An Open Source Advertisement Server

A Project Report

Presented to

The Faculty of Department of Computer Science

San Jose State University

In Partial Fulfillment

Of the Requirements for the Degree

Master of Science in Computer Science

By

Pushkar Umaranikar

May 2015

SAN JOSE STATE UNIVERSITY

The Undersigned Project Committee Approves the Project Titled

AN OPEN SOURCE ADVERTISEMENT SERVER

By

Pushkar Umaranikar

APPROVED FOR THE DEPARTMENT OF COMPUTER SCIENCE

Dr. Chris Pollett, Department of Computer Science Date

Dr. Sami Khuri, Department of Computer Science Date

Dr. Ronald Mak, Department of Computer Science Date

APPROVED FOR THE UNIVERSITY

Associate Dean Office of Graduate Studies and Research Date

ABSTRACT

AN OPEN SOURCE ADVERTISEMENT SERVER

This report describes a new online advertisement system and its implementation for the Yioop open source search engine. This system was implemented for my CS298 project. It supports both selling advertisements and displaying them within search results. The selling of advertisement is done using a novel auction system, which we describe in this paper. With this auction system, it is possible to create an advertisement, attach keywords to it, and add it to the advertisement inventory. An advertisement is displayed on a search results page if the search keyword matches the keywords attached to the advertisement. Display of advertisements is done using a probabilistic ranking algorithm which determines the most relevant advertisements for the search query. Our system also supports the display of advertisements from external advertisement servers.

ACKNOWLEDGEMENTS

I would like to thank my advisor, Dr. Chris Pollett for his guidance. His remarks and suggestions throughout the project were really helpful. I also express my gratitude to Dr. Sami Khuri and Dr. Ronald Mak for their time and advice. Finally, thanks to my family and all persons involved directly or indirectly for their support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 Introduction

Online advertising is growing rapidly, as such advertisements can potentially reach a targeted audiences anywhere in the world [23]. Search engine marketing is one type of online advertising where advertisements are placed on web pages that are shown on the basis of search queries [18]. The rendered advertisements are targeted to match key search terms entered in search engines. This targeting ability has attracted many advertisers to search engine marketing. Many Online advertising platforms involve auction systems and advertising services, which help to price and place advertisement copy.

Online advertising platforms are used by advertisers to create their promotional campaigns for their products [1]. If advertisers use these auctioning systems correctly, they avoid wasting money advertising to people who have no interest in what they are selling. Advertisers are only charged if a user clicks on the advertisement. Advertising service is used for placing advertising copy at the ad spaces available on the search results page.

One prominent revenue source for most modern search engines, such as Google, Yahoo, and Bing, is to display advertisements. Advertisements constitute over 80% to 90 % of total revenue for these search engines [24] [25]. Sophisticated algorithms are used to compute the relevance of these ads to an entered search query.

People who own websites can display ads on their websites with the help of advertisement placing service [8]. They can request advertisement server to display targeted advertisements on a websites based on content. If an end user clicks on these advertisements, the advertisement placing service shares revenue with the website owner. It is an easy way to earn money for website owners.

In this project, we implemented an open source advertisement server that is integrated with Yioop. It mainly includes an online auction system and advertising service. Business users can create their campaign with the help of the online auction system to reach a targeted audience. With the help of an advertising service, advertisements from other advertisement servers can be displayed within Yioop.

The initial part of the project was to study existing online advertising systems. We mostly looked into Google's advertising program and the open source advertisements sever known as AdSpeed [19] [11]. Google AdWords limits the number of impressions and clicks per day based on the user's budget. The same is the case with AdSpeed. For example, if an advertiser has created a campaign but wants it to be activated after a week, in AdWords, the advertiser creates a campaign and pauses it for a week, but then the advertiser has to again return to change its status to active. In AdSpeed, as well, advertisers need to return to change the status. In this case, users may have to do frequent activation and deactivation for campaigns. Google also looks at the quality of the landing page the advertisement points to as it relates to what is in the advertisements [20]. If an advertiser is still working on the landing page, the campaign may get delayed.

With this new advertisement system we are developing, users will have the option of setting a start date and end date for the campaign. They do not have to return for frequent activation and deactivation. Also, with the current implementation, the number of impressions is not limited by budget. While retrieving advertisements from the inventory, the new advertisement platform uses a probabilistic model in addition to budget to select advertisements. Hence, even if a user has a limited budget, the platform can display the advertisement on the search results page based on random selection from a set of relevant advertisements.

2 Overview of Yioop

The project focuses on the design and implementation of a new online advertisement system. It is an online auction based system where business users create their campaign. These product or service campaigns are displayed on a web page based on a certain strategy. Search engine marketing, where advertisement is displayed on the search result page, is a major part of online advertising

Major search engines have integrated advertisement programs. Hence, we decided to integrate our new online advertisement system with Yioop. Yioop is an open PHP search engine. With the integration of an advertisement system, Yioop allows the display of contextual advertisements on the search results page. This advertisement system will evolve as a major source of revenue for Yioop.

3 Google's Online Advertising System

Most online information is "free" today because of advertisements shown on websites. Online advertising is ranked 2nd after TV advertising in terms of revenue [22]. Initially, advertisement space sold on a per-thousand-impression basis. But seeing an advertisement does not mean clicking on it or buying the product [17]. Hence, the per-thousand-impression pricing changed to a per-click basis.

Google started their advertisement platform in 2002, known as Google AdWords [19].

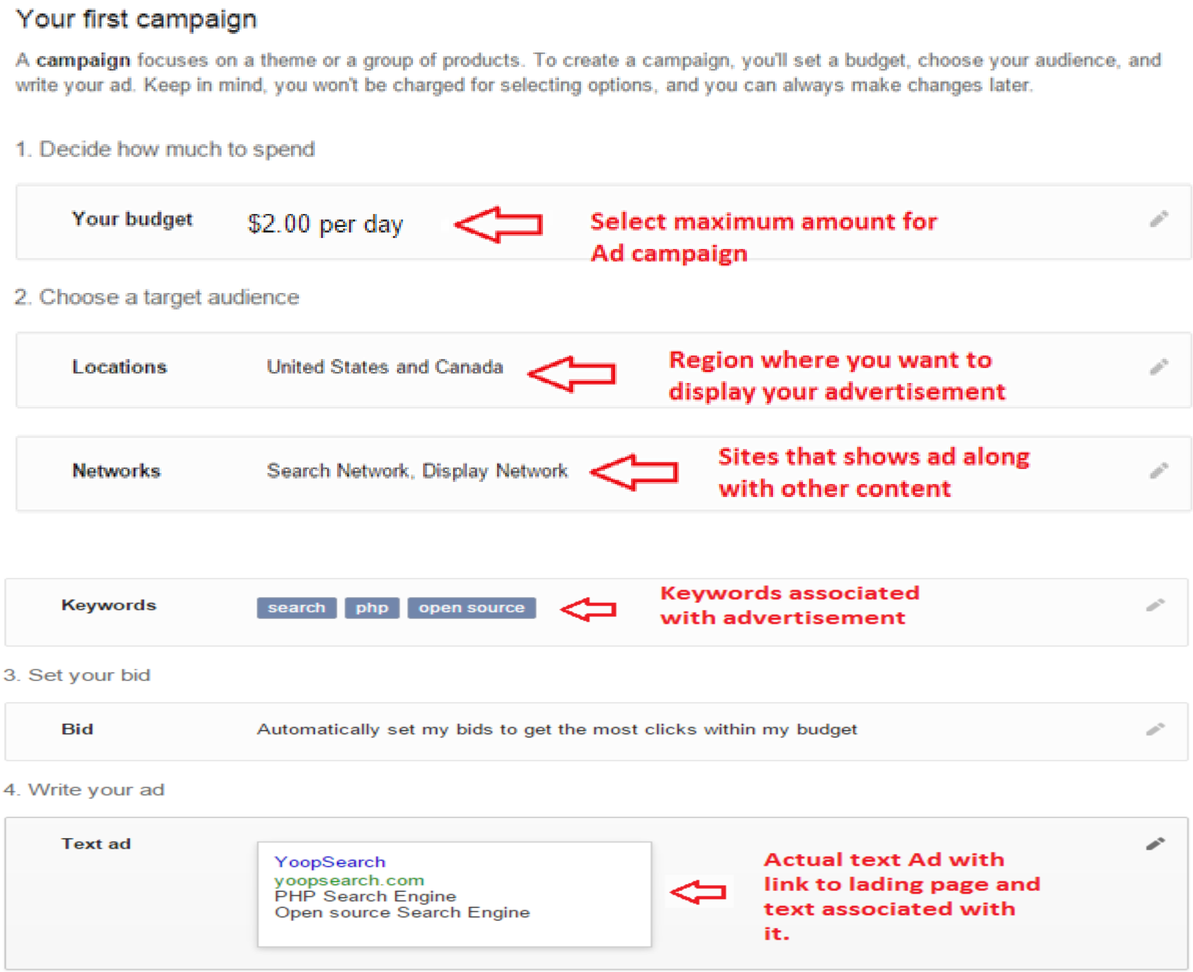
3.1 Google AdWords

AdWords is Google's online advertising platform and it places advertisement copy in the space available on Google's search results page and helps advertisers drive targeted users to their website. AdWords allows users to create advertisements and choose when they want them to appear next to relevant search results [6]. While creating advertisement campaign, users associate words that are relevant to their product or service. Then AdWords shows that advertisement on Google search results if someone searches for that or a similar word. AdWords also allows advertisers to choose geographical regions to display advertisements in if they want to target a local area. With AdWords, it is possible to display advertisements on thousands of sites across the web.

With AdWords, it is possible to manage your advertisement spending. Users can select the maximum amount that they are willing to pay, and they are only charged if someone clicks on the advertisement.

To make advertisement campaign successful, it is important to choose the right keywords, design a good landing page, and use attention grabbing words [6]. Keywords are the words that advertiser think people will search for in order to find the required search result. It is also important to choose a well suited landing page when users click on the displayed advertisement.

AdWords offers pay-per-click (PPC), that is, cost-per-click (CPC), cost-per-thousand-impressions or cost per mille (CPM) advertising, site-targeted advertising for text, banner, and rich-media ads, and remarketing [1]. The AdWords program includes local, national, and international distribution. Google's text advertisements are short, consisting of one headline of 25 characters, two additional text lines of 35 characters each, and a display URL of 35 characters. Image advertisements can be one of several different Interactive Advertising Bureau (IAB) standard sizes [19].



Your first campaign

A **campaign** focuses on a theme or a group of products. To create a campaign, you'll set a budget, choose your audience, and write your ad. Keep in mind, you won't be charged for selecting options, and you can always make changes later.

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

Keywords search php open source

Keywords associated with advertisement

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 lading page and text associated with it.

Figure 1 : Creating your advertisement Campaign

3.2 Google AdSense

Google AdSense allows website owners to earn revenue from their online content. AdSense works by matching images and text to the owner’s site based on content and visitors. Website owners can choose the size and color that best matches their website [8].

AdSense publishers must agree to the terms and conditions in order to display AdSense advertisements. Publishers are not allowed to click their own advertisements or encourage others to do so. Also, they should not place advertisements on sites that include incentives for users to click on advertisements [8]. Google may terminate use of AdSense services due to invalid clicks or impressions or other violations of the program policies.

Created advertisements are managed, arranged, and maintained by Google, and they can generate revenue on either a per-click or per-impression basis [19]. AdSense's innovative targeting options allow advertisers to more precisely reach their desired audiences on a site, which includes contextual targeting, placement targeting, and internet based targeting. AdSense has a variety of advertisement formats such as mobile web pages, videos, online games, TV inventory, feeds and site search results [8].

Advertisers bid for advertising space and the highest paying advertisement is displayed on owner’s website based the on content of the website. To display advertisements, website owners place a Javascript code provided by AdSense on web pages. Each time this page is visited, the script makes use of inline JSON to render content fetched from the Google server. As Javascript is executed in the web browser, it is possible for website owners to make use of the AdSense program by adding Javascript to their own web pages [8]. Search advertisements are rendered after the end user performs a search and added, along with list a of search results. To maximize revenue, use of valuable text content is encouraged in order to maximize the number of visitors clicking on advertisements.

Advertisements displayed by Google AdSense can contain an advertising icon in the corner of the advertisement. It is known as AdChoices. It is not an online advertising provider but a program for users targeted by advertising. It is used for interest based advertising. When this icon is available on an advertisement, it means information about the user's interest is being gathered to improve advertisements displayed for that user [21]. It helps to show ads based on the individual's interests.

3.3 How Google Sell Advertisement Spaces

The question may arise in a publisher's mind, where will my advertisement emerge on the page? Google uses an advertisement space allocation strategy for allocating advertisement space for advertisements [1]. Allocation of advertisement spaces happens according to some auction rules. There is only one seller, which is Google, and many advertisers and many advertisement spaces. Each advertiser can bid for ad spaces, and Google will allocate the ad space following some rules. The process is known as an auction.

The auction rules are designed in such a way that advertisers bid for the true valuation of the advertisement spaces.

