

GOOGLE'S KEYWORD ANALYSIS TOOL AND COMPARE ITS  
VARIOUS METRICS FOR FINDING THE MOST POPULAR  
SEARCH TERMS ON THE WEB

Md. Fazlul Karim Tusar  
ID: 05141001

**Department of Computer Science and Engineering**  
August 2009



**BRAC University, Dhaka, Bangladesh**

## DECLARATION

I hereby declare that this thesis is based on the results found by myself. Materials of work found by other researcher are mentioned by reference. This thesis, neither in whole nor in part, has been previously submitted for any degree.

---

Signature of Supervisor

---

Signature of Author

## ACKNOWLEDGMENTS

Special thanks to Dr. Mumit Khan who took my undergraduate level *Compiler Design* class and taught me regular expressions, parsing, lexical analysis and for accepting the difficult task of overseeing this industry level work to completion, to Md. Abul Hasnat who took *Compiler Design* lab classes. Basically this undergraduate class taught me how to think of random data set clearly and made me to use different tools together to accomplish a complex task.

## ABSTRACT

Among the search engines, *Google* is the most powerful and popular search engine in the web. *Google* handles roughly 60% of world wide search and *Google Network* is the largest advertising network available online, reaching over 86% of internet users across the web. *Google AdWords Keyword Tool* which works based on searches occurred in *Google* and in *Google Network* is very popular and widely used keyword analysis tool. The keyword tool provides seven different statistics or metrics such as *Estimated Ad Position*, *Estimated Avg. CPC*, *Advertiser Competition*, *Local Search Volume [Last Available Month]*, *Global Monthly Search Volume*, *Search Volume Trends and Highest Volume Occurred In*. But it does not provide any *Keyword Effectiveness Indicator* like **keyword density** based on a web page URL. Providing these *KEI* along with *AdWords Keyword Tool's* existing seven different metrics, make it more effective and noticeably popular for search engine optimization and for *AdSense* publisher.

## TABLE OF CONTENTS

	Page
TITLE .....	i
DECLARATION .....	ii
ACKNOWLEDGEMENTS.....	iii
ABSTRACT .....	iv
TABLE OF CONTENTS .....	v
CHAPTER I. INTRODUCTION	
1.1 An Overview .....	1
1.2 Research Objective .....	2
CHAPTER II. KEYWORDS	
2.1 Keyword .....	3
2.2 Keyword Categories .....	3
2.3 Keyword Research .....	4
2.4 Keyword Research Process .....	4
2.5 Keyword Tools .....	5
CHAPTER III. GOOGLE'S KEYWORD TOOL	
3.1 An Overview .....	6
3.2 Google AdWords API .....	7
3.2.1 Overview at a Glance .....	7
3.2.2 Specifications .....	7
3.2.3 Signup and Licensing .....	9
3.2.4 Support .....	9
3.2.5 Working Procedure with AdWords API .....	10
CHAPTER IV. SYSTEM DEVELOPMENT	
4.1 An Overview .....	11
4.2 Methodology in Details .....	11
4.2.1 Keyword Frequency Retriever .....	11
4.2.2 Accessing AdWords API .....	13
4.2.3 Integration .....	13
4.2.4 Analysis .....	15

4.3 Standalone Keyword Density Checker .....	20
4.4 Wrong Step .....	22
CHAPTER V. COMPARE BETWEEN DIFFERENT METRICS .....	23
CHAPTER VI. SUMMARY .....	24
CHAPTER VII. FUTURE PLAN .....	25
REFERENCES .....	26
KEY TERMS .....	27

## CHAPTER I

### INTRODUCTION

#### 1.1 An Overview

Keyword density refers to the ratio (percentage) of keywords contained within the total number of index able words within a web page. The preferred keyword density ratio varies from search engine to search engine. In general, optimum keyword density for targeting the majority of search engines is considered to be around 3% to 9%. It is calculated from keyword frequency. And keyword frequency refers to the number of times a keyword or keyword phrase appears within a web page. The theory is that the more times a keyword or keyword phrase appears within a web page, the more relevance a search engine is likely to give the page for a search with those keywords.

