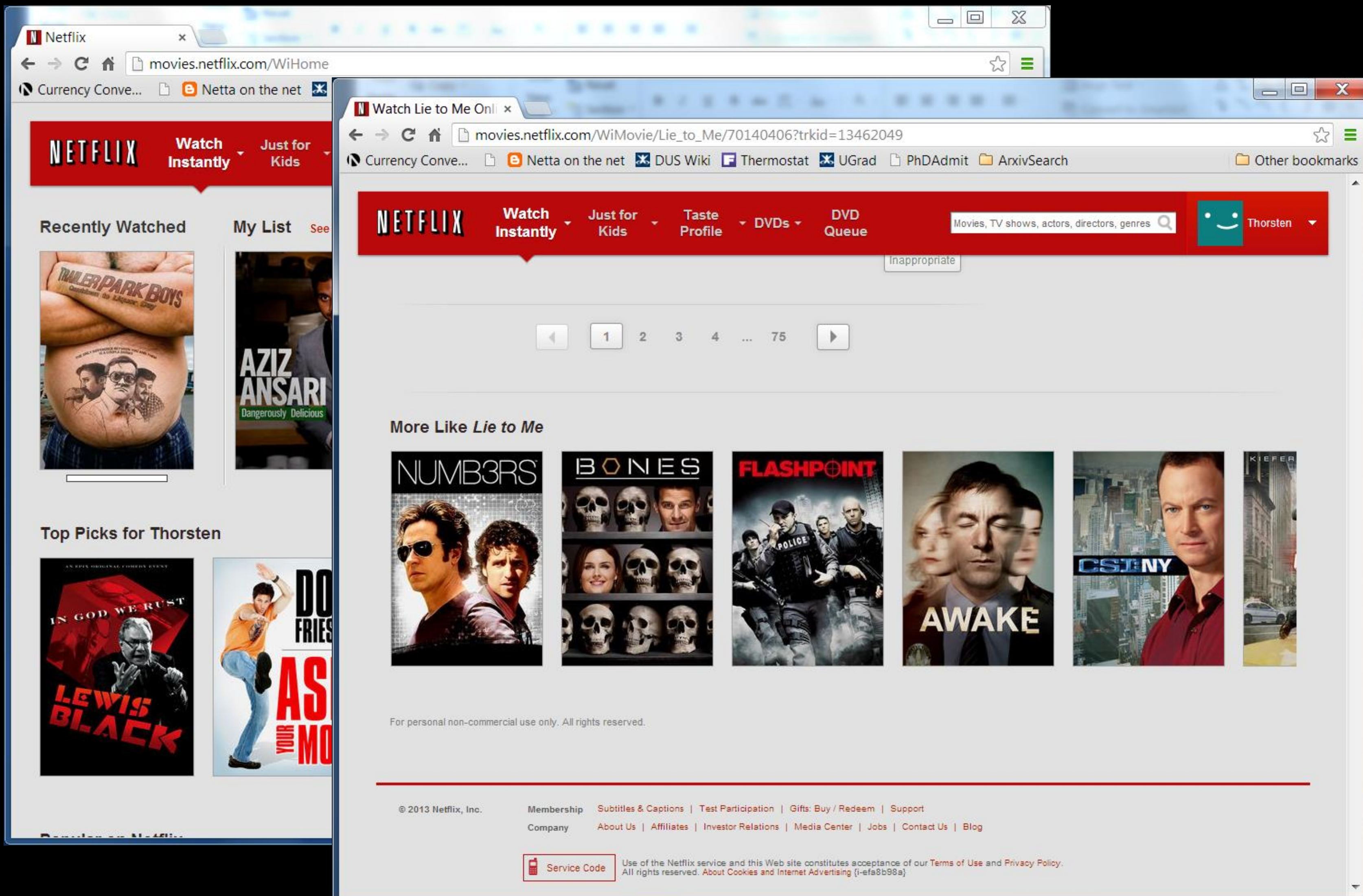
- Observed Data ≠ Training Data
  - Observed data is user's decisions
  - Need to understand decision process to infer feedback
- Decisions → Feedback → Learning Algorithm