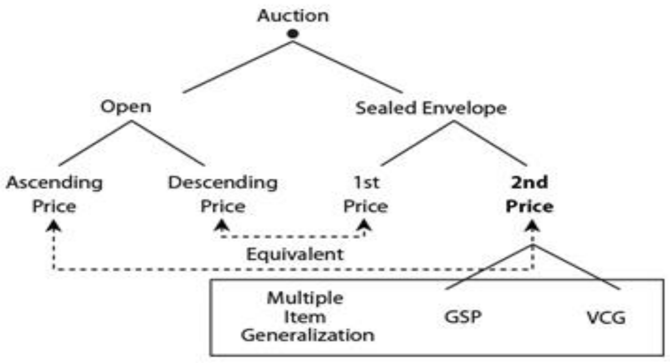


Figure 2 : Auction methods [1]

In most of the auction rules, the highest bidder gets the most valuable advertisement space. Auction methods are mainly divided into two, open and sealed envelope auctions.

Open auctions are conducted at a public venue where advertisers announce their bids. The two types of open auction are ascending price auctions and descending price auctions. In an **ascending price auction**, an auctioneer comes up with base price i.e. a minimum bid, and then each bidder can bid for a higher price. The auction continues until one bidder submits a price and no other bidder goes above. The last bidder wins and pays the last announced bid price. In **descending price auction**, an auctioneer starts with a high price. The price is meant to be so high that no bidder would be willing to accept it. The auctioneer then lowers the bid amount until one of bidders agrees to pay the announced price. This bidder wins and pays the last price announced by the auctioneer [1].

An alternative to an open auction **is a sealed envelope auction**. In this type of auction, each bidder submits bid privately. All bids are revealed to the auctioneer who then decides the allocation and charges for each advertisement space. The highest bidder gets the advertisement space. The amount charged depends on which auction method is used, first price or second price auction.

In a **first price auction**, the winner pays the highest bid. In a **second price auction**, the winner pays the second highest bid. A second price auction can work in the wrong way. If a bidder knows that he or she is paying the second highest bid, the bidder can bid an extremely high price. In that case, the bidder wins and pays a much lower price for the advertisement space. But the assumption of "much lower prices being bid by other bidders" does not hold when everyone engages in the same strategic thinking.

In practice, a **generalized second price** (GSP) auction is used, as there are multiple advertising spaces and advertisers. If the number of advertising spaces and the number of advertisers are different, then some advertisers won't even get an advertising space [1] [15].

In a generalized second price auction, the highest bidder gets the most valuable advertising space. This advertiser is then left out from future bidding. The price is charged using the second price approach.

- **Multi-item GSP auction in Google :**

  Imagine there are two advertising spaces on a web page and three bidders. Let's say, the first advertising space receives 400 clicks per day and the second ad space receives 200 clicks per day.

Table 1 : Advertising space with clicks/day [1]

Advertising Space	Clicks / day
1	400
2	200

Table 2 : Bidder competing for advertising spaces [1]

Value per click in USD	Bidder	Advertising space	Valuation
12	1	1	4800
		2	2400
8	2	1	3200
		2	1600
4	3	1	1600
		2	800

From Table 2, three bidders have values of $12, $8, and $4 respectively. If they bid truthfully, then their bids for each advertising space will be those under the valuation column. According to this strategy, Bidder 1 wins the first advertising space and pays $8 per click. Similarly, bidder 2 wins the second advertising space, paying $4 per click. Hence, their payoffs are $1600 and $800 respectively.

Truth-telling is not a dominant strategy for GSP. For example, imagine the first advertising space receives 400 clicks a day, and the second receives 300 clicks per day. If all bid truthfully, then bidder 1's payoff, is ($12 − $8) X 400 = $1600.

Table 3 : Clicks / day updated for GSP auction

Advertising Space	Clicks / day
1	400
2	300

Table 4 : Updated valuation for GSP auction

Value per click in USD	Bidder	Advertising space	Valuation
7	1	1	2800
		2	2100
8	2	1	3200
		2	2400
4	3	1	1600
		2	1200

From Table 4, let us assume Bidder 1 bids $7 per click for second advertising space, the payoff will be ($12 − $4) X 300 = $2400, which is more than $1600. Hence, Bidder 1 receives a higher payoff. The difference between the first and second advertising spaces click through rates is simply not large enough relative to the difference in per-click payment to bid truthfully.

As GSP does not guarantee a truthful auction, we will use a **Vickrey-Clarke-Groves** (VCG) auction. In VCG, matching is computed by the seller for optimization [14]. Consider, the following example where there are 3 bidder and 3 ad spaces.

Table 5 : Advertising spaces with click through rate for VCG auction [1]

Advertising Space	Click through rate
1	5
2	3
3	1

Table 6 : VCG Auction – Bidders and their bid amounts

Bidder	Value per click in USD
1	10
2	5
3	1

Table 7 : Bidders with bid amounts for VCG type of auction

Bidder	Advertising Space	Valuation
1	1	50
	2	30
	3	10
2	1	25
	2	15
	3	5
3	1	5
	2	3
	3	1

In VCG, we do matching such that we get maximum valuation for the system. For example, a different matching of {(1, 1), (2, 2), (3, 3)} would return a total valuation 50 + 15 + 1 = 66 which is the maximum [1].

4 Advertisement Placing Service in Yioop

4.1 Overview

An advertisement placing service is used to display contextual advertisements with content. With search engine marketing, these advertisements are displayed next to search results. Advertisement placing services allow advertisements from an external advertisement server to be displayed on a web page. It is an easy way to make money, as the advertisement service shares revenue with the website owner [16].

In Yioop, we are providing a configurable option for an advertisement placing service. Hence, Yioop can render contextual advertisements from an external advertisement program as well as advertisements from its own advertisement program. This is beneficial to the user, as we are rendering advertisements from more than one advertisement network on the same web page.

4.2 Implementation details

To render advertisements from an external source, we need to place a Javascript code provided by that resource. When the page loads, the Javascript code requests a contextual advertisement response from an external advertisement network.

In Yioop, we have provided configuration options for advertising space and advertisement placing service. The Yioop administrator has to select ad space within Yioop. It is the location where the advertisement from the external resource would be displayed. The administrator then needs to place a Javascript code provided by the external advertisement network for the selected advertising space.

Yioop has this configuration option under the server settings [4]. Currently, we have three options for advertisement alignment, top, side, and both. Users need to place Javascript code provided by the external advertising resource based on the selected alignment. For example, if the user selects "Top" alignment, then Javascript should be pasted under the "Top Element Ad Script".

This Javascript is passed to the search controller and is executed during rendering the view. On execution, it requests the external advertisement program and waits for a contextual advertisement as a response. Search view displays the response and places it at the user selected location.

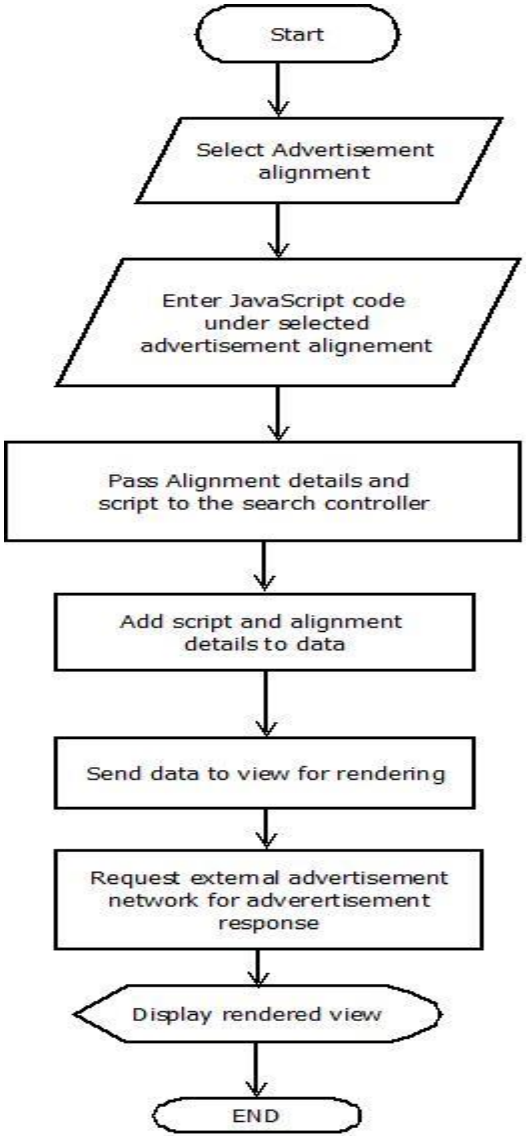
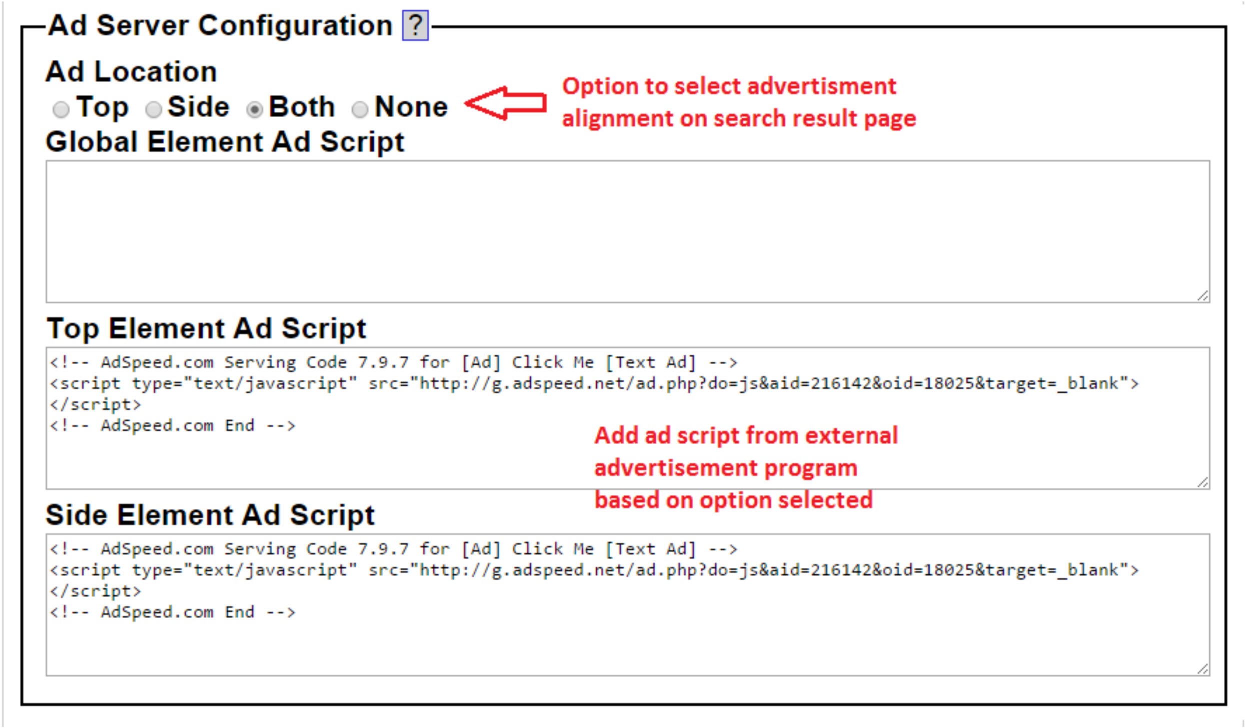

Start
Select Advertisement alignment
Enter JavaScript code under selected advertisement alignement
Pass Alignment details and script to the search controller
Add script and alignment details to data
Send data to view for rendering
Request external advertisement network for adverertisement response
Display rendered view
END

Figure 3 : Flow chart for advertisement placing service

To test the implementation, AdSpeed was used which is an open source high performance advertisement server for advertisers and publishers [11]. Users can login as an advertiser to AdSpeed and create an advertisement. Then we have to associate the newly created advertisement with available zones. Then AdSpeed would generate serving code for the respective advertisement [11].



Ad Server Configuration ?

