Evaluation Methods and Challenges

Evaluation Methods

- Ideal method
  - Experimental Design: Run side-by-side experiments on a small fraction of **randomly** selected traffic with new method (treatment) and status quo (control)
  - Limitation
    - Often expensive and difficult to test large number of methods

- Problem: How do we evaluate methods offline on logged data?
  - Goal: To maximize clicks/revenue and not prediction accuracy on the entire system. Cost of predictive inaccuracy for different instances vary.
    - E.g. 100% error on a low CTR article may not matter much because it always co-occurs with a high CTR article that is predicted accurately

Usual Metrics

- Predictive accuracy
  - Root Mean Squared Error (RMSE)
  - Mean Absolute Error (MAE)
  - Area under the Curve, ROC
- Other rank based measures based on retrieval accuracy for top-k
  - Recall in test data
    - What Fraction of items that user actually liked in the test data were among the top-k recommended by the algorithm (fraction of hits, e.g. Karypsis, CIKM 2001)
- One flaw in several papers
  - Training and test split are not based on time.
    - Information leakage
    - Even in Netflix, this is the case to some extent
      - Time split per user, not per event. For instance, information may leak if models are based on user-user similarity.

Metrics continued..

- Recall per event based on Replay-Match method
  - Fraction of clicked events where the top recommended item matches the clicked one.

- This is good if logged data collected from a randomized serving scheme, with biased data this could be a problem
  - We will be inventing algorithms that provide recommendations that are similar to the current one
    - No reward for novel recommendations

Details on Replay-Match method (Li, Langford, et al)

- x: feature vector for a visit
- **r** = $[r_1, r_2, \ldots, r_K]$: reward vector for the K items in inventory
- h(x): recommendation algorithm to be evaluated
- Goal: Estimate expected reward for h(x)

$$E_{(x,r)\sim\mathcal{P}}\left[\sum_i \Pr(h(x)=i)\cdot r_i\right]$$

- s(x): recommendation scheme that generated logged-data
- $x_1,\ldots,x_T$: visits in the logged data
- r_{ti}: reward for visit t, where i = $s(x_t)$

Replay-Match continued

- Estimator

$$\frac{1}{T}\sum_t \sum_i I(h(x_t) = i \text{ and } s(x_t) = i) \cdot r_{ti} \cdot \alpha_t$$

- If importance weights $\alpha_t = \dfrac{1}{\Pr(s(x_t) = i \mid h(x_t) = i)}$ and $(x_t, r_t) \text{ iid} \sim \mathcal{P}.$
  - It can be shown estimator is unbiased
- E.g. if s(x) is random serving scheme, importance weights are uniform over the item set
- If s(x) is not random, importance weights have to be estimated through a model

Back to Multi-Objective Optimization

Recommender

EDITORIAL

content

•Clicks on FP links influence downstream supply distribution

SPORTS

NEWS

OMG

FINANCE

AD SERVER

PREMIUM display (GUARANTEED)

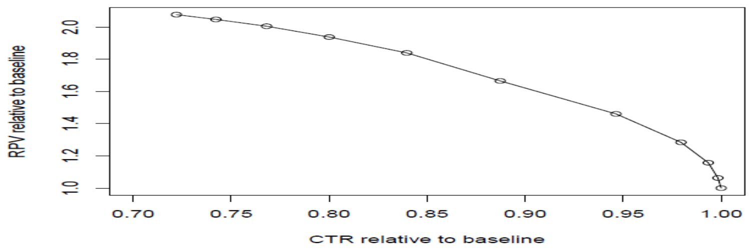
Spot Market (Cheaper)

Downstream engagement

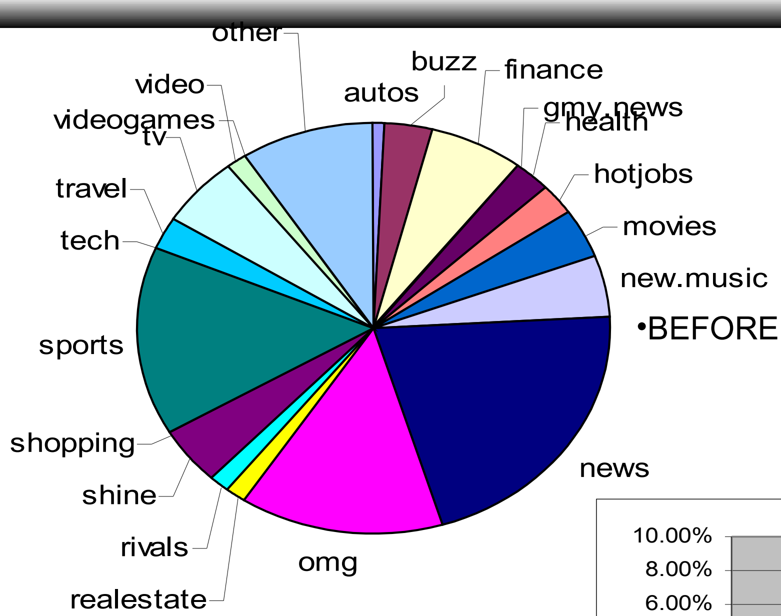
(Time spent)