Interactive Learning System

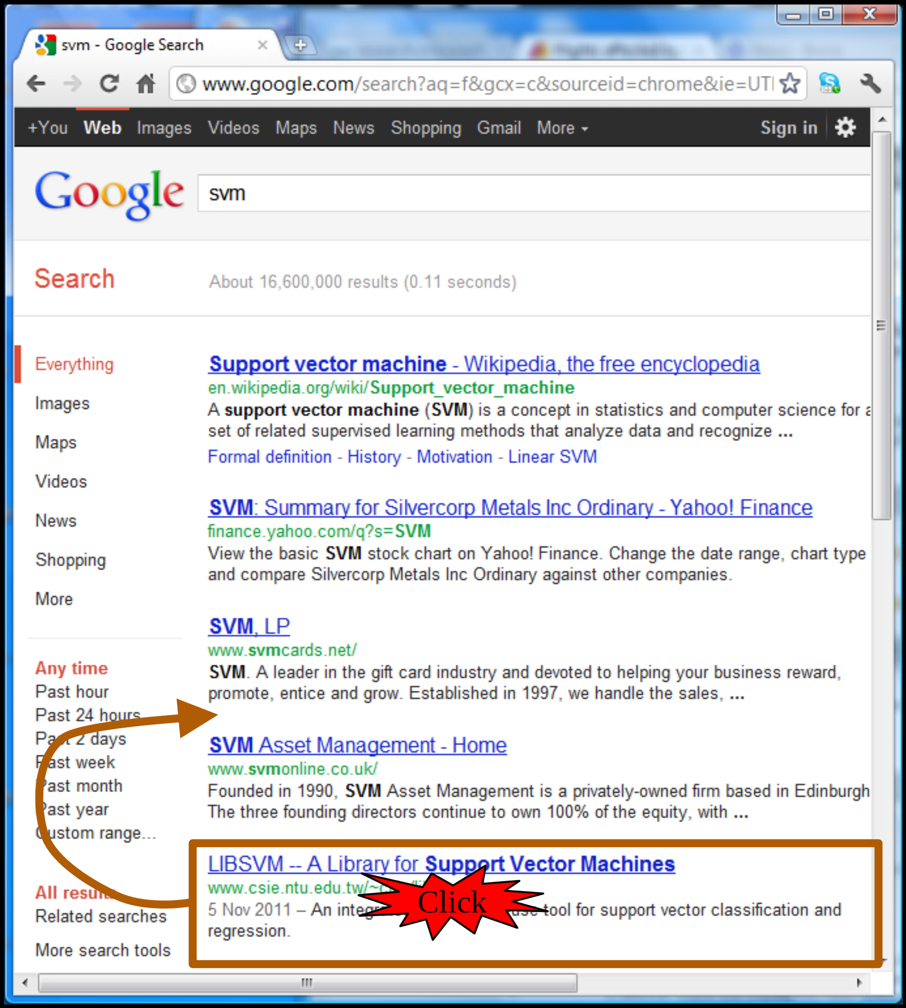


- Observed Data ≠ Training Data ✓
- Decisions → Feedback → Learning Algorithm
  - Model the users decision process to extract feedback ✓
    → Pairwise comparison test $P(y_i \succ y_j \mid U(y_i)>U(y_j))$
  - Design learning algorithm for this type of feedback ✓
    → Dueling Bandits problem and algorithms (e.g. IF1 and IF2)

