An Open Source Advertisement Server

Advisor : Dr. Pollett
Committee Members : Dr. Khuri and Dr. Mak

Presented By
Pushkar Umaranikar
Agenda

• Introduction to Online Advertising
• Project Idea
• Google’s Advertising platform
• Different types of auctions
• Advertisement ranking using Cosine Similarity
• Advertisement placing service in Yioop
• Advertisement Platform in Yioop
• A novel Auction System in Yioop
• Conclusion and Future Work
Why Online Advertising?

- Can reach a targeted audiences anywhere in the world
- Unique targeting Capabilities
- Simple and fast
- Cost Effective
- Flexibility
Search Engine Marketing

• Important type of online Advertising
• Promotion of Product or Service by increasing its visibility in search engine results pages
• Advertisements are placed on the basis of key search terms entered in search engines
• Involve auction systems to price and place advertisement
• Initially, advertisement spaces sold on a per-thousand-impression basis
• 1997, onwards sold on a per-click-basis
Project Idea

• Design and Implement a new Online Advertisement System for the open source search engine, Yioop

• Supports selling advertisements and displaying them within search results

• Involves a novel auction system

• Also supports display of advertisements from external advertisement servers
Overview: Google AdWords

• Google’s Online Advertising Platform, started in 2002

• User can create advertisement, attach keywords to it and send to Google’s database

• Google returns advertisements if search keyword matches with attached keywords

• Ads displayed on search results page are based on search query contents
Creating A Google AdWords Campaign

1. Decide how much to spend

   Your budget: $2.00 per day

   Select maximum amount for Ad campaign

2. Choose a target audience

   Locations: United States and Canada

   Region where you want to display your advertisement

   Networks: Search Network, Display Network

   Sites that shows ad along with other content

3. Set your bid

   Bid: Automatically set my bids to get the most clicks within my budget

4. Write your ad

   Text ad:
   - YoopSearch
   - Yoopsearch.com
   - PHP Search Engine
   - Open source Search Engine

   Actual text Ad with link to landing page and text associated with it.
Google AdSense

• Allows website owners to earn revenue

• Works by matching images and text to the owner’s site based on content

• Advertisements are managed, arranged, and maintained by Google

• Publishers are not allowed to click their own advertisements

• Variety of advertisement formats such as mobile web pages, videos, online games, TV inventory etc.
Google AdSense(2)

- To display advertisements, website owners place a Javascript code provided by AdSense on web pages

- The script makes use of inline JSON to render content fetched from the Google server

- Javascript makes possible for website owners to use of the AdSense program

- Use of valuable text content is encouraged to get more revenue
AdChoices

- An icon in the corner of the advertisement
- Not a advertising provider
- Used for interest based advertising
- Helps to show ads based on the individual’s interests
Questions to ask

• Where will my advertisement appear on the list?
  ▪ Using advertisement space allocation strategy

• When will advertisers pay?
  ▪ Only when someone clicks on link

• How much to pay?
  ▪ Revenue derived from number of clicks
Introduction to Auction System

• Allocates advertisement space according to some rule
Open Auctions

• Conducted at a public venue

• General idea
  ▪ Winner pays highest bid – own bid

• Useful in one item auctions

• Types
  ▪ Ascending Price
  ▪ Descending Price
Sealed Envelope Auctions

- Each bidder submits bid privately
- All bids are then revealed to Auctioneer who decides allocation
- Useful in multi-item auctions
- Types
  - First Price Auction
    - Winner pays the highest bid
  - Second Price Auction
    - Winner pays the second highest bid
Second Price Auctions

• Winner pays bid from the next highest bidder

• Sounds “wrong”
  ▪ why not bid extremely high so that I can win the item, and then pay a much lower price for it?