Ad Location

○ Top ○ Side ● Both ○ None

Option to select advertisment alignment on search result page

Global Element Ad Script

Top Element Ad Script

```
<!-- 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>
<!-- AdSpeed.com End -->
```

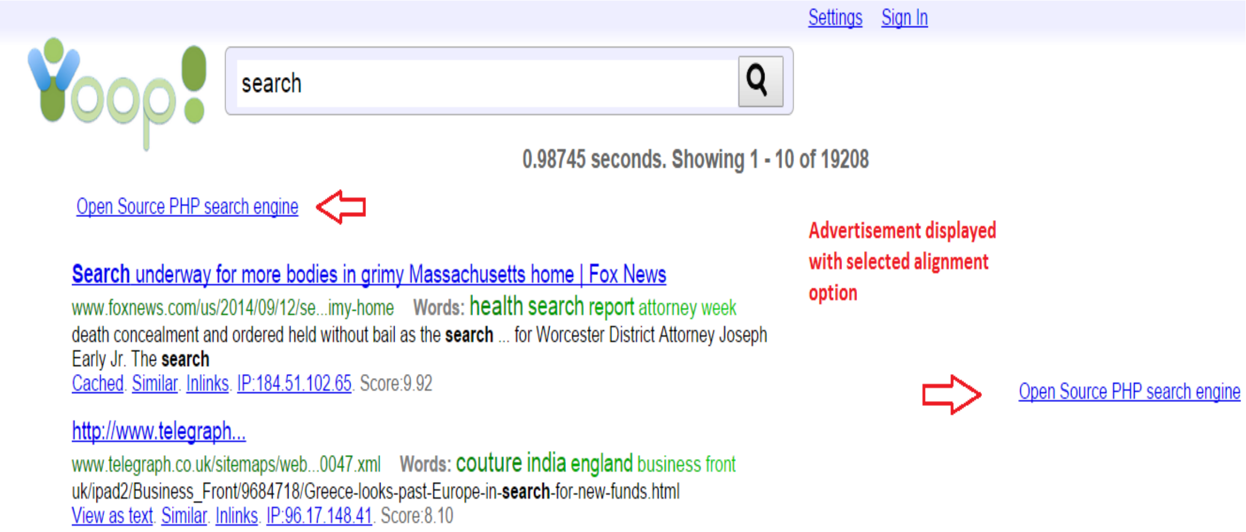
Add ad script from external advertisement program based on option selected

Side Element Ad Script

```
<!-- 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>
<!-- AdSpeed.com End -->
```

Figure 4 : Configuring the advertisement placing service in Yioop

This serving code is then added to Yioop as shown in Figure 4. We have selected “Both” in the alignment option. Hence, the script is added under both, the Top Element Ad Script and Side Element Ad Script. Yioop search engine then displays an advertisement from an external advertisement server using this script on the search results page.



0.98745 seconds. Showing 1 - 10 of 19208

Open Source PHP search engine

Advertisement displayed with selected alignment option

Search underway for more bodies in grimy Massachusetts home | Fox News
www.foxnews.com/us/2014/09/12/se...imy-home Words: health search report attorney week
death concealment and ordered held without bail as the **search** ... for Worcester District Attorney Joseph Early Jr. The **search**
Cached. Similar. Inlinks. IP:184.51.102.65. Score:9.92

Open Source PHP search engine

http://www.telegraph...
www.telegraph.co.uk/sitemaps/web...0047.xml Words: couture india england business front
uk/ipad2/Business_Front/9684718/Greece-looks-past-Europe-in-**search**-for-new-funds.html
View as text. Similar. Inlinks. IP:96.17.148.41. Score:8.10

Figure 5 : Search view of advertisement placing service

Figure 5 shows a sample advertisement displayed on Yioop's search results page. This advertisement was retrieved from the AdSpeed server with the help of a script that we added under the server settings of Yioop. With the "Both" option, we can see the advertisement is displayed at the top as well as a side banner. We have successfully added an advertisement placing service in Yioop.

5 Retrieving a Contextual Advertisement using Cosine Similarity

5.1 Overview

Search engines use a bag of words model to retrieve the documents that are most relevant to the entered query [12]. To rank these documents, they use a ranking algorithm like the cosine similarity ranking strategy. We extended this technique to retrieve contextual advertisements. To get context based advertisement we used words from the retrieved documents as an input to the ranking algorithm [3].

Users will click on an advertisement only if it is related to the user's interest. For example, if a user is searching for airline tickets, showing an advertisement for a restaurant is of no use. The best advertisements in this case, would include air ticket deals, airline company advertisements etc. Hence, displaying contextual advertisement is important.

5.2 Implementation details

5.2.1 Building bag of words model

We built a bag of words model using an inverted Index. An inverted index is a key data structure used for information retrieval by modern search engines [2].

An inverted index is an index data structure that stores the mapping of words to a set of documents containing those words. The inverted index for term *t* is a list of all documents that contain *t*. We identify each document by a document id which is a document serial number. We need variable size posting lists to store this kind of information. In memory, the posting lists can be represented as linked lists or variable length arrays. A posting list contains word and list of document ids containing that word. These postings are sorted by document ids.

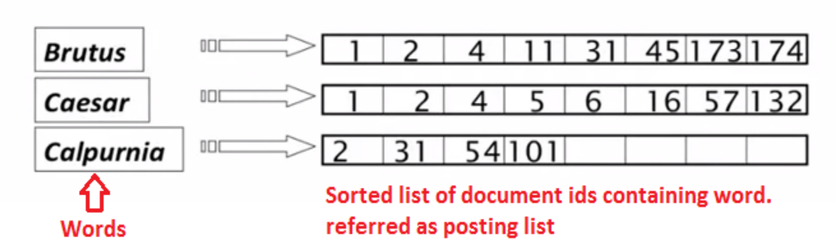


Figure 6 : Posting lists

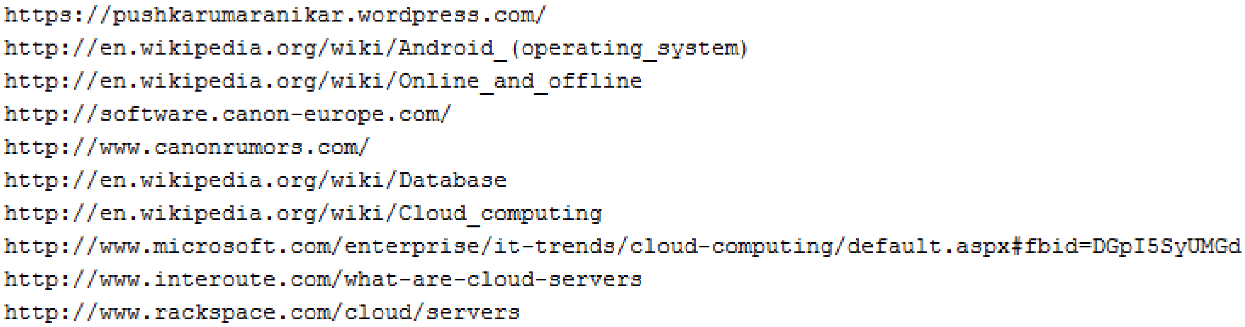
An inverted index is built by indexing a set of documents. These documents are generally a sequence of characters. Then a tokenizer is used to build a word token stream from a sequence of tokens. Words are then converted to some canonical form using linguistic modules. An inverted index is then built using an indexer [2]. But maintaining is not a standard solution as we need a lot of storage space if there are billions of documents. Also, it affects retrieval, since in the case of a conjunctive term we have to merge posting lists of multiple words. Hence, a positional index is used to build an inverted index.

In a positional index, we not only store a list of documents that contain the words but also the position of that word in each document where the term appears [2]. The format of a positional index is as below.

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

With a positional index, it is easy to deal with phrase queries. A positional index is a widely used model to build a bag of words as it is easy to deal with phrase queries with a positional index. In a bag of words model, terms from set of documents are indexed with positional index.

To build this bag of words model, we indexed documents from the below list of URL’s.



```
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/Database
http://en.wikipedia.org/wiki/Cloud_computing
http://www.microsoft.com/enterprise/it-trends/cloud-computing/default.aspx#fbid=DGpI5SyUMGd
http://www.interoute.com/what-are-cloud-servers
http://www.rackspace.com/cloud/servers
```

Figure 7 : List of URL’s for building bag of words model

We tokenized words from these URL’s and formed a bag of words model.

5.2.2 Cosine similarity

Cosine similarity is a measure of similarity between two vectors. It is obtained by calculating the inner product space of two vectors that measure the cosine of the angle between them. The cosine of 0° is 1 and is less than 1 for any other angle. It gives judgment of orientation. Two vectors having the same orientation have a cosine similarity of 1. Cosine similarity of two vectors is zero if they are at 90° from each other and if they are diagonally opposite, cosine similarity is -1. This judgment of orientation is independent of magnitude. Cosine similarity is generally used in positive quadrants or spaces where the value is between 0 and 1 [5]. For a query vector $\vec{q}$ and a document vector $\vec{d}$, when similarity is defined as the cosine of the angle between them, then :-

$$\text{sim}\,(\vec{d},\vec{q}) = \frac{\vec{d}}{|\vec{d}|} \cdot \frac{\vec{q}}{|\vec{q}|}$$

$|\vec{d}|$ Represents magnitude of document vector $\vec{d}$.

$|\vec{q}|$ Represents magnitude of query vector $\vec{q}$.

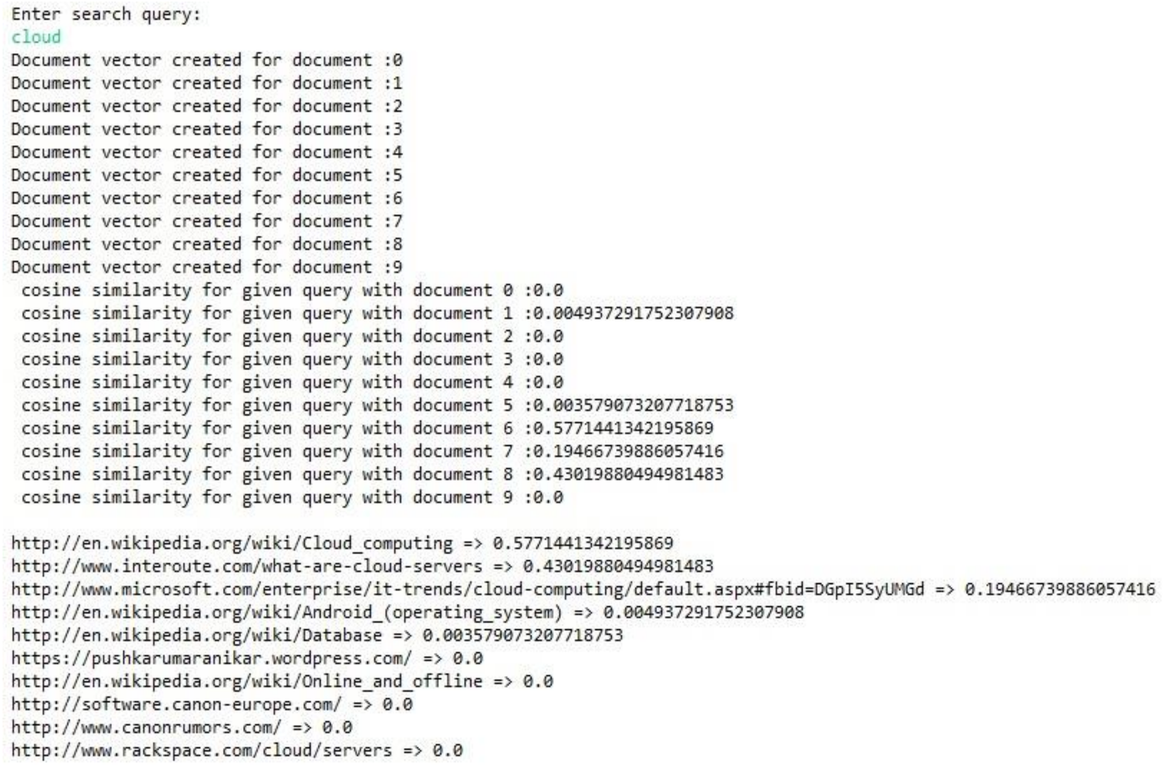
The cosine similarity measure is used in finding relevant documents for a given search query. For that, we have to build a query vector and document vectors for the entered query and list of documents respectively.

5.2.3 Finding Relevant Document for Entered Search Query

To implement the system, we first created a bag of words model based on documents available for indexing. The bag of words was model used to retrieve documents relevant to the search query using the cosine similarity ranking algorithm [12].

Cosine similarity can be applied to the query vector and document vector. Hence, we built a query vector and a document vector for every document in the document list. Let us consider a user entered query, “cloud”. We built a query vector for the query “cloud” using a bag of words model. We then applied the cosine similarity algorithm for the query vector and every document vector for documents from the list. If the result of the cosine similarity is close to 1, then we can say that the document is relevant to the entered search query. In our case, the following is the list of documents that are relevant to the entered search query “cloud.”

Figure 8 below shows outcomes of cosine similarity for every document with respect to the entered search query. Documents having outcome more than 0 and close to 1 are then returned as relevant documents. The document having the highest cosine similarity among all of them is the most relevant to the entered query. From Figure 8, we observe that a Wikipedia link to Cloud Computing is the most relevant to the query “cloud” among the list of documents that we considered.