But it is very important not to abuse the system by repeating the same keyword or keyword phrases over and over again. The higher the better - but not too high. Search engines penalize sites heavily for keyword stuffing that is repeating same keywords or keyword phrases over and over in the hopes of improving the page's ranking.

Although these analyses have long been a standard among optimizer tools, keyword density and keyword frequency is not how search engines evaluate words on website pages. Search engines use “*Term Vector Theory*”, a form of linear algebra, to analyze pages. Keyword density is considered in *Google PageRank*<sup>™</sup> algorithm and has great priority in Contextual Advertising<sup>3</sup>. In contextual advertising system, the system scans the text of a webpage for keywords and returns advertisements to the webpage based on the content of the page. Even though details access to this information is not public; keyword density does have great value for some reasons like –

- Keyword density analysis can make it easier to find a few good keyword phrases to use.
- For Keyword Prominence<sup>1</sup> and Keyword Proximity<sup>2</sup>.
- Optimize publisher sites for Contextual Advertising (e.g. AdSense).
- Promote sites ranking in search engines.
- Prevent websites from being penalized for keyword stuffing.

## 1.2 Research Objective

*Google AdWords Keyword Tool* provides seven different statistics or metrics such as *Estimated Ad Position*, *Estimated Avg. CPC*, *Advertiser Competition*, *Local Search Volume [Last Available Month]*, *Global Monthly Search Volume*, *Search Volume Trends* and *Highest Volume Occurred In*. But it does not provide any *Keyword Effectiveness Indicator* like **keyword density** based on a web page URL. There are so many keyword density checkers and so many keyword tools but no one uses *Google's* keywords database to check their density on a webpage. All popular existing keyword density checkers mainly show single words' keyword density. Some of them produce results for 2-word-density and 3-word-density also but these 2-word and 3-word phrases don't produce meaningful or search engine keywords always. They don't use any standard database of keywords to find keyword phrases. They just find out which consecutive 2/3 words repeatedly use in a web page in same pattern. Then the result is - that consecutive 2/3 words, as a 2-word or 3-word phrase and how many times it repeatedly used, is its frequency.

So my plan is to develop a keyword tool based on *Google's AdWords Keyword Tool API*, which can produce keyword density for API's resulting keywords and keyword phrases. These are *Google's* actual keywords for a given web page URL from *Google's* keywords database. *Google* uses these keywords for their search engine indexing and online advertising for both *AdWords* and *AdSense*.



## CHAPTER II

### KEYWORDS

#### 2.1 Keyword

Keyword is a word or words used by a searcher on a search engine. Keywords in fact are words or phrases that describe contents and are used to identify what a user is searching for on the web.

In SEO, keywords are the words that a website is optimized to rank for and in PPC, keywords are bid on by advertisers.

#### 2.2 Keyword Categories

Keywords can be broken down into three categories –

Broad Keywords: These are very general keywords that are searched often. Many times are searched for information purposes as much as purchasing reasons. For some sites and keywords, these can convert well enough to have a positive ROI, but in general they don't have the conversion rate of the other keyword types.

Direct Keywords: These keywords are directly related to products or services. They usually include a descriptive term. These keywords make up the bulk of most campaigns, usually have a positive ROI.

They are not searched for as often as broad keywords, so sometimes a campaign can be helped by the increased exposure from broad keywords.

Niche Keywords: These keywords are highly specific, but have low search rate. They usually have the highest ROI and conversion rates among all keywords in a campaign. Often, they are not enough niche keywords to only rely on these to make up a campaign as there just are not enough searches done in a month for them to sustain a business. They are a nice addition to direct keywords as they usually have a lower bid price.