• The Assumption does not hold when everyone uses same strategy
Generalized Second Price Auction (GSP)

- The highest bidder gets the most valuable ad space
- The Bidder is then left out from future bidding

<table>
<thead>
<tr>
<th>Advertising Space</th>
<th>Clicks /day</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>400</td>
</tr>
<tr>
<td>2</td>
<td>200</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Value per Click</th>
<th>Bidder</th>
<th>Advertising Space</th>
<th>Valuation</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>1</td>
<td>1</td>
<td>4800</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>1</td>
<td>3200</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>1</td>
<td>1600</td>
</tr>
</tbody>
</table>

- Bidder 1 gets ad space 1 and pays $8 per click - payoff $1600
- Bidder 2 gets ad space 2 and pays $4 per click – payoff $800
VCG Auction

• Truth Telling is not dominant in GSP
• Modification to previous example
  ▪ Bidder 1 bids $7 to target ad space 2 and has expected clicks per day 300
  ▪ Difference between the click through rates is not large enough relative to the difference in per-click payment

• In VCG, seller computes matching
• Maximizes total valuation of the system
Example: VCG Auction

<table>
<thead>
<tr>
<th>Advertising Space</th>
<th>Click Through Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bidder</th>
<th>Advertising Space</th>
<th>Valuation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>50</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>25</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>15</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>15</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

• Matching of \{(1, 1), (2, 2), (3, 3)\} would return a total valuation \(50 + 15 + 1 = 66\) which is the maximum
Advertising using Cosine Similarity

• Search Engines use a bag of words model to retrieve relevant documents

• Ranking algorithms are used to rank the documents

• We extend this technique to get contextual advertisements

• We used words from the retrieved documents to get advertisements
Advertising using Cosine Similarity(2)

• Built a bag of words model using an inverted Index

• Positional Index is used to build inverted Index

  < term: number of docs containing term
  doc1:position1,position2...;   
  doc2:position1,position2...; >

• With a positional index, it is easy to deal with phrase queries
Advertising using Cosine Similarity(3)

• Bag of words model built using document indexing. Following URL’s are used.

https://pushkarumaranikar.wordpress.com/
http://en.wikipedia.org/wiki/Android_(operating_system)
http://en.wikipedia.org/wiki/Online_and_offline
http://software.canon-europe.com/
http://www.canonrumors.com/
http://en.wikipedia.org/wiki/Cloud_computing
http://www.interoute.com/what-are-cloud-servers
http://www.rackspace.com/cloud/servers

• Tokenized words from these URL’s and formed a bag of words model
Cosine Similarity Ranking

• Measure of Similarity between two vectors
• Calculated using the inner product space of two vectors that measure the cosine of the angle between them
• Gives judgment of orientation
• For a query vector $\mathbf{q}$ and a document vector $\mathbf{d}$, when similarity is defined as the cosine of the angle between them

$$\text{sim} (\mathbf{d}, \mathbf{q}) = \frac{\mathbf{d} \cdot \mathbf{q}}{|\mathbf{d}| \cdot |\mathbf{q}|}$$
Finding relevant documents

Cosine similarity applied to the query vector and document vector

```
Enter search query:
cloud
Document vector created for document :0
cosine similarity for given query with document 0 :0.0
Document vector created for document :1
cosine similarity for given query with document 1 :0.004937291752307908
Document vector created for document :2
cosine similarity for given query with document 2 :0.0
Document vector created for document :3
cosine similarity for given query with document 3 :0.0
Document vector created for document :4
cosine similarity for given query with document 4 :0.0
Document vector created for document :5
cosine similarity for given query with document 5 :0.003579073207718753
Document vector created for document :6
cosine similarity for given query with document 6 :0.5771441342195869
Document vector created for document :7
cosine similarity for given query with document 7 :0.10466739886057416
Document vector created for document :8
cosine similarity for given query with document 8 :0.43019880494981483
Document vector created for document :9
cosine similarity for given query with document 9 :0.0
```

http://en.wikipedia.org/wiki/Cloud_computing => 0.5771441342195869
http://www.interoute.com/what-arc-cloud-servers => 0.43019880494981483
http://en.wikipedia.org/wiki/Android_(operating_system) => 0.004937291752307908
http://en.wikipedia.org/wiki/Database => 0.003579073207718753
https://pushkarumaranikar.wordpress.com/ => 0.0
http://en.wikipedia.org/wiki/Online_and_offline => 0.0
http://software.canon-europe.com/ => 0.0
http://www.canonrumors.com/ => 0.0
http://www.rackspace.com/cloud/servers => 0.0
Retrieving relevant advertisements