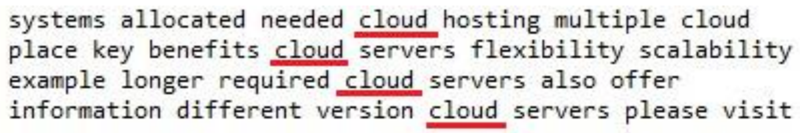
```
Enter search query:
cloud
Document vector created for document :0
Document vector created for document :1
Document vector created for document :2
Document vector created for document :3
Document vector created for document :4
Document vector created for document :5
Document vector created for document :6
Document vector created for document :7
Document vector created for document :8
Document vector created for document :9
 cosine similarity for given query with document 0 :0.0
 cosine similarity for given query with document 1 :0.004937291752307908
 cosine similarity for given query with document 2 :0.0
 cosine similarity for given query with document 3 :0.0
 cosine similarity for given query with document 4 :0.0
 cosine similarity for given query with document 5 :0.003579073207718753
 cosine similarity for given query with document 6 :0.5771441342195869
 cosine similarity for given query with document 7 :0.19466739886057416
 cosine similarity for given query with document 8 :0.43019880494981483
 cosine similarity for given query with document 9 :0.0

http://en.wikipedia.org/wiki/Cloud_computing => 0.5771441342195869
http://www.interoute.com/what-are-cloud-servers => 0.43019880494981483
http://www.microsoft.com/enterprise/it-trends/cloud-computing/default.aspx#fbid=DGpI5SyUMGd => 0.19466739886057416
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
```

Figure 8 : Results of cosine similarity for entered search query “cloud”

5.2.4 Retrieving Relevant Advertisements for Entered Search Query

We used cosine similarity for retrieving relevant documents for a given search query. We used a similar ranking strategy to get contextual advertisements. Now, we have a set of documents that are the most relevant. We used a subset of these documents to build a document vector for ranking advertisements based on cosine similarity.

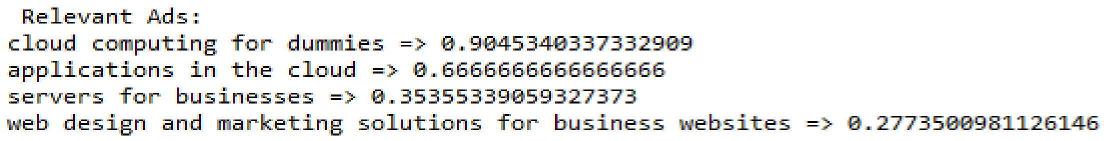
A set of documents is built using the retrieved documents. Each of these documents is formed with the entered keywords and the immediate three words associated with it on either side [3]. In the above case, we parsed through the retrieved documents to locate “cloud” and formed documents containing “< 3 words > cloud <3 words>” as shown in Figure 9.



```
systems allocated needed cloud hosting multiple cloud
place key benefits cloud servers flexibility scalability
example longer required cloud servers also offer
information different version cloud servers please visit
```

Figure 9 : Document set for query “cloud”

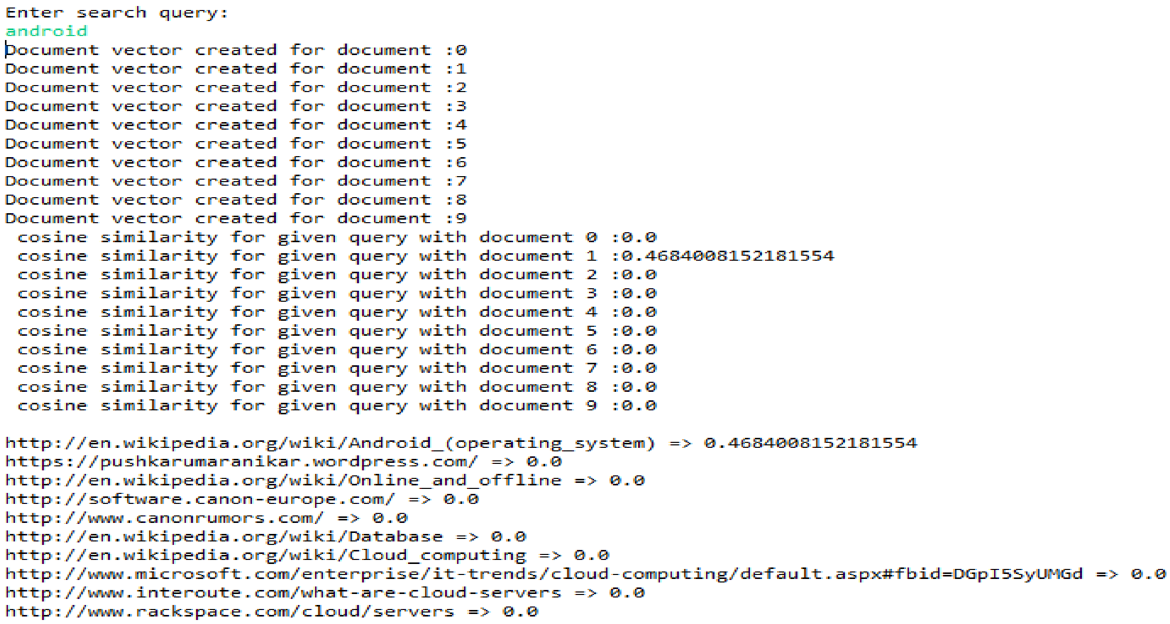
Then, using this list of documents, we calculated cosine similarity for each of these documents with advertisements in the inventory by forming vectors. The cosine similarity ranking algorithm will give the most relevant ads for a given search query. The advertisement vector is formed using keywords associated with every advertisement. Figure 10 below shows the relevant advertisements from the advertisements inventory that are most relevant to the entered query “cloud”. The cosine outcome for each advertisement is also associated with the advertisement name in the Figure.



```
 Relevant Ads:
cloud computing for dummies => 0.9045340337332909
applications in the cloud => 0.6666666666666666
servers for businesses => 0.35355339059327373
web design and marketing solutions for business websites => 0.2773500981126146
```

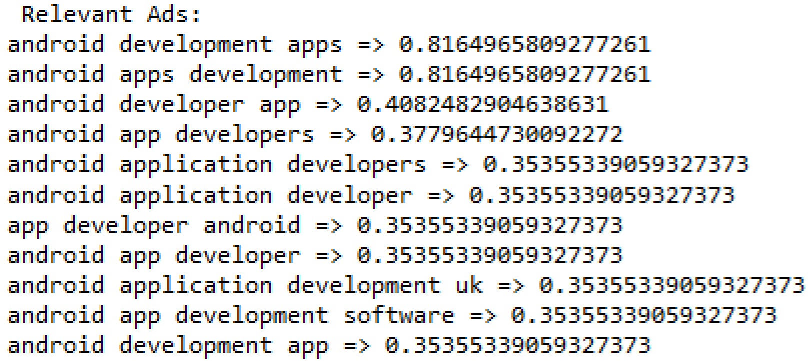
Figure 10 : Relevant advertisements for entered query “cloud”

Here is another example search query: “Android”



```
Enter search query:
android
Document vector created for document :0
Document vector created for document :1
Document vector created for document :2
Document vector created for document :3
Document vector created for document :4
Document vector created for document :5
Document vector created for document :6
Document vector created for document :7
Document vector created for document :8
Document vector created for document :9
 cosine similarity for given query with document 0 :0.0
 cosine similarity for given query with document 1 :0.4684008152181554
 cosine similarity for given query with document 2 :0.0
 cosine similarity for given query with document 3 :0.0
 cosine similarity for given query with document 4 :0.0
 cosine similarity for given query with document 5 :0.0
 cosine similarity for given query with document 6 :0.0
 cosine similarity for given query with document 7 :0.0
 cosine similarity for given query with document 8 :0.0
 cosine similarity for given query with document 9 :0.0

http://en.wikipedia.org/wiki/Android_(operating_system) => 0.4684008152181554
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://en.wikipedia.org/wiki/Database => 0.0
http://en.wikipedia.org/wiki/Cloud_computing => 0.0
http://www.microsoft.com/enterprise/it-trends/cloud-computing/default.aspx#fbid=DGpI5SyUMGd => 0.0
http://www.interoute.com/what-are-cloud-servers => 0.0
http://www.rackspace.com/cloud/servers => 0.0
```

Figure 11 : List of relevant documents for search query “Android”



```
 Relevant Ads:
android development apps => 0.8164965809277261
android apps development => 0.8164965809277261
android developer app => 0.4082482904638631
android app developers => 0.3779644730092272
android application developers => 0.35355339059327373
android application developer => 0.35355339059327373
app developer android => 0.35355339059327373
android app developer => 0.35355339059327373
android application development uk => 0.35355339059327373
android app development software => 0.35355339059327373
android development app => 0.35355339059327373
```

Figure 12 : Relevant advertisement for the query “Android”

5.2.5 Source code details:

Bag of words data format:

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

Input:
Search query

Intermediate output:
documents relevant to search query

Output:
Ads relevant to the search query

Important source files:

BagOfWords.java : Responsible for building inverted index.

DcoumentVector.java : This class contains methods for building document vector.

ContentBasedAds.java : Responsible for retrieving ads relevant to the search query from ads inventory.

TermInvertedIndex.java : Domain class for inverted index.

StopWordsHelper.java : Responsible for removing stop words from the document text.

IRServiceHelper.java : This class contains helper methods for finding relevant ads.

BagOfWordsHelper.java : This class contains helper methods for building inverted index.

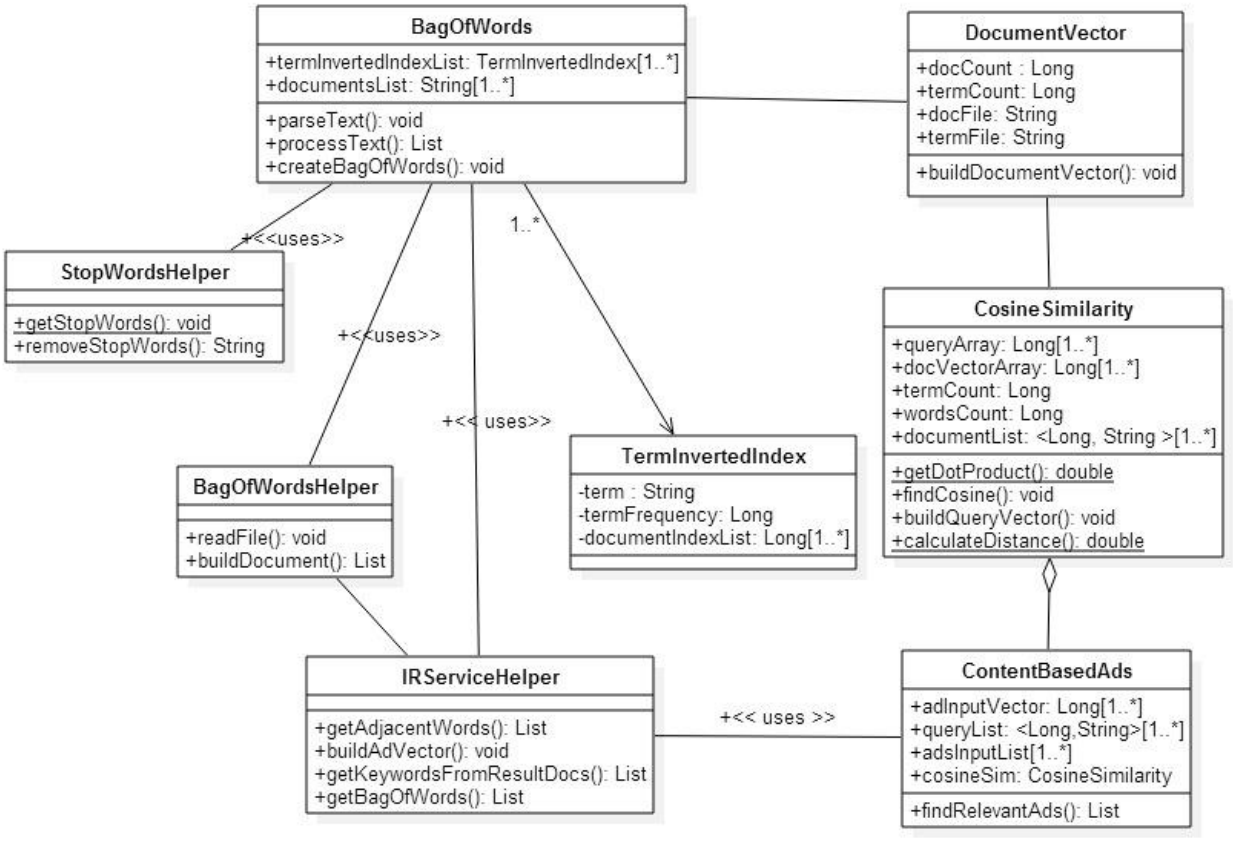


Figure 13 : Class diagram for Contextual advertisement retrieval using cosine similarity

Figure 13 shows important classes and the relationship among them. We built a BagOfWords model using DocumentVector and TermInvertedIndex. The ContentBasedAds class is used to retrieve contextual advertisements for a given search query using the cosine similarity ranking strategy.

From the results, we observe that we have successfully retrieved context-based advertisement for the given query using the cosine similarity ranking strategy.

6 Online Auction System in Yioop

6.1 Overview

Product or service promotion through search engines is an integral part of online advertising. It basically includes displaying contextual advertisements based on the entered query. Advertisements are displayed when search results are retrieved. To display advertisements from an inventory of advertisements programs use auction systems. Sophisticated algorithms are used to run these auction systems and makes sure that expected results are sent.