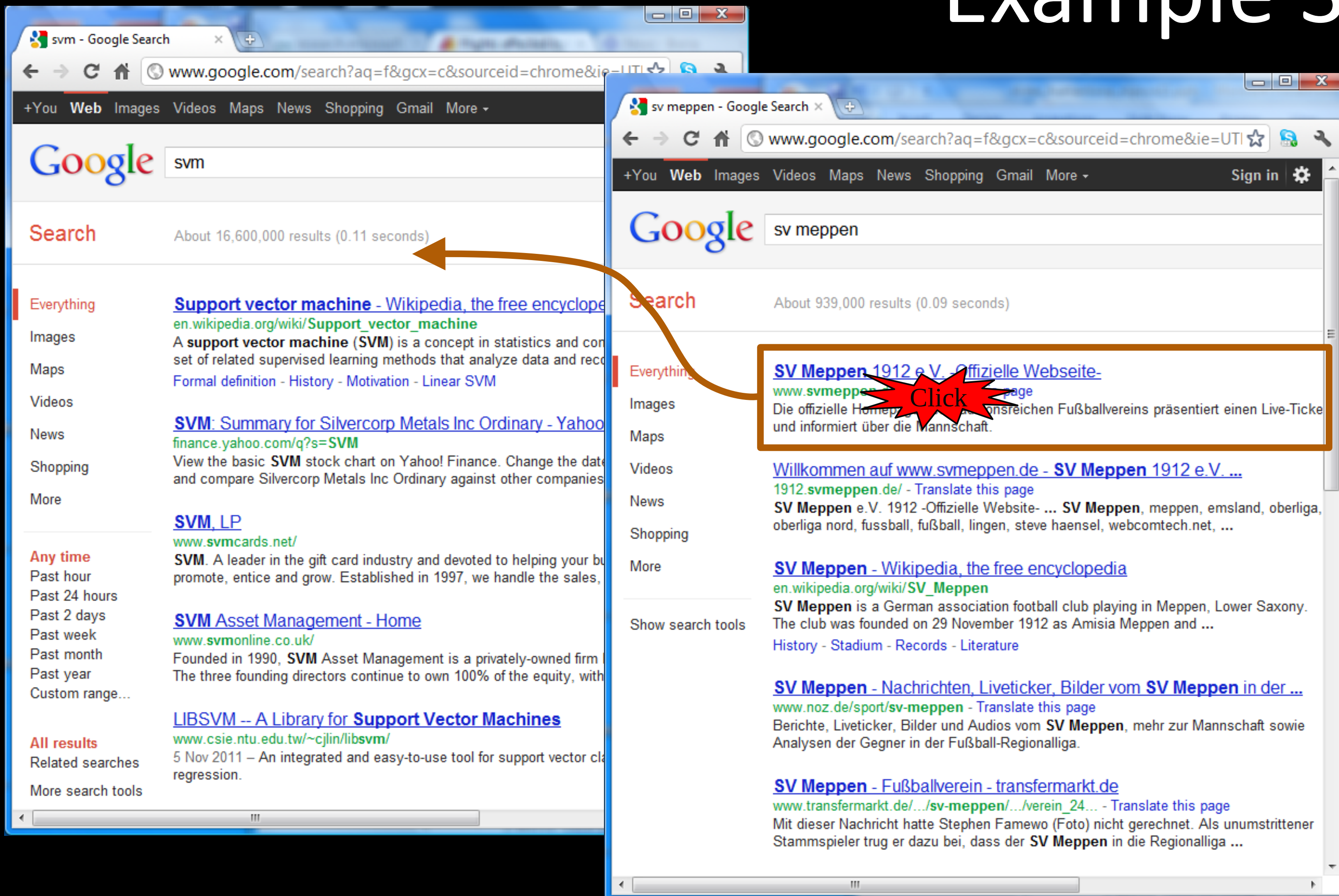
Who does the exploring?
Example 1



Who does the exploring? Example 2

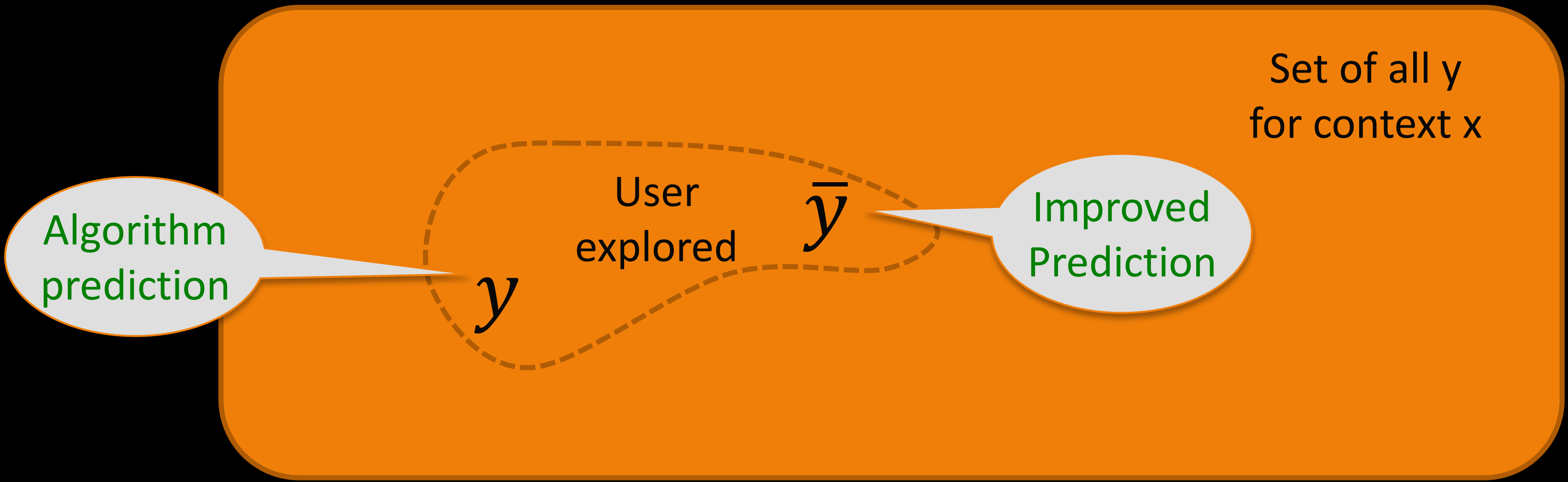


Who does the exploring?
Example 3



Coactive Feedback Model

- Interaction: given x



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

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

  - Supervised learning: optimal prediction y_t^*

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

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 Learning Model

- Unknown Utility Function: $U(y|x)$
  - Boundedly rational user
- Algorithm/User Interaction:
  - LOOP FOREVER
    - Observe context x (e.g. query)
    - Learning algorithm presents y (e.g. ranking)
    - User returns $\bar{y}$ with $U(\bar{y}|x) > U(y|x)$
    - Regret = Regret + [$U(y^*|x) - U(y|x)$]

Never revealed:
- cardinal feedback
- optimal y^*

Loss for prediction $\hat{y}$

Optimal prediction $y^* = \mathrm{argmax}_y \{ U(x,y) \}$

- Relationship to other online learning models
  - Expert setting: receive $U(y|x)$ for all y
  - Bandit setting: receive $U(y|x)$ only for selected y
  - Dueling bandits: for selected y and $\bar{y}$, receive $U(\bar{y}|x) > U(y|x)$
  - Coactive setting: for selected y, receive $\bar{y}$ with $U(\bar{y}|x) > U(y|x)$

