Show Me the Money: Dynamic Recommendations for Revenue Maximization

Wei Lu
Shanshan Chen
Keqian Li
Laks V.S. Lakshmanan

University of British Columbia
Why Dynamic?

• Dynamic recommendations for **optimal revenue** (VLDB’14)
  
  • A model connecting recommendations and revenue

  • Near-optimal revenue-maximizing algorithm
    
    • approximation by using matroids theory
Repeated Recommendations?

user $i$
Saturation

• Repetitions could result in *saturation*

• Cyclic trends in social choices (Das Sarma et al., WSDM’12)

• Devaluation occurs after repeated exposure (Kapoor et al., KDD’13)

• **Impression discounting**: impressions without action are *implicit negative feedback* (Li et al., KDD’14)
A set of items

- capacity constraint $c_i$
- prices $p(i, t), \forall t \in [T]$ (or, a distribution of prices)
- class label $C(i)$

Adoption probability $q(u, i, t)$, for all $u \in U, i \in I, t \in [T]$
- e.g., derived from a valuation distribution

k items per day
(partition matroid constraint)
Revenue Function

- Let \( S \subseteq U \times I \times [T] \) be a **recommender strategy**
  - \( S \) is a set of **user-item-time triples**
  - \((u, i, t) \in S\): recommend item \( i \) to user \( u \) at time step \( t \)
- For any triple \((u, i, t)\), its **dynamic adoption probability** w.r.t. \( S \) is defined as

\[
q(u, i, t|S) = q(u, i, t|\emptyset) \cdot \beta_i^{M(u, i, t|S)} \cdot \Pr[\neg \text{buy}(u, C(i), t - 1)]
\]

- \( q(u, i, t|\emptyset) \): basic adoption probability
- \( \beta_i \in [0, 1] \): saturation parameter (smaller \( \rightarrow \) stronger effect)
- \( M(u, i, t|S) \): user \( u \)'s memory on item \( i \) up to time \( t \)
- **Total expected revenue** generated by \( S \) is

\[
\mathcal{R}(S) = \sum_{(u, i, t) \in S} p(i, t) \cdot q(u, i, t|S)
\]
Revenue Maximization

- **Revenue Maximization**: Find the triple-set $S$ that maximizes expected total revenue

- while respecting both item capacity constraint and user attention constraint (top-k)

- NP-hard, reduction from Timetable Design (1975)

- Practically, input size need not be $|U \times I \times [T]|$. 
Approximation

• **Theorem:** $R(S)$ is **submodular** and **non-monotone**
  
  - submodularity: diminishing marginal return
  
  - monotonicity: non-decreasing value

• **Theorem** (Lee et al., STOC’09): For maximizing a non-monotone submodular function s.t. a partition matroid constraint, there is a local search algorithm with approximation factor of $1/(4 + \text{eps})$
Faster Heuristic Algorithm

- Time complexity of Local search $O((U \times I \times [T])^4)$

- **Global Greedy heuristic:**
  - **UNTIL** all users receive $k$ items in every time step
    - find a candidate triple with the *largest positive marginal revenue* (MR)
    - **IF** both constraints OK: take it
    - **IF** no triple gives positive MR: stop
Global Greedy Speedup

- Greedy has a priority queue for fast retrieval of values
- We use a **two-level heap structure** for better efficiency

**lower-level heaps:**
- one per user-item pair
- size at most $T$
- not visible to greedy

**1 upper-level heap:**
- queue of “champions”
- size at most $|U| \times |I|$
- visible to greedy
Global Greedy Speedup

• Once a triple is selected by Greedy, all other triples with same user and item class should have marginal revenue updated

• Lazy evaluation:
  • Most candidate triples are never promising
  • Thus, update only a candidate appears at top of queue
  • (Trick: use a flag value to keep track)
Empirical Evaluations

- **Amazon**: 5K items, 23K users, 16.1M triples
  - 3-month crawling of prices and reviews
  - ground-truth price series
- **Epinions**: 1K items, 21.3 K users, 14.9M triples
  - some users report price paid in reviews
- **Synthetic**: up to 250M triples, for scalability
Scalability of Global Greedy
Revenue Comparison

Baselines:
• Global Greedy ignoring saturation
• Top-K items by expected revenue (price * basic adoption probability.)
• Top-K items by predicted ratings
“Submodularity Trend”

Growth rate of revenue diminishes (submodular)
Repeated Recommendations

As $\beta$ increases, saturation effect is less, more repetitions are observed.
Thank you!

http://www.cs.ubc.ca/~welu
welu@cs.ubc.ca