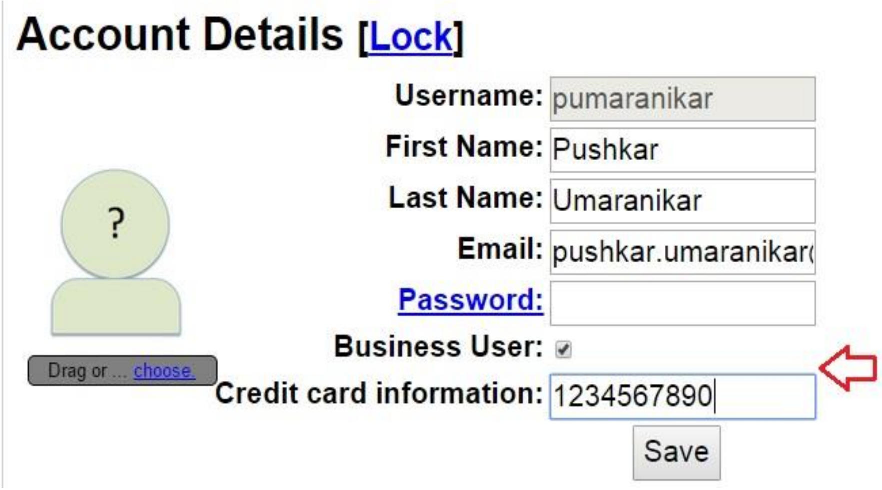
We integrated a new advertising system with Yioop to make use of search engine marketing. It allows users to create their campaigns. It includes an auction system where bids are accepted for every campaign. The advertising program then runs a ranking algorithm when users enter certain search queries. The ranking algorithm makes sure that only contextual advertisements are displayed on the search results page considering factors like budget and keyword association. It ensures display of advertisements that represent the user's interests.

6.2 Implementation details

6.2.1 New User Role for Accessing Online Auction System

Yioop is now integrated with an advertising program where users create advertisements, attach keywords to them and send them to Yioop's advertisement inventory. Yioop returns advertisements if the keyword matches the attached keywords.

The advertising program in Yioop is accessible only to particular users. For this, we have added a new user role to Yioop known as a "Business user". Anyone can become a business user. Yioop asks for credit card details in addition to other profile related details if someone requests a business user account. This configurable setting is available under the Manage Accounts section [4]. Users will get an option to become a business user under Edit Profile.



Account Details [Lock]

Username: pumaranikar

First Name: Pushkar

Last Name: Umaranikar

Email: pushkar.umaranikar(

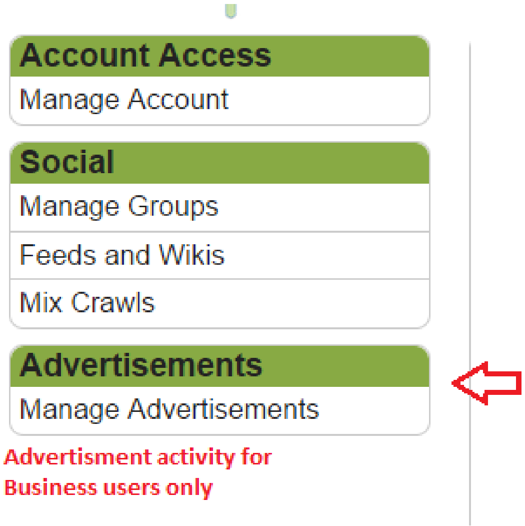
Password:

Business User: ☑

Credit card information: 1234567890

Figure 14 : Business user role configuration in Yioop

Users can see a new activity in the allowable activity panel on the left called Advertisements, after entering credit card information. Under this activity, the user has the option to "Manage Advertisements".



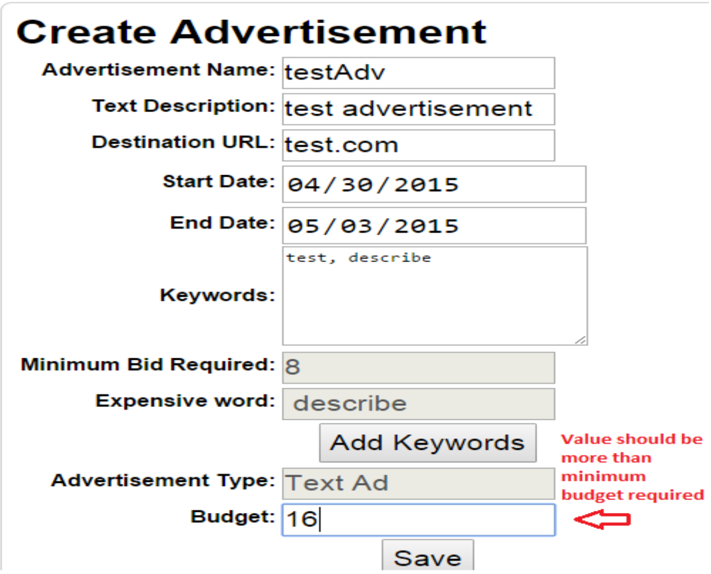
Advertisment activity for Business users only

Figure 15 : Advertisements activity for business users

6.2.2 Create Advertisement Form

The “Manage Advertisements” option under Advertisements navigates to the Create Advertisement form. The fields in the Create Advertisement form include name, description, destination URL, start date, end date, keywords and budget. In the Budget field, the user has to enter an amount that is equal to or greater than the minimum required budget. This minimum required budget value is calculated by the auction system depending upon the number of keywords and their popularity.

Along with the minimum budget required, the auction system displays expensive words among the keywords associated with an advertisement. It helps users to manage their budget. User can try minimizing their budget by removing expensive words from the keywords list. To get this information, the user has to enter the keywords list and click on the “Add Keywords” button on the form. An advertisement is created only if the user’s budget is greater than the minimum budget.



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

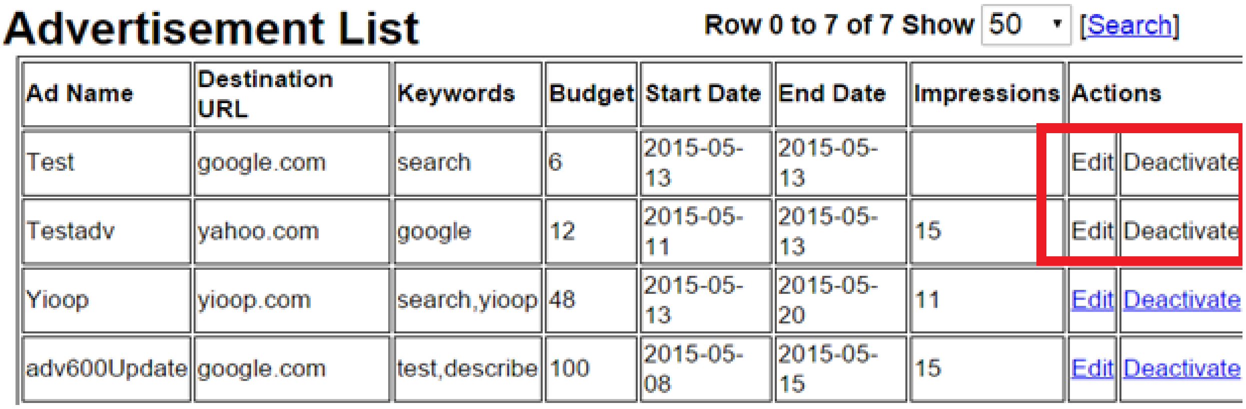
Advertisement Type: Text Ad

Budget: 16

Value should be more than minimum budget required

Figure 16 : Create advertisement form

The advertisement description is the text that would be displayed along with the name on the web page and would give more information about the advertisement. Destination URL indicates the landing page linked to when someone clicks on the advertisement. Start date and end date refer to the time period when this campaign remains activate. In Keywords section, the user can enter words that the user wants to associate with the advertisement. Keywords are the most important while displaying advertisements on the search results page.



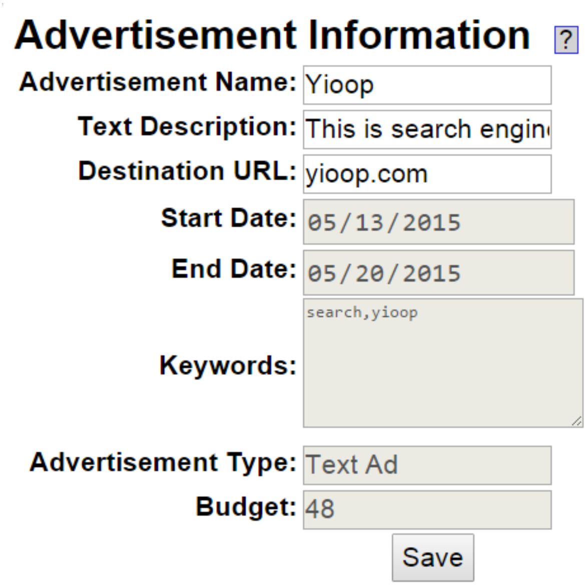
Advertisement List

Row 0 to 7 of 7 Show 50 [Search]

Ad Name	Destination URL	Keywords	Budget	Start Date	End Date	Impressions	Actions	
Test	google.com	search	6	2015-05-13	2015-05-13		Edit	Deactivate
Testadv	yahoo.com	google	12	2015-05-11	2015-05-13	15	Edit	Deactivate
Yioop	yioop.com	search,yioop	48	2015-05-13	2015-05-20	11	Edit	Deactivate
adv600Update	google.com	test,describe	100	2015-05-08	2015-05-15	15	Edit	Deactivate

Figure 17 : List of created advertisements by user

Figure 17 shows the tabular view of a created advertisements by the user. User can edit, deactivate and activate already created campaigns. Advertisements are automatically deactivated on the expiry date. Once, they an advertisement is deactivated, user cannot edit it.



Advertisement Information

Advertisement Name: Yioop

Text Description: This is search engin

Destination URL: yioop.com

Start Date: 05/13/2015

End Date: 05/20/2015

Keywords: search,yioop

Advertisement Type: Text Ad

Budget: 48

Figure 18 : Edit Advertisement Form

Figure 18 shows the Edit Advertisement form. Editable fields are name, text description and destination URL.

6.2.3 Understanding Database Schema for the Online Advertising System

The auction system must maintain an inventory of where advertisements created by users are stored. Advertisements from the same inventory are referred to while rendering contextual advertisements on the search results page. There are mainly three tables added to the Yioop database schema for the advertising and auction system.

- Advertisements Table
  - Adv_id: A system generated unique identifier for an advertisement.
  - User_id: This field gives information about which user has created this advertisement.
  - Adv_name: The advertisement name entered by user.
  - Adv_description: Description of the advertisement entered by user.

- Adv_destination: The field indicates the landing page when someone clicks on the advertisement.
- Adv_keywords: Comma separated list of keywords associated with advertisement
- Status: Indicates whether the advertisement is active or not.
- Adv_budget: Indicates the maximum amount that user willing to spend.
- Adv_date: Indicates the date when the campaign is active. We do not store the end date as for every day we store a different entry for the advertisement. For example, if the start date on the Create Advertisement form is April 30, 2015 and end date is May 1, 2015, then in the database we will have 2 separate entries with different adv_date fields. One entry is for April 30 and the other is for May 1.

- Advertisement_keyword

  It mainly contains keywords and associated bid amount with each keyword.

- Id: System generated unique identifier for each keyword.
- Keyword: Keyword associated with one of the advertisements.
- Bid_amount: Bid amount of the keyword.
- Bid_date: Indicates the date of a particular bid amount. As users can select start and end dates, we consider every day as a separate transaction. Hence, the bid amount for each keyword is different for different dates.

- Advertisement_metadata

  Table mainly contains metadata information regarding advertisement.

- Adv_id: A unique id associated with the advertisement. Must exist in the advertisements table.
- Clicks: Number of clicks on the advertisement
- Impressions: Number of times the advertisement has been rendered on a web page.