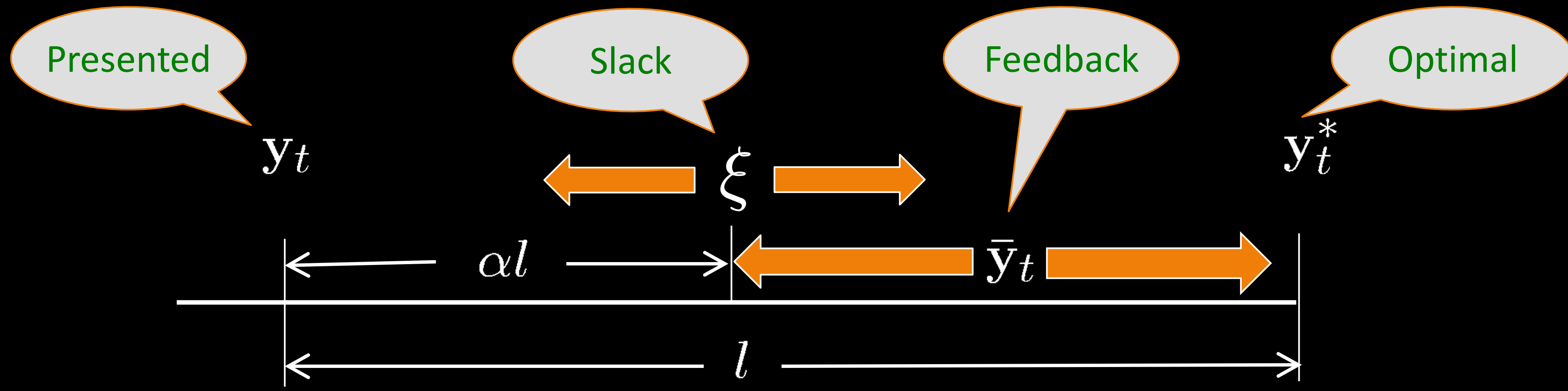
Preference Perceptron: Algorithm

- Model
  - Linear model of user utility: $U(y|x) = w^T \phi(x,y)$
- Algorithm
  - Set $w_1 = 0$
  - FOR t = 1 TO T DO
    - Observe x_t
    - Present $y_t = \text{argmax}_y \{ w_t^T \phi(x_t,y) \}$
    - Obtain feedback $\bar{y}_t$
    - Update $w_{t+1} = w_t + \phi(x_t,\bar{y}_t) - \phi(x_t,y_t)$
- 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)

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

[Shivaswamy, Joachims, 2012]

α-Informative Feedback



Feedback ≥ Presented + α (Best – Presented)

- Definition: Strict α-Informative Feedback

$$U(\mathbf{x}_t, \bar{\mathbf{y}}_t) \geq U(\mathbf{x}_t, \mathbf{y}_t) + \alpha(U(\mathbf{x}_t, \mathbf{y}_t^*) - U(\mathbf{x}_t, \mathbf{y}_t))$$

- Definition: α-Informative Feedback

Slacks both pos/neg

$$U(\mathbf{x}_t, \bar{\mathbf{y}}_t) = U(\mathbf{x}_t, \mathbf{y}_t) + \alpha(U(\mathbf{x}_t, \mathbf{y}_t^*) - U(\mathbf{x}_t, \mathbf{y}_t)) - \xi_t$$

[Shivaswamy, Joachims, 2012]

Preference Perceptron: Regret Bound

- Assumption
  - U(**y**|**x**) = $\mathbf{w}^T \phi(\mathbf{x},\mathbf{y})$, but w is unknown
- Theorem

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

$$\frac{1}{T}\sum_{t=1}^{T}[U(\mathbf{y}_t^*|\mathbf{x}) - U(\mathbf{y}_t|\mathbf{x})] \leq \frac{1}{\alpha T}\sum_{t=1}^{T}\xi_t + \frac{2R\|w\|}{\alpha\sqrt{T}}$$