• Use a subset retrieved documents to build a document vector for ranking advertisements
• New set of documents is formed with the entered keywords and the immediate three words associated with it on either side
• For Example, set of documents for query “cloud” is,
Retrieving relevant advertisements (2)

- Cosine similarity measured for each of these documents with advertisements in inventory

- Most relevant advertisements for query “cloud” are,

  Relevant Ads:
  cloud computing for dummies => 0.9045340337332909
  applications in the cloud => 0.6666666666666666
  servers for businesses => 0.3535339059327373
  web design and marketing solutions for business websites => 0.2773500981126146
Advertisement Placing Service in Yioop

• Allows advertisements from an external advertisement server to be displayed on a web page

• Provider shares revenue with web site owner

• Easy way to make money

• In Yioop, we are providing a configurable option for an advertisement placing service
Getting into it...

• Added configuration options for advertising space and advertisement placing service
• Select advertisement space within Yioop
• Place a Javascript code provided by the external advertisement network for the selected advertising space
• Currently, Yioop supports top, side, and both alignment options
• Javascript is passed to the search controller and is executed during rendering the view
Flow Chart for Advertisement Placing Service

1. Start
2. Select Advertisement alignment
3. Enter JavaScript code under selected advertisement alignment
4. Pass Alignment details and script to the search controller.
5. Add script and alignment details to data
6. Send data to view for rendering
7. Request external advertisement network for advertisement response
8. Display rendered view
9. END
Configuration of Ad Placing Service

Ad Server Configuration

- Ad Location
  - Top
  - Side
  - Both
  - None

Global Element Ad Script

Top Element Ad Script

```html
<!-- AdSpeed.com Serving Code 7.9.7 for [Ad] Click Me [Text Ad] -->
<script type="text/javascript" src="http://g.adspeed.net/ad.php?do=js&aid=216142&oid=18025&target=blank"></script>
</script>
<!-- AdSpeed.com End -->
```

Side Element Ad Script

```html
<!-- AdSpeed.com Serving Code 7.9.7 for [Ad] Click Me [Text Ad] -->
<script type="text/javascript" src="http://g.adspeed.net/ad.php?do=js&aid=216142&oid=18025&target=blank"></script>
</script>
<!-- AdSpeed.com End -->
```

Option to select advertisement alignment on search result page

Add ad script from external advertisement program based on option selected
Online Auction System in Yioop

- Accepts bids for every Campaign
- Calculates minimum bid required and campaign created only when budget is over minimum bid
- Program then runs a ranking algorithm when users enter certain search queries
- Factors considered for ranking are budget and keywords association
- Only contextual advertisements are displayed on the search results page
Details...

• Added new user role, known as “Business user”
• Only Business user can create campaign
• Anyone can become a business user by providing credit card details
• Configuration available under Manage Accounts section
Manages Advertisements Activity

**Account Access**
- Manage Account

**Social**
- Manage Groups
- Feeds and Wikis
- Mix Crawls

**Advertisements**
- Manage Advertisements

---

**Create Advertisement**

- Advertisement Name: testAdv
- Text Description: test advertisement
- Destination URL: test.com
- Start Date: 04/30/2015
- End Date: 05/03/2015
- Keywords: test, describe
- Minimum Bid Required: 8
- Expensive word: describe
- Advertisement Type: Text Ad
- Budget: 16

Value should be more than minimum budget required

Advertisement activity for Business users only
Advertisement List

• Option for activate and deactivate advertisement is available

• Expired advertisements gets automatically deactivated

<table>
<thead>
<tr>
<th>Ad Name</th>
<th>Destination URL</th>
<th>Keywords</th>
<th>Budget</th>
<th>Start Date</th>
<th>End Date</th>
<th>Impressions</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
<td>google.com</td>
<td>search</td>
<td>6</td>
<td>2015-05-13</td>
<td>2015-05-13</td>
<td></td>
<td>Edit, Deactivate</td>
</tr>
<tr>
<td>adv600Update</td>
<td>google.com</td>
<td>test, describe</td>
<td>100</td>
<td>2015-05-08</td>
<td>2015-05-15</td>
<td>15</td>
<td>Edit, Deactivate</td>
</tr>
</tbody>
</table>
About Create Advertisement Form

• Includes name, description, destination URL, start date, end date, keywords and budget
• Minimum bid required value is calculated by the auction system depending upon the number of keywords and their popularity
• Enter a budget amount that is equal to or greater than the minimum required budget
• System also displays expensive words among the keywords associated
• It helps to manage budget
Database Schema for an Auction System
How system calculates minimum bid?

