Google’s AdWords Market: How Theory Influenced Practice

Vijay V. Vazirani

University of California, Irvine
Two major achievements of Google

1). **Search**: That is robust to spam

2). **Advertising**: Business model
Revolution in advertising

- Matching merchants with customers
  - How can a merchant pitch ads in a targeted way to customers who are interested in his goods.

- Difficulty: How to find out needs/desires of customers in a super-efficient manner?

- Insight: A user’s search queries reveals to Google a succinct window into her mind/needs
Solution: Auction off queried keywords to advertisers!

This converts a giant undefined matching problem into a gigantic auction!

Google is world’s biggest auction house: billions daily!
How to allocate keywords to advertisers?

- Advertisers bid for specific keywords
- Maximize Google’s revenue – efficient solution!

- **GREEDY**: Display the ad of highest bidder
  - Assume: only 1 ad is shown

- Charge bid, or second highest bid (Vickery auction)

This soln. was being used!
Problem ...
Start Reaching Prospects with AdWords Today:

1: Choose target languages and countries
   Learn how

2: Create Ad Groups
   Learn how
   ✓

3: Set your daily budget
   Learn how
   ✓

4: Create your account
   Learn how

Done:

Help
Example:

<table>
<thead>
<tr>
<th></th>
<th>Bidder 1</th>
<th>Bidder 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book</td>
<td>$1</td>
<td>$0.99</td>
</tr>
<tr>
<td>CD</td>
<td>$1</td>
<td>$0</td>
</tr>
</tbody>
</table>

Bidder 1 = Bidder 2 = $100

Queries: 100 Books then 100 CDs

Algorithm Greedy

Lost

Revenue $100
Example:

<table>
<thead>
<tr>
<th></th>
<th>Bidder 1</th>
<th>Bidder 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book</td>
<td>$1</td>
<td>$0.99</td>
</tr>
<tr>
<td>CD</td>
<td>$1</td>
<td>$0</td>
</tr>
</tbody>
</table>

B1 = B2 = $100

Queries: 100 Books then 100 CDs

Optimal Allocation

Revenue $199
Must incorporate budgets in solution

- Large fraction of advertisers are budget-constrained
- Budget distribution is heavy-tailed
  
  Must handle small budget advertisers adequately

- Solution must be real time and particularly simple!
Adwords Problem

Mehta, Saberi, Vazirani & V, 2005:

Simple framework that captures essential features.

Will add bells and whistles later!
The Adwords Problem

N advertisers.

- Daily Budgets $B_1, B_2, \ldots, B_N = \$1$
- Each advertiser provides bids for keywords he is interested in.

queries (online) \hspace{2cm} \text{Search Engine} \hspace{2cm} \text{Select one Ad}

Advertiser pays his bid
Online competitive analysis

- Compare revenue of online algorithm with that of best offline allocation.

- E.g., GREEDY has competitive ratio of $\frac{1}{2}$. 
Algorithm

- Will use agents’
  - Bid
  - Fraction of budget spent

- “Correct” tradeoff between them given by a special function $f$
Algorithm

\[ f(\alpha) = 1 - e^{-(1-\alpha)} \]

Award next query to bidder with max bid \( \times f \) (fraction of budget spent)

1 \(-\) 1/e competitive ratio, assuming bid \( \ll \) budget

Simple, minimalistic solution!
Impact

- Provided a general framework for thinking about budget constrained auctions for many Ad products

- **Bid scaling** used widely in industry for ad auctions and for display ads:
  - Very fast: one operation per bid
  - Low on memory: one number per bid
  - No extra communication
New algorithmic work

- Numerous models:
  - Queries arrive in random order
  - Queries are picked from a distribution
  - Submodular constraints on allocations
  - Use historical data about query arrival
  - . . .
Solution has roots in “pure” theory, in particular, Matching Theory!
Online bipartite matching

$U$

. . . . .
Online bipartite matching
Online bipartite matching

$U \quad V$

Diagram showing connections between nodes in U and V.
Online bipartite matching
Online bipartite matching
Online bipartite matching
Any deterministic algorithm has competitive ratio of $\frac{1}{2}$.