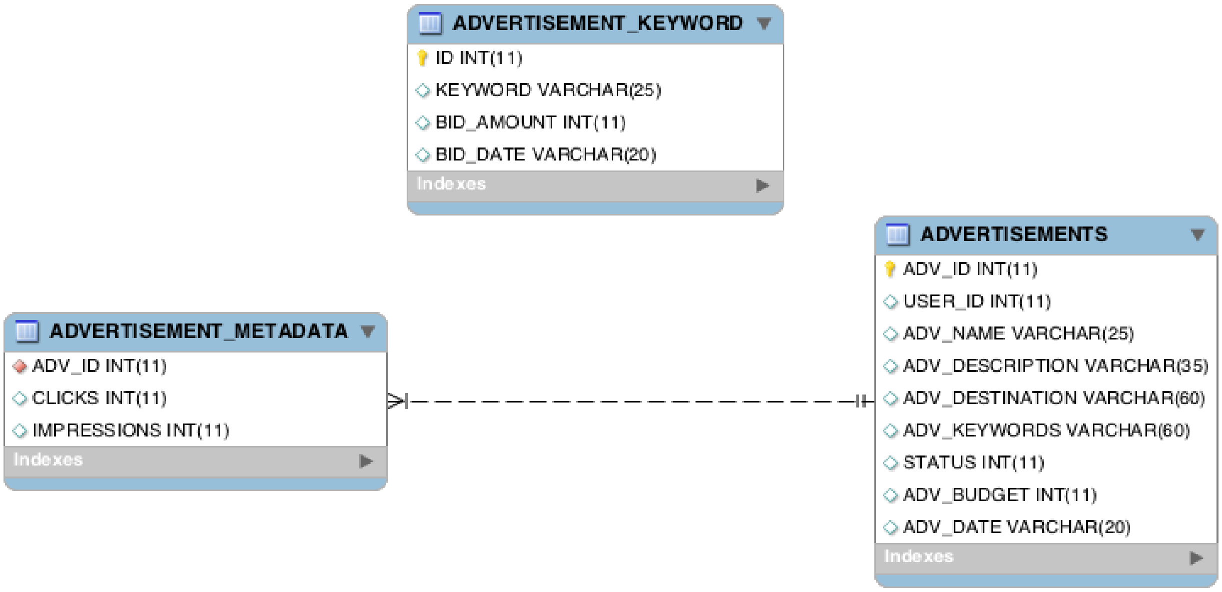


Figure 19 : Database schema for online auction system

6.2.4 Calculating Minimum Bid Amount

The auction system calculates the minimum bid required to create a new campaign based on the popularity of keywords. Every day is considered as a separate transaction for the same advertisement. If some keyword has not already been used before, then a minimum bid set is to 1 for that keyword. For the same keyword, the bid amount may vary for different dates as we are considering every day as a different transaction.

Consider following example.

A user is creating a new campaign called “Test Advertisement”. Keywords associated with it are “test” and “adv”. The user entered start date as May 1, 2015 and end date as May 7, 2015. Now, we will discuss how the auction system calculates minimum bid required to create this campaign.

As every day is a separate transaction, the bid amount associated with keywords “test” and “adv” for different dates may vary. Consider the following table describing the existing bid amount associated with each keyword. If for a given date, the keyword has not used been before then the minimum bid for that day is set to 1.

Table 8 : Calculating minimum bid required

Keyword	Bid Amount	Bid Date
Test	10	May 1
Test	8	May 2
Test	10	May 3
Test	12	May 4
Test	10	May 5
Test	8	May 6

Keyword	Bid Amount	Bid Date
Adv	8	May 1
Adv	12	May 2
Adv	10	May 3
Adv	12	May 4
Adv	8	May 5
Adv	4	May 6
Adv	12	May 7

Table 8 shows the bid amount for both keywords for the user entered time period. Using these bid amounts, the auction system calculates the minimum bid required. To calculate the minimum bid, we will half the existing bid amount for every day for every associated keyword.

For example, the bid amount for keywords test on for May 1, is 10. We will consider half the bid amount i.e. 5 for calculating minimum bid. For May 7, keyword test has not been used yet. Hence, the bid amount of “test” on that day is 1. We divide the bid amount by 2 so as to have logarithmic distribution of the total amount of money involved. Also, the minimum bid required would be high if we use bid values as it is.

The minimum bid for the word “test” = (10/2) + (8/2) + (10/2) + (12/2) + (10/2) + (8/2) + 1

= 5 + 4 + 5 + 6 + 5 + 4 + 1

= 30.

Similarly, the minimum bid required for word the “adv” is,

= 4 + 6 + 5 + 6 + 4 + 2 + 6

= 33.

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

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

6.2.5 Updating the Bid Amount for the Associated Keyword

In the above, when the user enters a budget greater than 63, the auction system creates a campaign. Let us consider when a user has entered a budget of 126. Now, we will see how the auction system updates the bid amount for associated keywords for a given time period.

The user has entered the start date as May 1 and the end date as May 7. Hence, the active time period of the campaign is 7 days. We have 2 associated keywords with the newly created campaign. The updated bid amount for the keyword is,

$$updateFactor = ((Bid\ of\ the\ day\ \div 2) \div \min bid\ required\ ) \times Budget$$

In the above case, Update factor of the keyword “test” on May 1 is,

= (5 /63) × 126

= 10.

Update factor for each keyword on a particular day is calculated by calculating fraction of the minimum bid required and we scale it to the budget of the user.

Hence, 10 is added to the existing bid amount of the keyword “test” for date May 1

Table 9 : Bid amount updated for associated keywords

Keyword	Bid Amount	Bid Date
Test	10 + 10 =20	May 1
Test	8 + 8 = 16	May 2
Test	10 + 10= 20	May 3
Test	12 + 12 =24	May 4
Test	10 + 10 =20	May 5
Test	8 + 8 =16	May 6
Test	2	May 7

Keyword	Bid Amount	Bid Date
Adv	8 + 8 = 16	May 1
Adv	12 + 12 = 24	May 2
Adv	10 + 10 =20	May 3
Adv	12 + 12 = 24	May 4
Adv	8 + 8 =16	May 5
Adv	4 + 4 = 8	May 6
Adv	12 + 12 = 24	May 7

6.2.6 Rendering Relevant Advertisements on the Search Results page

Now, we have an advertisement inventory that contains all created advertisements. When a user enters a search query, the job of the search controller is to retrieve the relevant advertisement from the advertisement repository in addition to the search results.

Let us imagine, the user has entered the search query as “computer”. The search controller matches the keyword “computer” and finds advertisements that have the keyword “computer” associated with them. Let us say we have 5 advertisements matching the given search query. Now, the question is which one to display on the search results page out of the 5.

The following table shows those 5 advertisements and the budget associated with each.

Table 10 : Relevant advertisements with their budget

Relevant advertisement	Budget
Adv1	10
Adv2	20
Adv3	30
Adv4	40
Adv5	50

We do a sum of the budgets of all advertisements. In this case:-

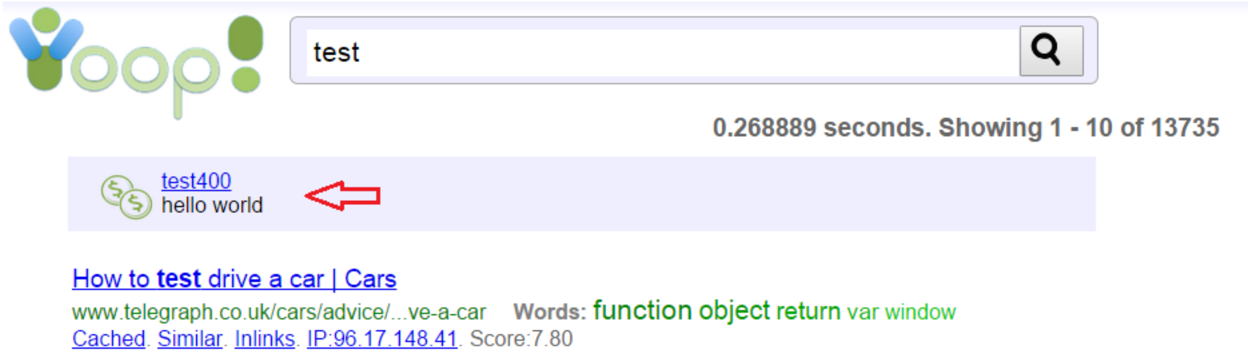
SumOfBudget = 10 + 20 + 30 + 40 + 50 = 150

This number is used as a max limit for a random number generator. Hence, in our case the rand function would generate a random number in the range of 0 to 150. Then for a random number, we check the budget range of each advertisement. We display an advertisement when it lies in the budget range of that advertisement. Consider, the following Table.

Table 11 : Relevant advertisements with defined budget range

Relevant advertisement	Budget	Advertisement range
Adv1	10	0 + 10 = 0 -10
Adv2	20	10+ 20 = 11 – 30
Adv3	30	31 – 60
Adv4	40	61 – 100
Adv5	50	101 – 150

From Table 11, we can observe that if the random number lies in the range 0-10, adv1 gets displayed on the search results page. If the random number lies in the range 11-30, Adv2 gets displayed and so on. We are using a probability based model to display advertisements so that even if the advertisement has a low budget but is relevant, it will be displayed but with lower probability than others. Hence, from the results above we conclude that we have successfully a retrieved contextual advertisement on the search results page.



test

0.268889 seconds. Showing 1 - 10 of 13735

test400
hello world

How to test drive a car | Cars
www.telegraph.co.uk/cars/advice/...ve-a-car Words: function object return var window
Cached. Similar. Inlinks. IP:96.17.148.41. Score:7.80

Figure 20 : Relevant advertisement on the search results page

6.2.7 Architecture of the Advertisement System

Figure 21 shows an architectural overview of the advertisement system in Yioop. It has a main controller that receives requests. Furthermore, the controller passes requests to one of the concrete controllers, in our case its AdminController. AdminController processes the request and initiates the component if it is a request related to the activity ManageAdvertisements. The request is then handled by the concrete component known as AdvertisementComponent. AdvertisementComponent handles actions like create advertisement, delete advertisement, edit advertisement etc.

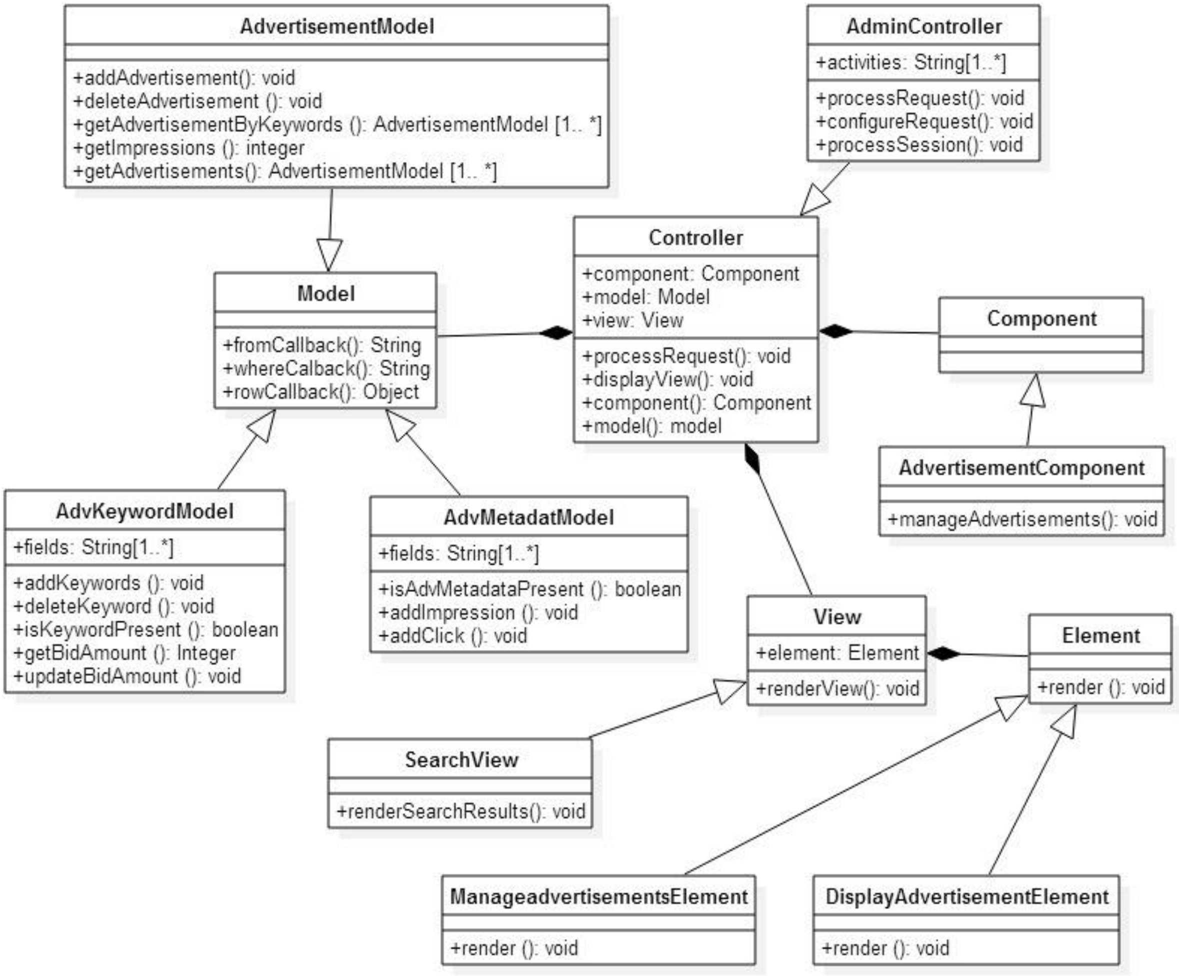


Figure 21 : Architecture of advertisement system in Yioop

AdvertisementComponent uses a model to perform database operations. We have three models in our advertisement system. The Advertisement model is responsible for actions related to campaigns like create, and edit. The AdvKeyword model records keywords related to information including adding and updating the bid amount, adding keywords, and checking if a keyword is already exists or not.