- Every day is considered as a separate transaction for the same advertisement.
- If keyword has not already been used before, then a minimum bid set is to 1 on that date.
- For the same keyword, the bid amount may vary for different dates.
- Example
  - A user is creating a new campaign called “Test Advertisement”. Keywords entered are “test” and “adv”. Date range is May 1, 2015 to May 7, 2015.
How system calculates minimum bid? (2)

To calculate the minimum bid, half the existing bid amount for every day for every associated keyword. The minimum bid for the word “test”

\[
\begin{align*}
(10/2) + (8/2) + (10/2) + (12/2) + (10/2) + (8/2) + 1 \\
= 5 + 4 + 5 + 6 + 5 + 4 + 1 = 30
\end{align*}
\]

Similarly, the minimum bid required for word the “adv” = 33

The minimum bid required to create the campaign = 30 + 33 = 63

And the expensive word in this case is, “adv”.

### Keywords & Bids

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Bid Amount</th>
<th>Bid Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
<td>10</td>
<td>May 1</td>
</tr>
<tr>
<td>Test</td>
<td>8</td>
<td>May 2</td>
</tr>
<tr>
<td>Test</td>
<td>10</td>
<td>May 3</td>
</tr>
<tr>
<td>Test</td>
<td>12</td>
<td>May 4</td>
</tr>
<tr>
<td>Test</td>
<td>10</td>
<td>May 5</td>
</tr>
<tr>
<td>Test</td>
<td>8</td>
<td>May 6</td>
</tr>
<tr>
<td>Test</td>
<td></td>
<td>May 7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Bid Amount</th>
<th>Bid Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adv</td>
<td>8</td>
<td>May 1</td>
</tr>
<tr>
<td>Adv</td>
<td>12</td>
<td>May 2</td>
</tr>
<tr>
<td>Adv</td>
<td>10</td>
<td>May 3</td>
</tr>
<tr>
<td>Adv</td>
<td>12</td>
<td>May 4</td>
</tr>
<tr>
<td>Adv</td>
<td>8</td>
<td>May 5</td>
</tr>
<tr>
<td>Adv</td>
<td>4</td>
<td>May 6</td>
</tr>
<tr>
<td>Adv</td>
<td>12</td>
<td>May 7</td>
</tr>
</tbody>
</table>
How system updates bid amount?

- Let us consider budget entered = 126

\[
update Factor = \left( \frac{Bid \ of \ the \ day}{2} \div \text{min bid required} \right) \times Budget
\]

- It is a fraction of a minimum bid and budget distributed to scale over proportionally
- For Example, Update factor of the keyword “test” on May 1 is,

\[
= \left( \frac{5}{63} \right) \times 126
\]

\[
= 10
\]

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Bid Amount</th>
<th>Bid Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
<td>10 + 10 = 20</td>
<td>May 1</td>
</tr>
<tr>
<td>Test</td>
<td>8 + 8 = 16</td>
<td>May 2</td>
</tr>
<tr>
<td>Test</td>
<td>10 + 10 = 20</td>
<td>May 3</td>
</tr>
<tr>
<td>Test</td>
<td>12 + 12 = 24</td>
<td>May 4</td>
</tr>
<tr>
<td>Test</td>
<td>10 + 10 = 20</td>
<td>May 5</td>
</tr>
<tr>
<td>Test</td>
<td>8 + 8 = 16</td>
<td>May 6</td>
</tr>
<tr>
<td>Test</td>
<td>2</td>
<td>May 7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Bid Amount</th>
<th>Bid Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adv</td>
<td>8 + 8 = 16</td>
<td>May 1</td>
</tr>
<tr>
<td>Adv</td>
<td>12 + 12 = 24</td>
<td>May 2</td>
</tr>
<tr>
<td>Adv</td>
<td>10 + 10 = 20</td>
<td>May 3</td>
</tr>
<tr>
<td>Adv</td>
<td>12 + 12 = 24</td>
<td>May 4</td>
</tr>
<tr>
<td>Adv</td>
<td>8 + 8 = 16</td>
<td>May 5</td>
</tr>
<tr>
<td>Adv</td>
<td>4 + 4 = 8</td>
<td>May 6</td>
</tr>
<tr>
<td>Adv</td>
<td>12 + 12 = 24</td>
<td>May 7</td>
</tr>
</tbody>
</table>
Display relevant advertisements