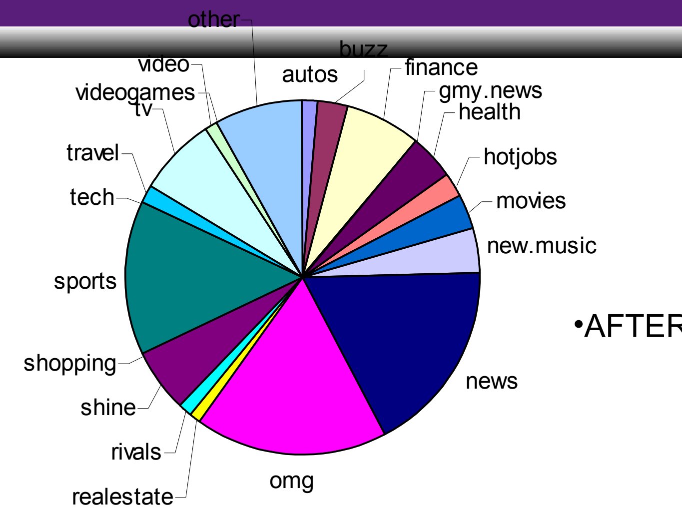
Serving Content on Front Page: Click Shaping

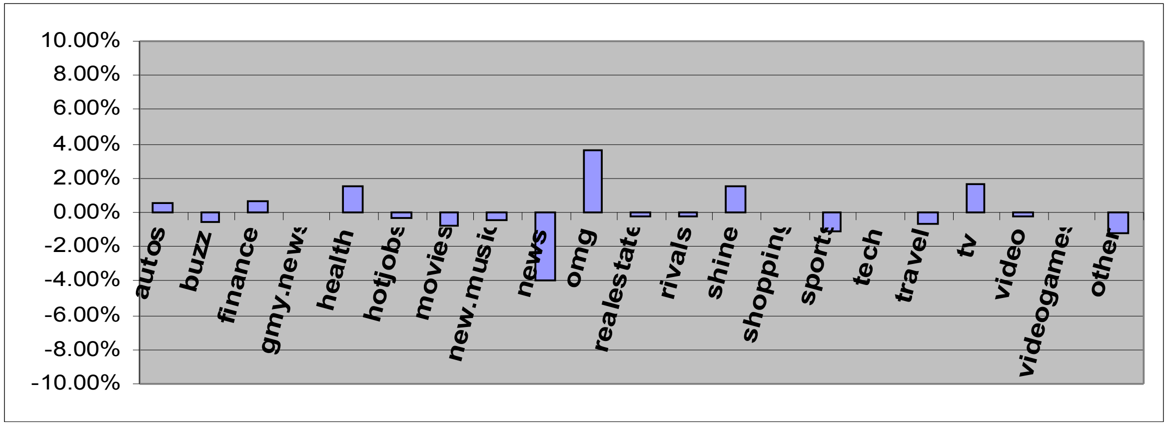
- What do we want to optimize?
- Current: Maximize clicks (maximize downstream supply from FP)
- But consider the following
  - Article 1: CTR=5%, utility per click = 5
  - Article 2: CTR=4.9%, utility per click=10
    - By promoting 2, we lose 1 click/100 visits, gain 5 utils
- If we do this for a large number of visits --- lose some clicks but obtain significant gains in utility?
  - E.g. lose 5% relative CTR, gain 40% in utility (revenue, engagement, etc)



Why call it Click Shaping?