After performing operation on the model, the component uses a view to render the updated information. A view consists of many elements. In our case, we have ManageAdvertisementsElement, which renders the Create Advertisement form and displays existing advertisements. DisplayAdvertisementElement is rendered within search view, which actually places advertisement on the search results page.

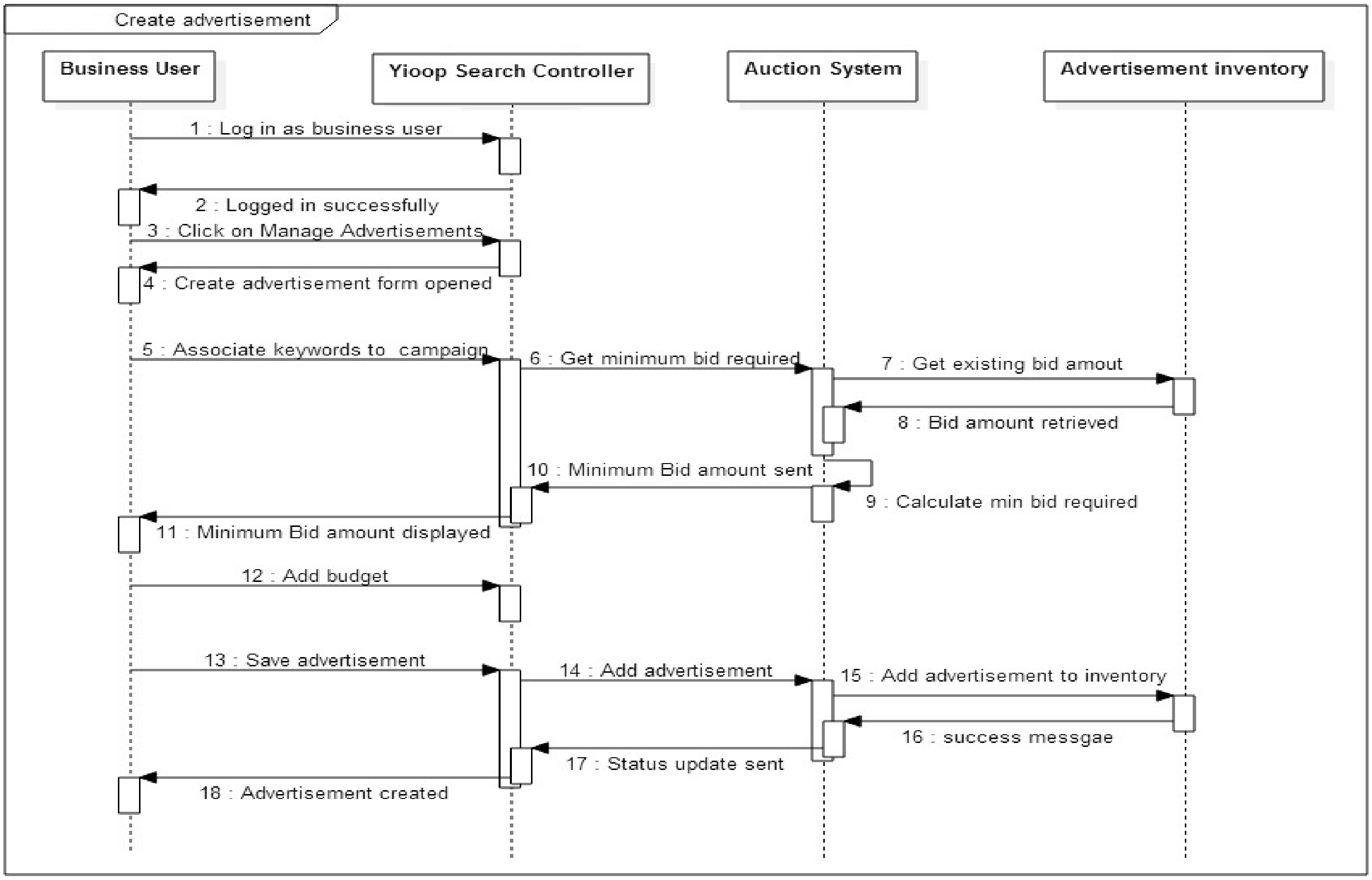

Create advertisement
Business User
Yioop Search Controller
Auction System
Advertisement inventory
1 : Log in as business user
2 : Logged in successfully
3 : Click on Manage Advertisements
4 : Create advertisement form opened
5 : Associate keywords to campaign
6 : Get minimum bid required
7 : Get existing bid amout
8 : Bid amount retrieved
10 : Minimum Bid amount sent
9 : Calculate min bid required
11 : Minimum Bid amount displayed
12 : Add budget
13 : Save advertisement
14 : Add advertisement
15 : Add advertisement to inventory
16 : success messgae
17 : Status update sent
18 : Advertisement created

Figure 22 : Sequence diagram for creating an advertisement

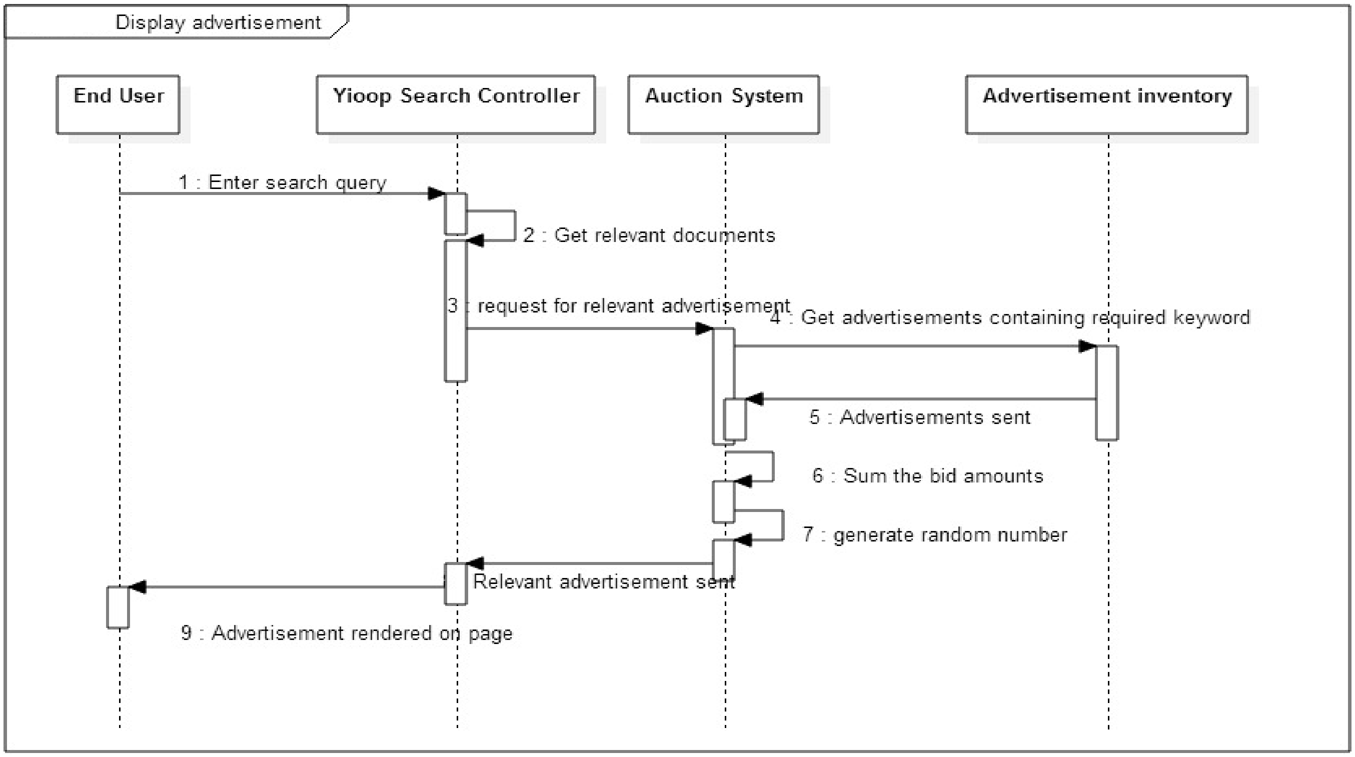

Display advertisement
End User
Yioop Search Controller
Auction System
Advertisement inventory
1 : Enter search query
2 : Get relevant documents
3 : request for relevant advertisement
4 : Get advertisements containing required keyword
5 : Advertisements sent
6 : Sum the bid amounts
7 : generate random number
Relevant advertisement sent
9 : Advertisement rendered on page

Figure 23 : Sequence diagram for displaying an advertisement

6.2.8 Comparison with Google AdWords

The new auction system is based on probabilistic ranking. This ensures that even if a user has a small budget for a campaign, advertisement displayed on the search results page. But it will have lower probability than other relevant advertisements. Google AdWords uses a generalized second price auction; the where most visited advertising place is allocated to the highest bidder. In this case, low price bidders may not get their advertisements placed [6]. In the new auction system, 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.

7 Experiments

After implementing an advertising system in Yioop, we performed some experiments to check its design and functionality. We did two types of experiments, load testing and user acceptance testing.

In load testing, we checked server response time when multiple users accessed the system at the same time. We did load testing using Apache Bench which is a benchmarking tool [10]. We also used Apache JMeter that generates multiple concurrent http requests and records average time required by the server to respond to each request [9].

We did tests with JMeter and Apache Bench by changing workload. The number of requests and concurrency were different in each case.

In user acceptance testing [13], we asked users to use the system. The aim of user acceptance testing is to check whether users can use the newly designed advertisement platform easily or not. When users were playing around with the system, we noticed difficulties faced by them. This may include bad user interface, improper help text or complicated work flow.

- Load Testing
    - Single User – multiple requests

Table 12 : Response time in seconds when single user sends multiple requests

Number of Requests	Response Time in seconds
10	6.4
25	12.11
50	23.98
100	49.02
200	98.82
500	234.17

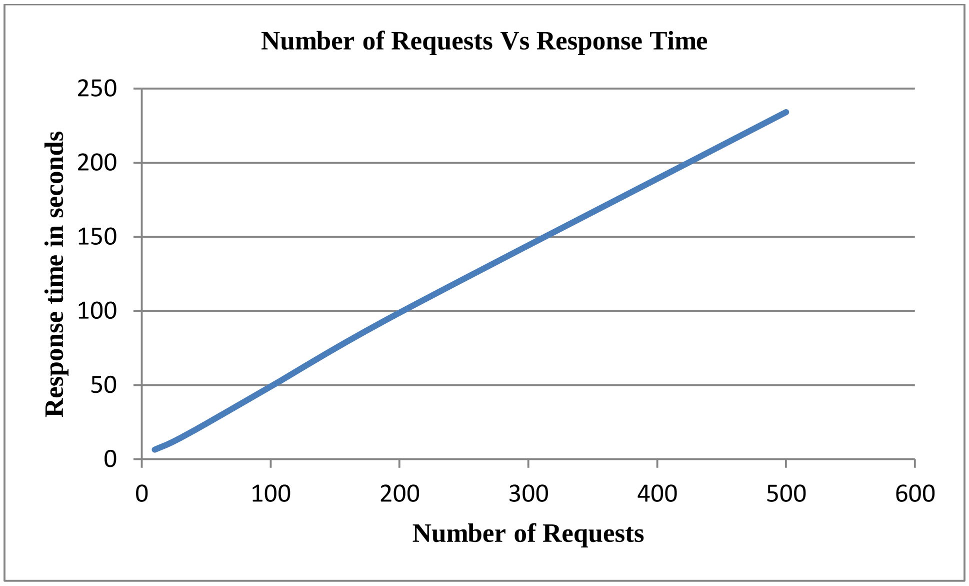


Figure 24 : Graph of multiple requests against response time for a user

From Figure 24, we can conclude that response time increases as number of requests increases.

- Multiple users accessing the system with multiple search queries.

Table 13 : Response time for multiple users with concurrent user requests

Users	Response Time in seconds
100	37.18
200	74.57
300	96.92
400	118.94
500	134.92
600	143.68

Table 13 shows the response time for the multiple users accessing the system with multiple search queries like “Cloud Computing”, “test”, “update” for which we have advertisements in the inventory.