## 2.3 Keyword Research

Keyword Research is by far the most important aspect in any Search Engine Optimization initiative. Keyword Phrase Research is a process of selecting the most optimum performance keyword phrases that can help visitors find the targeted site. Webmaster may have spent days and months on fine-tuning a web page for a better ranking with the major search engines, yet it will all amount to a big waste if the right keyword phrases are not targeted. Even if anyone achieves high search engine rankings, he may not get relevant traffic if he selects the wrong keywords.

For any marketing strategy to succeed, it is critical to know the audience and the means to reach them. A certain focus is required which could be location specific, region specific or country specific; it could be business, trade, service, product specific, since we are talking specific audience. Keywords are the building blocks of a pay per click campaign. These are the search queries that users type into a search engine which trigger advertiser's ads. Choosing keywords which aren't highly relevant to specific products and offerings will lead to either a low click through rate, or money spent on these visitors who don't convert.

## 2.4 Keyword Research Process

Keyword Research process involves the following important steps. These steps can be described as –

1. Discovering Keywords: The discovering phase should focus on identifying as many keywords as possible that are related to our site and target audience.
2. Analyzing Keywords: The analysis phase involves adding information about existing competition; PageRank based limitations and potential for ranking.
3. Selecting Keywords: The selection process involves objective measurement based short listing of keywords keeping the site focus and target audience within the limitations analyzed.
4. Deploying Keywords: The deploying phase is about making optimum use of our selected keywords on our website copy, HTML code, tags and online advertisements.

## 2.5 Keyword Tools

Not all keywords are worthy to target. One should select his list by considering the demand and competition for each keyword. Here demand means how many people used the keyword during a certain time period and competition means the number of web pages a particular search engine brings as the result. Competition will determine how easy or difficult it to fight for that keyword.

Keyword types based on demand and competition –

1. Low demand and Low competition - Worthy to target.
2. Low demand and High competition - Won't be profitable to target.
3. High demand and High competition - Will be hard to dominate.
4. High demand and Low competition - Preferred ones. Begin with these.

Above list indicates just the natural and common preference. But subject knowledge and experiences must be used to refining keyword list. So learning and understanding keyword related different metrics is essential for keyword research. Therefore, if anyone wants to do well in the internet marketing business and keyword research then a good, highly professional and high quality keyword analysis and selector tool is a must for him.

There are a large number of such tools available in the market as well as on the internet. Most popular and widely used keyword tools are –

1. Google AdWords Keyword Tool  
[<https://adwords.google.com/select/KeywordToolExternal>]
2. Wordtracker  
[<http://www.wordtracker.com>]
3. Keyword Discovery  
[<http://www.keyworddiscovery.com>]
4. Overture Search Term Suggestion Tool [now Yahoo! Search Marketing]

## CHAPTER III

### GOOGLE'S KEYWORD TOOL

#### 3.1 An Overview

The *Google AdWords Keyword Tool* provides several statistics with keyword results. This information can help us pick the keywords most likely to return quality leads within our budget. Here's a short description of each type of statistic –

Estimated Ad Position: This column shows the estimated position for ads appearing on each keyword based on the maximum cost-per-click (CPC) that we enter in the calculate estimates using a different maximum CPC bid field.

Estimated Avg. CPC: This column shows the estimated average cost-per-click (CPC) for each keyword based on the maximum CPC that we enter in the calculate estimates using a different maximum CPC bid field.

Advertiser Competition: This column shows the number of advertiser worldwide bidding on each keyword relative to all keywords across Google. The shaded bar represents a general low-to-high quantitative guide to help us determine how competitive ad placement is for a particular keyword.

Local Search Volume [Last Available Month]: This column shows the approximate number of search queries matching each keyword. This statistic applies to searches performed on Google and the search network in the most recent month that Google has data for. If Google doesn't have enough data for a particular keyword, we'll see not enough data.