- Supply distribution
- Changes

- SHAPING can happen with respect to any downstream metrics (like engagement)

Multi-Objective Optimization

K properties

$\mathcal{P} = \{P_1, ..., P_K\}$

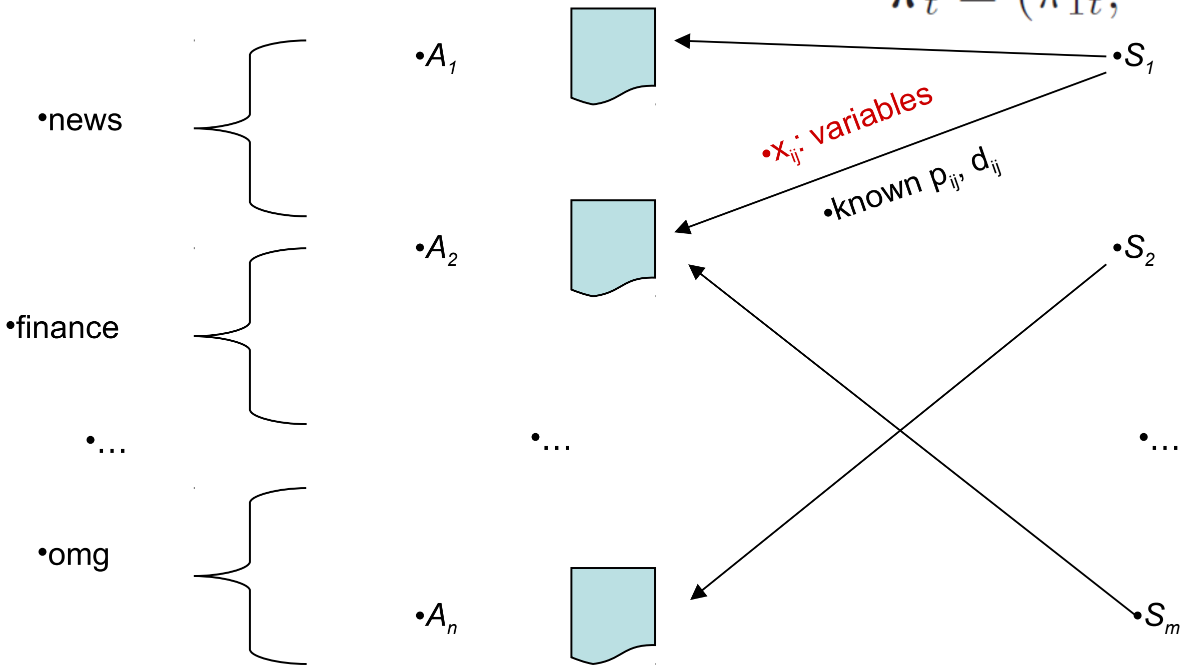
n articles

$\mathcal{A}_t = (A_1, \cdots, A_{n_t})$

m user segments

$\mathcal{S} = \{S_1, \cdots, S_m\}$

$\boldsymbol{\pi}_t = (\pi_{1t}, \cdots, \pi_{Mt})$



- CTR of user segment *i* on article *j*: p_{ij}
- Time duration of *i* on *j*: d_{ij}

Multi-Objective Program

- Scalarization

$$\lambda \cdot TotalClicks(\mathbf{x}) + (1 - \lambda) \cdot Downstream(\mathbf{x})$$

$$x_{ij} = \begin{cases} 1, & \text{if } j = \arg\max_J \lambda \cdot p_{iJ} + (1-\lambda) \cdot p_{iJ} d_{iJ} \\ 0, & \text{otherwise} \end{cases}$$

Goal Programming

$$\begin{aligned} \text{maximize} \; & Downstream(\mathbf{x}) \\ \text{s.t.} \; & TotalClicks(\mathbf{x}) \geq \alpha \cdot TotalClicks^* \end{aligned}$$