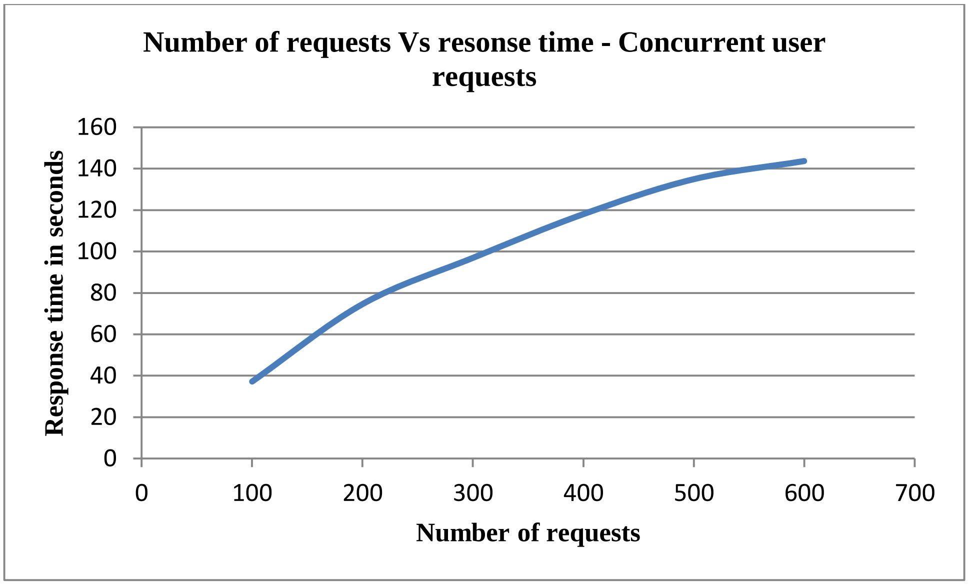


Figure 25 : Graph of multiple users with different search query requests

Now, let us check the response time when the advertisements are not enabled. We carried this test when the concurrent users accessing the system with multiple search queries.

Table 14 : Response time of concurrent requests when advertisements are not enabled

Users	Response Time in seconds
100	34.17
200	70.47
300	90.72
400	108.64
500	127.79
600	132.51

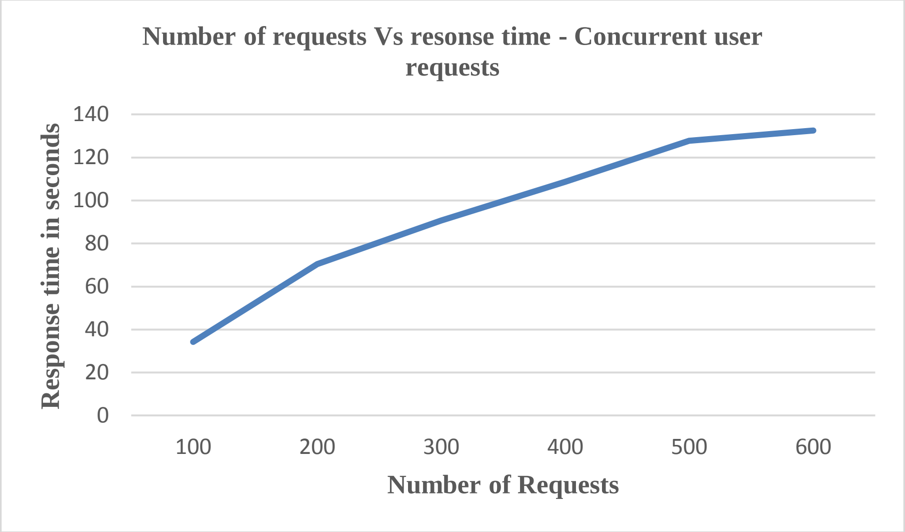


Figure 26 : Graph of concurrent requests when advertisements are disabled

From the Figure. 25 and Figure. 26, we can observe that there is no significant change in the response time of the search results when the advertisements are displayed on a search results page in addition to the relevant documents.

- User Acceptance Testing

  - User 1

Action: Create Advertisement

Time Taken: 2 min 30 seconds

Feedback received:

1. Add help text about how to enter the keywords.

  - User 2

Action:

1) Create Advertisement

2) Search for the query that matcher the keywords associated.

3) Click on the advertisement displayed.

Time Taken: More than 4 minutes

Problems faced:

- Asked about what is destination URL.

Feedback received:

1) It is useful to have a preview of the created advertisement.

  - User 3

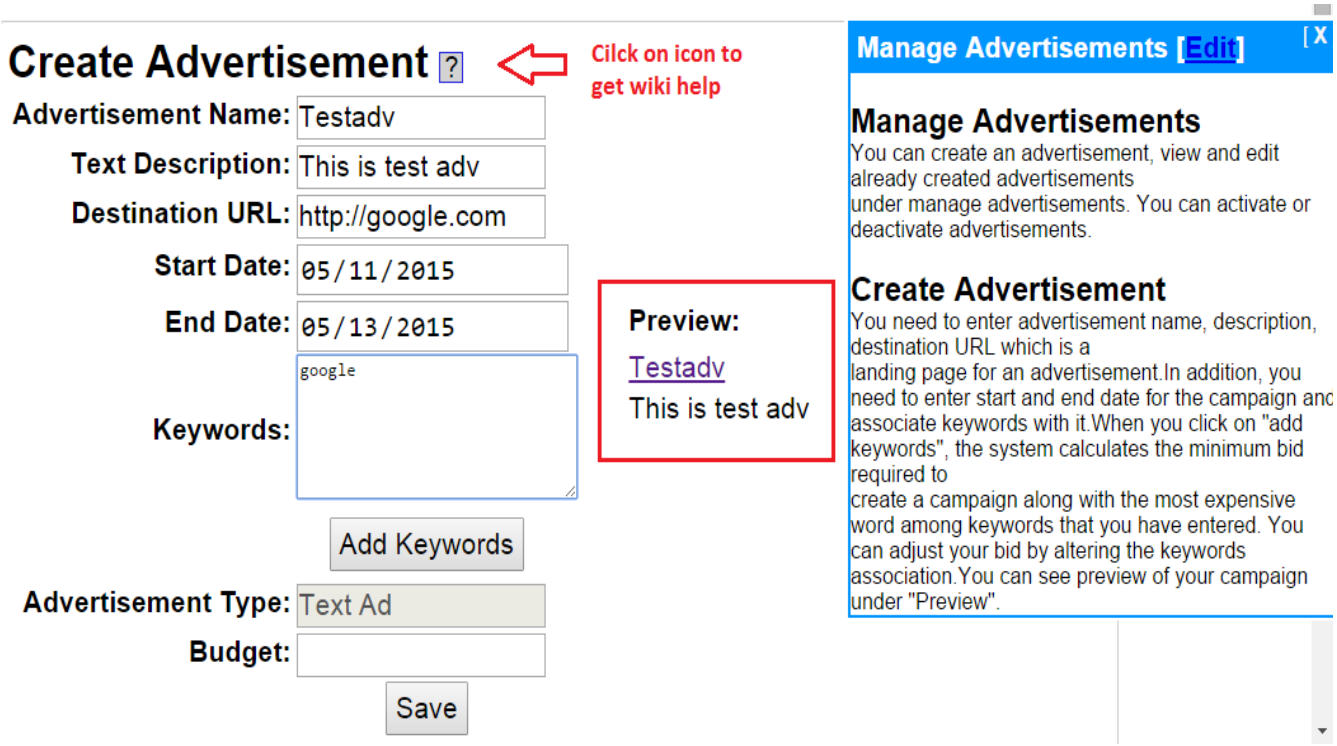
Action: Create Advertisement

Time Taken: More than 3 minutes

Feedback received:

1. Add help text about Add keywords button.

Preview and wiki help added on the create advertisement form after receiving a feedback.



Create Advertisement ?

Click on icon to get wiki help

Advertisement Name: Testadv

Text Description: This is test adv

Destination URL: http://google.com

Start Date: 05/11/2015

End Date: 05/13/2015

Keywords: google

Advertisement Type: Text Ad

Budget:

Preview:

Testadv

This is test adv

Manage Advertisements [Edit] [X

Manage Advertisements

You can create an advertisement, view and edit already created advertisements under manage advertisements. You can activate or deactivate advertisements.

Create Advertisement

You need to enter advertisement name, description, destination URL which is a landing page for an advertisement.In addition, you need to enter start and end date for the campaign anc associate keywords with it.When you click on "add keywords", the system calculates the minimum bid required to create a campaign along with the most expensive word among keywords that you have entered. You can adjust your bid by altering the keywords association.You can see preview of your campaign under "Preview".

Figure 27 : Advertisement preview and Wiki help on create advertisement form

8 Conclusion

We implemented a new online advertisement system to enable search engine marketing in Yioop. With our system, a business user can create an advertising campaign. These advertisements are displayed on the search results page. We studied Google AdWords to understand how online advertising works.

Our system mainly supports selling advertisements and displaying them within search results. We developed an auction system that allows the selling of advertisements. Newly created advertisements are added to the advertisement repository. We implemented a probabilistic ranking algorithm to display these advertisements on the search results page. A ranking algorithm displays the most relevant advertisements for the search query. Advertisements are ranked by matching search keywords with keywords attached to it.

We implemented an advertisement placing service that displays advertisements from external advertisement servers within the search page. An advertisement placing service uses a java script code provided by the external advertisement program to display advertisements. This service provides a configurable option for aligning advertisements within the search results page.

We also implemented a prototype for rendering the most relevant advertisements using a cosine similarity ranking algorithm. In this approach, documents are retrieved based on their relevance to the search keywords. Words from these retrieved documents are used as an input vector to find the most relevant advertisements from the inventory.

We did load testing and user acceptance testing on a new advertising platform. We tested system against maximum load of 500 users that are concurrently sending different search query requests. For user acceptance testing, we asked users to use the system and received feedback regarding usability, and user interface design.

In this project, I learned about online advertising, its working, and its importance in the world of advertising.

9 References

[1] Section 2 and section 5 from Networked life: 20 questions and Answers by Mung Chiang published by Cambridge University Press, New York.

[2] Information retrieval by Stefan Buettcher, Charles L. A. Clarke and Gordon V. Cormack published by the MIT press.

[3] Technical paper : Search advertising using web relevance feedback by Marcus Fontura, Lance Riedel and others http://gabrilovich.com/publications/papers/Broder2008SAW.pdf

[4] Yioop Documentation from SeekQuery. Retrieved on Apr 30, 2015, from http://www.seekquarry.com/?c=static&p=Documentation

[5] Wikipedia for Cosine Similarity Ranking. Retrieved on Apr 30, 2015, from http://en.wikipedia.org/wiki/Cosine_similarity

[6] Inside Google AdWords. Retrieved on Apr 30, 2015 from http://adwords.blogspot.com/

[7] Google AdWords – Pros and Cons Retrieved on Apr 30, 2015 from https://www.hallaminternet.com/2012/google-adwords-pros-and-cons/

[8] AdSense by Google Retrieved on Apr 30, 2015 from http://en.wikipedia.org/wiki/AdSense

[9] Apache JMeter Retrieved on Apr 30, 2015 from http://jmeter.apache.org/usermanual/index.html

[10] Apache http server project. Retrieved on Apr 30, 2015 from http://httpd.apache.org/docs/2.4/

[11] AdSpeed, open source high performance advertisement server. Retrieved on Apr 30, 2015 from https://www.adspeed.com

[12] Bag of words model. Wikipedia. Retrieved on May 1, 2015 from http://en.wikipedia.org/wiki/Bag-of-words_model

[13] Acceptance Testing. Wikipedia. Retrieved on May 1, 2015 from http://en.wikipedia.org/wiki/Acceptance_testing

[14] Keyword based advertisement. Retrieved on May 1, 2015 from https://courses.cit.cornell.edu/info204_2007sp/kba.pdf

[15] Generalized second prize auction. Wikipedia. Retrieved on May 1, 2015 from http://en.wikipedia.org/wiki/Generalized_second-price_auction

[16] Online advertising. Wikipedia. Retrieved on May 1, 2015 from http://en.wikipedia.org/wiki/Online_advertising

[17] ComScore and Starcom USA Release Updated 'Natural Born Clickers' Study Showing 50 Percent Drop in Number of U.S. Internet Users Who Click on Display Ads". ComScore, Inc. 1 October 2009. Retrieved on May 1, 2015 from http://www.comscore.com/Insights/Press-Releases/2009/10/comScore-and-Starcom-USA-Release-Updated-Natural-Born-Clickers-Study-Showing-50-Percent-Drop-in-Number-of-U.S.-Internet-Users-Who-Click-on-Display-Ads

[18] Search Engine Marketing. Wikipedia. Retrieved on May 1, 2015 from http://en.wikipedia.org/wiki/Search_engine_marketing

[19] Google Advertising Program. Retrieved on May 1, 2015 from https://www.google.com/adwords/

[20] Online Marketing Challenges. Retrieved on May 1, 2015 from http://www.google.com/onlinechallenge/discover/campaign-adwords.html

[21] AdChoices. Retrieved on May 1, 2015 from http://www.youradchoices.com/control.aspx

[22] 10 Things You Should Know About Global Ad Market. Retrieved on May 1, 2015 from http://adage.com/article/digital/online-measurement-16-web-clicking-display-ads/139367/

[23] comScore Releases “The 2010 U.S. Digital Year in Review” Retrieved on May 1, 2015 from http://www.comscore.com/Insights/Press-Releases/2011/2/comScore-Releases-The-2010-U.S.-Digital-Year-in-Review

[24] Key Online Advertising Updates. Retrieved on May 1, 2015 from http://marketrealist.com/2014/09/google-continues-dominate-online-advertising-market/

[25] Marketing Land. Retrieved on May 1, 2015 from http://marketingland.com/microsoft-posts-20-49-quarterly-revs-online-ads-worth-784-million-40654