Global Monthly Search Volume: This column shows the approximate average monthly number of search queries matching each keyword. This statistic applies to searches performed on Google and the search network over a recent 12-month period. It includes traffic in all countries and languages. If Google doesn't have enough data for a particular keyword, we'll see not enough data.

Search Volume Trends: This column shows fluctuations in Google search volume for each keyword over a recent twelve-month period. Each bar in the graph is relative to the keyword's overall performance for the 12-month period. This statistic is specific to our targeted country and language. If Google don't have enough data for the targeted country and language, we'll see no data.

Highest Volume Occurred In: This column shows the month that each keyword received the highest Google search volume within a recent 12-month period, specific to our targeted country and language. If Google doesn't have enough data for our targeted country and language, we'll see no data.

## 3.2 Google AdWords API

### 3.2.1 Overview at a Glance

Summary	: Search advertising in Google and Google Network
Protocol	: SOAP <sup>0</sup> (SOAP Toolkit: NuSOAP)
Data Formats	: XML
API Home	: <a href="http://code.google.com/apis/adwords">http://code.google.com/apis/adwords</a>

### 3.2.2 Specifications

Functionality:

API Web Services:

The AdWords API web services logically fall into two categories –

- Campaign Data Services: Use these web services to edit AdWords campaigns and associated data like manage campaigns, ad groups, criteria and ads. AdGroupService, AdService, CampaignService and CriterionService are considered as campaign data services.
1. AdGroupService: All operations that pertain to an ad group as a whole are handled by AdGroupService. Main operations –
    - a) Creating An Ad Group
    - b) Adding Targeting Criteria and Ads to Ad Groups
    - c) Updating Ad Groups
    - d) Getting Information about Ad Groups

2. AdService: AdService lets us create ads, associate them with an ad group, modify them and delete them. Operations –
  - a) Creating Ads
  - b) Updating Ads
  - c) Getting Information about Ads
  
3. CampaignService: This web service lets us perform campaign-wide operations. Main operations –
  - a) Creating A Campaign
  - b) Updating Campaigns
  - c) Getting Information about Campaigns
  - d) Pausing Campaigns
  - e) Enabling Conversion Optimizer
  
4. CriterionService: Use keyword and placement targeting to control how ads are considered for display. The CriterionService allows us to get, add, update and remove criteria associated with an ad group. It also allows us to get simple statistics for the criteria associated with an ad group. Operations –
  - a) Creating Keywords and Websites
  - b) Updating Keywords and Websites
  - c) Getting Information about Keywords and Websites
  
- Utility Services: Use these web services to manage AdWords account and to assist us intelligently editing our campaigns. AccountService, InfoService, KeywordToolService, ReportService, SiteSuggestionService and TrafficEstimatorService are considered as utility services.
  1. AccountService: The Account Service allows us to retrieve and update AdWords account data.
  
  2. InfoService: The Info Service allows us to obtain some basic information about our API usage.

3. KeywordToolService: The Keyword Tool Service allows us to generate new keywords based on seed keywords we provide or on the words found on a website from a URL that we provide. Each keyword includes traffic information, such as relative values of advertiser competition, search volume etc. Main operations –
  - a) Produces keyword variations based on a seed keyword
  - b) Produces keyword variations based on words found on your website
4. ReportService: Use ReportService to generate reports about the performance of AdWords campaigns.
5. SiteSuggestionService: SiteSuggestionService provides operations for getting web site suggestions four different ways - by category, topics we define, URLs or demographics from Google Network.
6. TrafficEstimatorService: TrafficEstimatorService provides operations for estimating keyword traffic, campaign traffic and ad group traffic.

### 3.2.3 Signup and Licensing

Requirements: Developer Token and Application Token

Licensing: Register as a developer at AdWords API is not free. But AdWords API Sandbox provides free testing and development environment without spending API units.