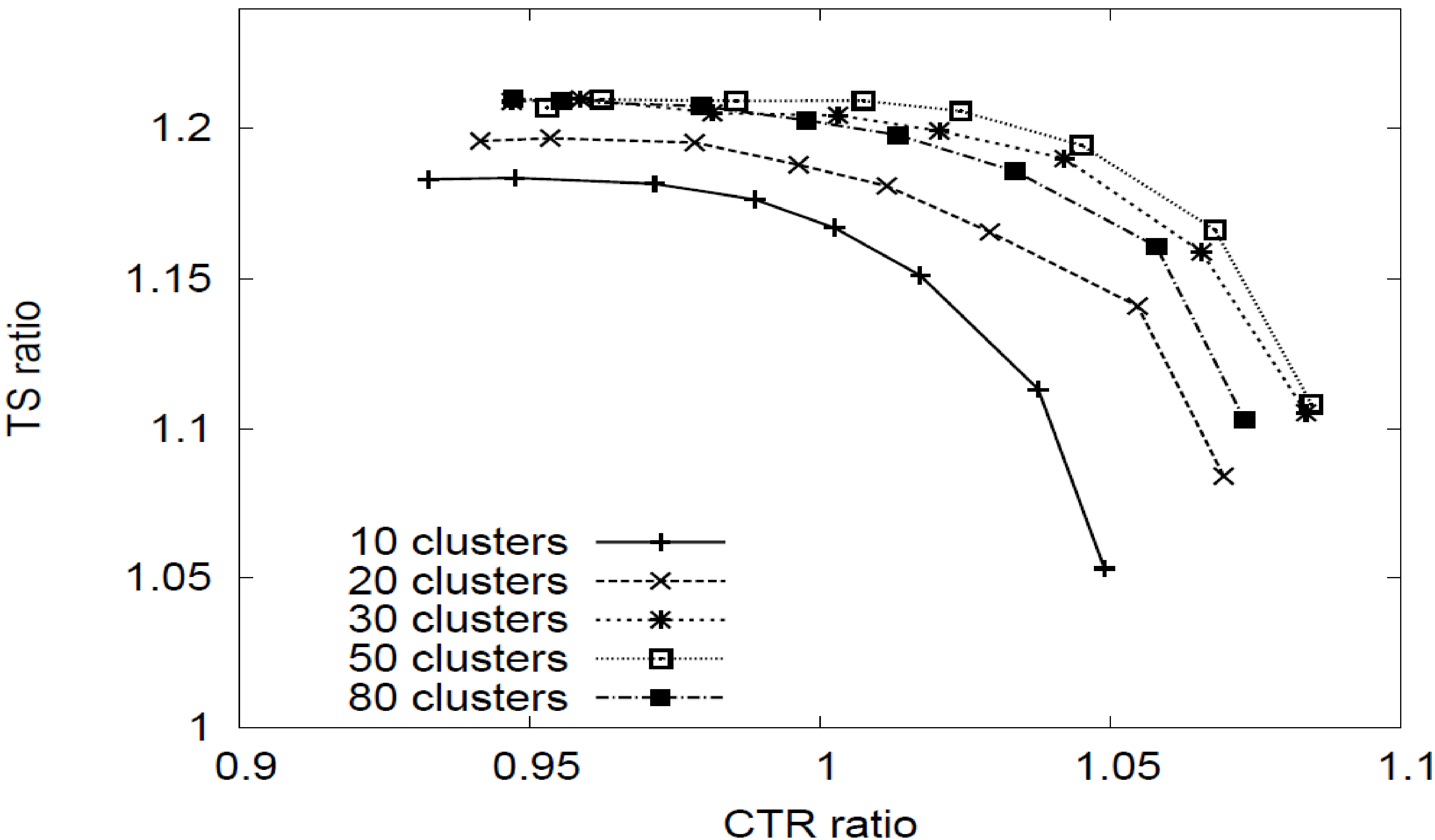
Simplex constraints on x_{iJ} is always applied

Constraints are linear

Every 10 mins, solve x

Use this x as the serving scheme in the next 10 mins

Pareto-optimal solution (more in KDD 2011)



Summary

- Modern recommendation systems on the web crucially depend on extracting intelligence from massive amounts of data collected on a routine basis
- Lots of data and processing power not enough, the number of things we need to learn grows with data size
- Extracting grouping structures at coarser resolutions based on similarity (correlations) is important
  - ML has a big role to play here
- Continuous and adaptive experimentation in a judicious manner crucial to maximize performance
  - Again, ML has a big role to play
- Multi-objective optimization is often required, the objectives are application dependent.
  - ML has to work in close collaboration with engineering, product & business execs

Challenges

Recall: Some examples

- Simple version
  - I have an important module on my page, content inventory is obtained from a third party source which is further refined through editorial oversight. Can I algorithmically recommend content on this module? I want to drive up total CTR on this module
- More advanced
  - I got X% lift in CTR. But I have additional information on other downstream utilities (e.g. dwell time). Can I increase downstream utility without losing too many clicks?
- Highly advanced
  - There are multiple modules running on my website. How do I take a holistic approach and perform a simultaneous optimization?

For the simple version

- Multi-position optimization
  - Explore/exploit, optimal subset selection
- Explore/Exploit strategies for large content pool and high dimensional problems
  - Some work on hierarchical bandits but more needs to be done
- Constructing user profiles from multiple sources with less than full coverage
  - Couple of papers at KDD 2011
- Content understanding
- Metrics to measure user engagement (other than CTR)

Other problems

- Whole page optimization
  - Incorporating correlations
- Incentivizing User generated content
- Incorporating Social information for better recommendation
- Multi-context Learning