Karp, Vazirani & V., 1990:

$1 - \frac{1}{e}$ factor randomized algorithm

Optimal!

Same as simple case of Adwords problem:

All budgets $1$, all bids $0/1$. 
Online $b$-matching

- Kalyanasundaram & Pruhs, 1996:
  $1 - 1/e$ factor algorithm for $b$-matching:
  Daily budgets $b$, bids $0/1$, $b \gg 1$

**BALANCE:** Assign query to interested bidder who has spent least so far.

Optimal!
Example: Balance

<table>
<thead>
<tr>
<th></th>
<th>Bidder 1</th>
<th>Bidder 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book</td>
<td>$1</td>
<td>$1</td>
</tr>
<tr>
<td>CD</td>
<td>$1</td>
<td>$0</td>
</tr>
</tbody>
</table>

\[ B_1 = B_2 = \$100 \]

Queries: 100 Books then 100 CDs

Balance Algorithm

Revenue
\[ \frac{3}{4} \text{-competitive} \]
Where did this tradeoff function come from?

1-1/e

New Proof for BALANCE \{0,1\}

Factor Revealing LP

(Jain, Mahdian, Markakis, Saberi & V., ’03)

Modify LP for arbitrary bids \[0,1\]

1-1/e

Use dual to get Tradeoff function

Tradeoff Revealing LP
New proof of BALANCE

- $N$ bidders, $\$1$ each. Each bid is $\$\epsilon$.

- OPT: optimal offline, ALG: BALANCE

- OPT = $N$

- Will show: \[ \text{ALG} \geq N \left( 1 - \frac{1}{e} \right) \]
Idea: Upper bound no. of bidders who spent less!

Assume $k$ large

$S_i$: Bidders who spent $\left[ \frac{i-1}{k}, \frac{i}{k} \right)$

Let $|S_i| = x_i$. Will constrain $x_1$, $x_1 + x_2$, …
Partition revenue of ALG

$\varepsilon \in \text{layer } i$ if bidder had \( \left[ \frac{i-1}{k}, \frac{i}{k} \right) \)
when this money was spent.
Layer 2
Layer 1

ALG

$S_2$
$S_1$
Partition revenue of ALG

$\varepsilon \in layer \ i \ if \ bidder \ had \ \left[ \frac{i-1}{k}, \frac{i}{k} \right] \ when \ this \ money \ was \ spent.$

BALANCE assigns next query to interested bidder who has spent least so far.
Layer 1 $\leq \frac{N}{k}$

• First Constraint: $|S_1| = x_1 \leq \text{layer 1} \leq \frac{N}{k}$
Layer 2 \leq \frac{N-x_1}{k}

Second Constraint: \(|S_1 \cup S_2| = x_1 + x_2 \leq \frac{N}{k} + \frac{N-x_1}{k}\)
Lower bound on ALG

LP\((N)\): \[ \min \sum_{i=1}^{k} \frac{i}{k} x_i \]

s.t. constraints on \(x_1, x_2, \ldots, x_k\)

\[
ALG \geq N \left( 1 - \left( 1 - \frac{1}{k} \right)^k \right) \geq N \left( 1 - \frac{1}{e} \right)
\]
Factor revealing LP

- Jain, Mahdian, Markakis, Saberi & V., ’03

- Family of LPs: LP($N$) encodes problem of finding lower bound for instance of size $N$.

- Infimum of optimal solutions gives approximation factor
Where did this tradeoff function come from?

Factor Revealing LP
(Jain, Mahdian, Markakis, Saberi & V., ’03)

New Proof for BALANCE {0,1}

Modify LP for arbitrary bids [0,1]

Use dual to get Tradeoff function

Tradeoff Revealing LP

1-1/e

1-1/e
Larger Theme
AdWords, Amazon, eBay, Yahoo!, Alibaba, Uber, Apple (iTunes), Airbnb, Cloud Computing …
Markets on the Internet

- Numerous new algorithmic and game-theoretic issues raised.
- Much scope for creative work that can have a huge impact!