### 3.2.4 Support

Language Support : All popular programming languages

API Documentation : <http://code.google.com/apis/adwords/docs>

API Blog : <http://adwordsapi.blogspot.com>

### 3.2.5 Working Procedure with AdWords API

1. If we choose PHP, then we have to write a PHP client script to access Google AdWords API.
2. To access Google AdWords API, developer and application tokens are needed.
3. After registering at AdWords API as a developer, we will receive the developer and application tokens.
4. Register as a developer at AdWords API is not free. Each operation performed on an AdWords account, such as adding a keyword, consumes a certain number of API units, on a regular basis, each developer will be billed USD\$0.25 per thousand units consumed.
5. AdWords API Sandbox provides testing and development environment without spending API units (calls made in the sandbox are free). But it does not provide accurate result as registered developer and application tokens do. It's good for practice and testing.
6. The PHP client requests for the AdWords Keyword Tool Service of the Google AdWords API by SOAP request.
7. This request returns the data in XML format. That XML data need to be parse and then save it to database.
8. Customize these data with individual preference and strategy for customize Keyword Tool and display in the browser.



## CHAPTER IV

### SYSTEM DEVELOPMENT

#### 4.1 An Overview

My main target is to develop a keyword tool based on *Google AdWords API's Keyword Tool Service* which shows *Google Keyword Tool's* existing keyword metrics based on a given webpage URL with a new metric **keyword density**.

To calculate keyword density on a given webpage URL, it's not possible to get any help from AdWords API. Because it doesn't provide any raw data from a parsed webpage.

To solve this issue I have decided to write a PHP script that would parse a given webpage URL and would return me index able words from that webpage with their frequency. So that I can easily calculate their density and can head forward to my main keyword tool.

#### 4.2 Methodology in Details

##### 4.2.1 Keyword Frequency Retriever

I have completed the implementation part of a *Keyword Frequency Retriever*. It can successfully calculate keyword frequency from any given webpage URL.

Here I have used two classes –

1. `Basic_HTTP`
2. `simple_html_dom`

HTTP page retrieve using built-in functions or user defined functions (like `file_get_contents()`) lack one point. Probably it transfers data as one character at a time and every time checks the EOF. This normally slow down the whole process in case of large web page that contains much more texts. And data transfer over internet using TCP/HTTP is not a continuous process. However, some web servers has disabled the URL retrieving capabilities in these functions for security reasons, this caused scripts written with these

functions throw out an error which can render the whole web application useless in that particular web server. There are some other problems those I saw at many forums' discussions and reviews. So I didn't use their solutions. Instead I've written a class which is much more efficient than those. `Basic_HTTP` class takes an URL as input, then gets the specified html page through the URL and retrieves the source file of that URL as fetched web page (in `fetchedData`).

At this stage of my work, the `Basic_HTTP` class took most of my time to be finished. I have to cover almost all the internal tasks that a web browser has to do. It takes a URL as input and then checks the protocol (`http/https/ftp/ftps`) and set corresponding port (`80/443/21/990`) with appropriate HTTP headers and their parameters. Here, I set get method in HTTP headed request. In HTTP header portion, I have to carefully set User-Agent string and Cookies. After that prepared the socket connection, request the data using `requestString` and read the server responses in `responseString`. Then processed it and fetched the content to handover it to the parser.

Now it can nicely handle only HTTP, HTTPS protocol, 301 and 302 redirections. I have used `simple_html_dom` class for parsing the html file. Here I used "PHP Simple HTML DOM Parser" written in PHP5+, which is an open source parser and licensed under the "MIT License". I downloaded it from <http://sourceforge.net/projects/simplehtmldom>. I didn't change anything in the parser. Though HTML is not a regular language, it nicely handles valid HTML and matches everything among a tag. This Simple HTML DOM Parser takes a html file as input and extracts tag from it, then produces plain text as output. I just created an object of that class (`simple_html_dom`) to use the parser.

Then I have used *brute force* method to count fetched keywords' frequency. Here I used PHP4+ built-in functions `str_word_count()` and `array_count_values()` to fetch keywords and to run *brute force*. Here I choose *brute force* instead of *dynamic* approach because of its simplicity. I think in this case it wouldn't be a big matter. In future improvement of my code I will use *dynamic* approach.