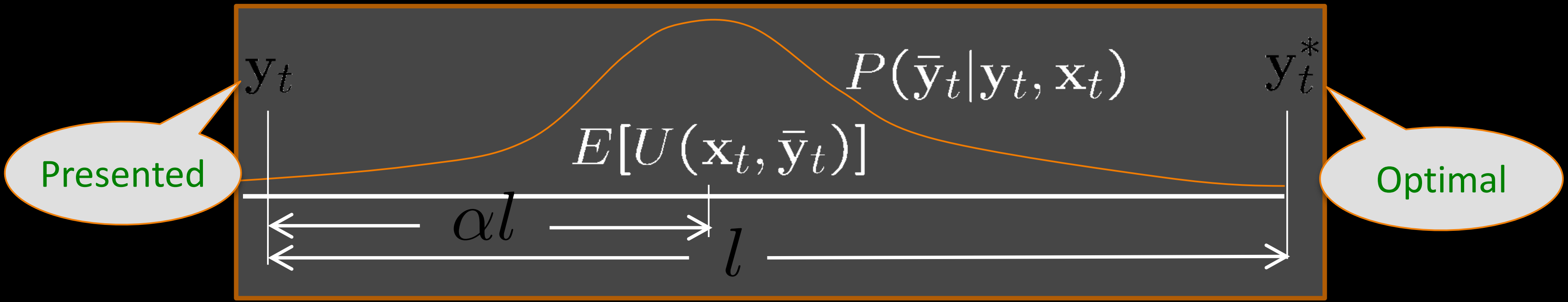
noise

→ zero

- Other Algorithms and Results
  - Feedback that is α-informative only in expectation
  - General convex loss functions of U(**y***|**x**)-U(**ŷ**|**x**)
  - Regret that scales log(T)/T instead of $T^{-0.5}$ for strongly convex

[Shivaswamy, Joachims, 2012]

Expected α-Informative Feedback



- Definition: Expected α-Informative Feedback

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

- Theorem: Coactive Pref Perceptron achieves

$$E[REG_T] \leq \frac{1}{\alpha T}\sum_{t=1}^{T}\bar{\xi}_t + \frac{2R||w||}{\alpha\sqrt{T}}$$

[Shivaswamy, Joachims, 2012]

Lower Bound

- Theorem: For any coactive learning algorithm A with linear utility, there exist $\mathbf{x}_t$, objects Y, and $\mathbf{w}$ such that REG_T of A in T steps is $\Omega(1/T^{0.5})$.

[Shivaswamy, Joachims, 2012]

Preference Perceptron: Experiment

Analogous to DCG

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 $w_t^T \phi(x,y^{(i)})$

Feedback

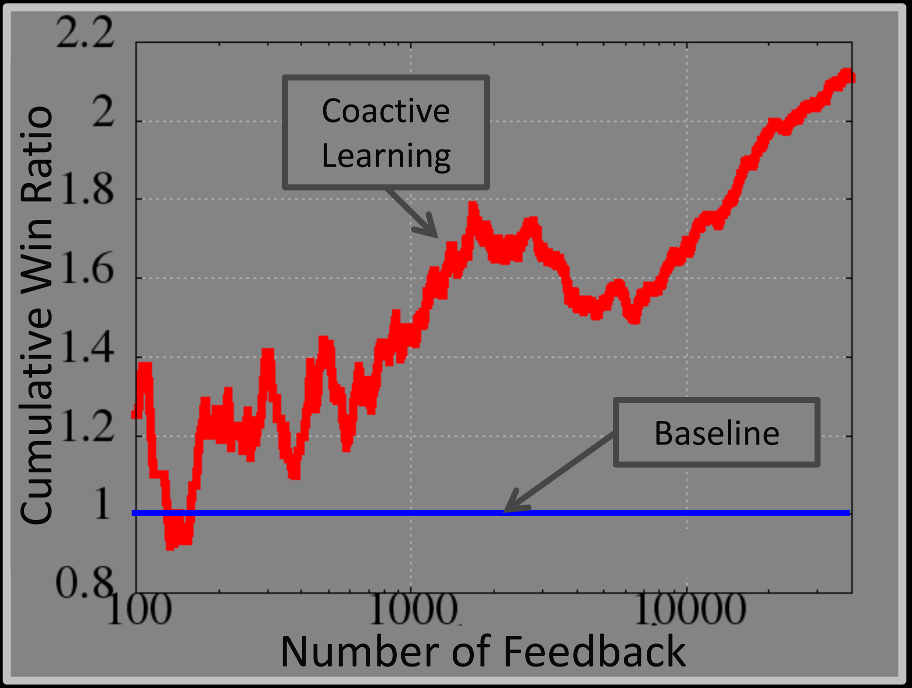
- Construct $\bar{y}_t$ from y_t by moving clicked links one position higher.