- Let us imagine, the user has entered the search query as “computer” and system finds 5 advertisements related to query

<table>
<thead>
<tr>
<th>Advertisement</th>
<th>Budget</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adv1</td>
<td>10</td>
</tr>
<tr>
<td>Adv2</td>
<td>20</td>
</tr>
<tr>
<td>Adv3</td>
<td>30</td>
</tr>
<tr>
<td>Adv4</td>
<td>40</td>
</tr>
<tr>
<td>Adv5</td>
<td>50</td>
</tr>
</tbody>
</table>

- We do a sum of the budgets of all advertisements. In this case:—
  \[
  \text{Sum} = 10 + 20 + 30 + 40 + 50 = 150
  \]
- Now, rand function will generate random number between 0 to Sum. In this case, 0 to 150
Display relevant advertisements(2)

<table>
<thead>
<tr>
<th>Advertisement</th>
<th>Budget</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adv1</td>
<td>10</td>
<td>0-10</td>
</tr>
<tr>
<td>Adv2</td>
<td>20</td>
<td>11-30</td>
</tr>
<tr>
<td>Adv3</td>
<td>30</td>
<td>31-60</td>
</tr>
<tr>
<td>Adv4</td>
<td>40</td>
<td>61 – 100</td>
</tr>
<tr>
<td>Adv5</td>
<td>50</td>
<td>101 -150</td>
</tr>
</tbody>
</table>

- Then, we calculate display range for each advertisement using budget. Range for Adv1 is 10, same as Budget. For Adv2 it is 11-30 as budget is 20.

- If the random number lies in the range 11-30, Adv2 gets displayed and so on.
Display relevant advertisements

• Probability based model is used to display advertisements
• A low budget add would also rendered but with less probability than others

• System also monitors clicks and impressions for each advertisement
Sequence Diagram – Create Advertisement

1: Log in as business user
2: Logged in successfully
3: Click on Manage Advertisements
4: Create advertisement form opened
5: Associate keywords to campaign
6: Get minimum bid required
7: Get existing bid amount
8: Bid amount retrieved
9: Calculate min bid required
10: Minimum Bid amount sent
11: Minimum Bid amount displayed
12: Add budget
13: Save advertisement
14: Add advertisement
15: Add advertisement to inventory
16: Success message
17: Status update sent
18: Advertisement created
Sequence Diagram – Display Advertisement

1: Enter search query
2: Get relevant documents
3: Request for relevant advertisement
4: Get advertisements containing required keyword
5: Advertisements sent
6: Sum the bid amounts
7: Generate random number
9: Advertisement rendered on page
Performance Testing

Number of requests Vs response time - Concurrent user requests

Rendering search results when advertisements are enabled

Rendering search results when advertisements are disabled
User Acceptance Testing

• Aim is to check whether users can use the newly designed advertisement platform easily or not.

• We noticed difficulties faced by them and received feedback.

• Feedback Comments:
  ▪ Add help text about Add keywords button.
  ▪ It is useful to have a preview of the created advertisement.
Comparison with Google AdWords

• Based on probabilistic ranking

• Even a low budget campaign gets displayed. Google uses GSP. Hence, low price bidders may not get their advertisements placed

• Users can also select active time periods for a newly created campaign. In AdWords, a user has to manually pause the campaign if the user wants to deactivate it
Conclusion

• Implemented advertising platform to enable search engine marketing in Yioop

• Developed a probability based novel auction system

• Enabled advertisement rendering from external advertisement network in Yioop

• Learned about online advertising, its working, and its importance in the world of advertising
Future Work

• Payment service can be incorporated

• Edit budget functionality can be added

• Implementation of advertisement service to display ads outside Yioop
Thank You