Baseline

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

Evaluation

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

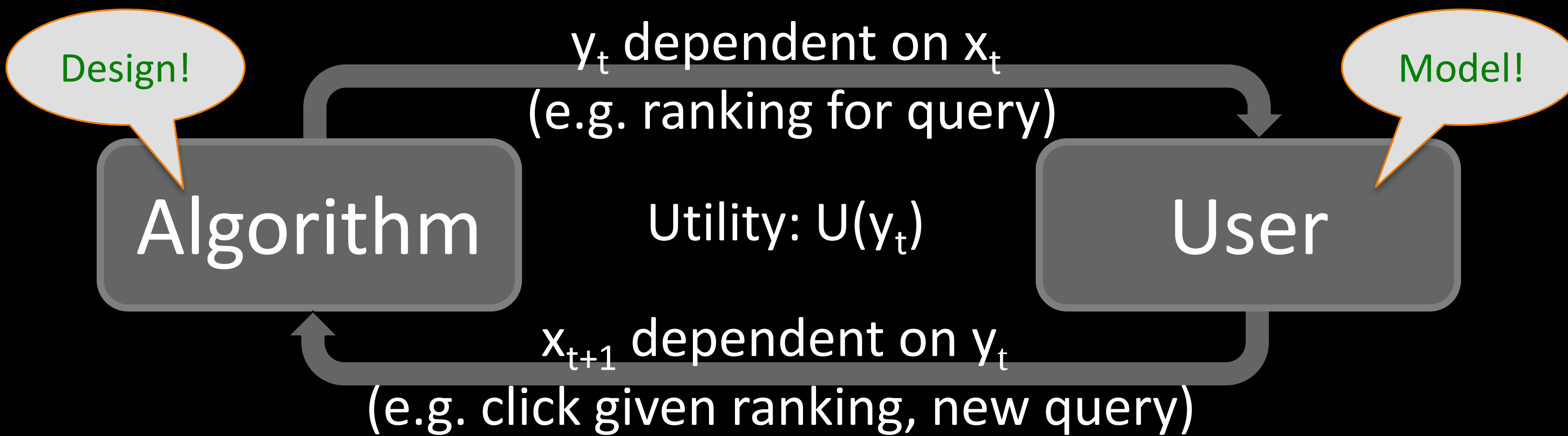


[Raman et al., 2013]

Related Models

- Ordinal Regression
  (Crammer & Singer 2001)
  – Examples: (x_i, r_i), r_i is numeric rank
- Pair Preference Learning
  (Herbrich et al., 1999; Freund et al. 2003)
  – Examples: (x_i, x_i')
  – i.i.d. assumption, batch
- Ranking
  (Joachims, 2002; Liu 2009)
  – Examples: (x_i, y_i^*), $\mathbf{y}_i^*$ is optimal ranking
  – Structured Prediction, list-wise ranking
- Expert Model
  – Cardinal feedback for all arms / optimal $\mathbf{y}_i^*$
- Bandit Model
  – Cardinal feedback only for chosen arm
- Dueling Bandit Model
  (Yue et al. 2009; Yue, Joachims 2009)
  – Preference feedback between two arms chosen by algorithm

Summary and Conclusions



	Utility model	**Decision model**	**Actions y_t / Experiment**	**Feedback**	**Exploration**	**Regret**
Dueling Bandits	Ordinal	Noisy rational	Comparison pairs	Noisy comparison	Algorithm	Lost comparisons
Coactive Learning	Linear	Bounded rational	Structured object	α-informative $\bar{y}$	User	Cardinal utility
⋮	⋮	⋮	⋮	⋮	⋮	⋮