Input & Output:

Input : An URL

Output: An array of words with their frequency

Sample Input: A webpage URL. For example: *www.domain.com/index.html*

Sample Output:

Keyword	Keyword Frequency
domain	14
bulk	3
hosting	5
private	4

*Table-1*

#### 4.2.2 Accessing AdWords API

I have completed my coding to access the *Google AdWords API's* `KeywordToolService`. Here I have used **APility** PHP Library, which is a *Google's* open-source project. The **APility** PHP Library provides an object-oriented way to easily access and manage the *Google AdWords API* from within PHP. This comes along with an abstraction from the SOAP and WSDL details. This helps programmers to concentrate on sophisticated programs and tools, not on complex API details.

#### 4.2.3 Integration

The most important part of my work is to integrate my *Keyword Frequency Retriever* with *Google AdWords API's* `KeywordToolService`. That is these two tools' result into one arranged result. For example, let consider a site (<http://www.hostingprices.com>) that contains data about hosting. So when we use *AdWords Keyword Tool External*, for example, we get the following keywords with some metrics –

Keywords	Google Keyword Tool's existing keyword metrics			Other metrics
	Global Monthly Search Volume	Local Search Volume (Last Month)	Estimated Avg. CPC	
hosting	11,100,000	7,480,000	\$11.04	Other data
web hosting	5,000,000	3,350,000	\$13.62	
dedicated hosting	246,000	246,000	\$19.58	
website hosting	450,000	301,000	\$11.35	
vps hosting	60,500	40,500	\$9.90	
free host	301,000	165,000	\$2.05	
dedicated server	14,800	14,800	\$12.57	

Table-2

And when I consider my *Keyword Frequency Retriever*, I get –

Keywords	Frequency
hosting	25
Prices	4
hostingprices	1
Web	6
Website	1
Free	10
Server	2
Internet	1
Severs	2
housing	1

Table-3

#### 4.2.4 Analysis

##### **Problem-1:**

I have to combine the above 2 tables into one. But here is the main problem I am facing. Actually *Google* provides keywords or keyword phrases that people used to search in search engines (here *Google*). So *Google's* keyword Tool first analyze the webpage, try to detect what the main topic or topics of that page is or are, detect the main keywords of that page, then shows result related to those topics from it's database. This is why; the resulting keywords are not always present in that page. For example – “dedicated”, “vps” and “host” are not present in our analyzed page (<http://www.hostingprices.com>) but *Google's Keyword Tool's* result containing these keywords. My keyword frequency finder only works on webpage's raw text. So it can't find these keywords. It's a great problem for mash-up.

##### **Problem-2:**

Second thing is *Google's Keyword Tool's* any keyword analysis result contains both keywords and keyword phrases. But I have found out keyword frequency for keyword, not for keyword phrase. So mash-up these two is not an easy task. Even if we consider two different tool's (*Google's AdWords Keyword Tool & Overture Search Term Suggestion Tool* (now *Yahoo! Search Marketing*)) keyword analysis results based on the above URL, we can't guarantee that we can merge these two results easily. Because two different search engines page analysis, keyword detection technique and keyword database should be different. So if I try to calculate frequencies of 2 word phrases and 3 word phrases rather than only single words, it will not be helpful in case of mash-up. I have checked most popular keyword density checkers around the web and I have noticed two main things –

1. There are so many keyword density checkers and so many keyword tools but no one shows results based on integration with *Google's Keyword Tool*. Other available keyword tools in the web mainly use *Overture's* and *Wordtracker's* API.

2. All popular existing keyword density checkers mainly show single words' keyword density. Some of them produce results for 2-word-density and 3-word-density also but these 2-word and 3-word phrases don't produce meaningful or search engine keywords always. They don't use any standard database of keywords to find keyword phrases. They just find out which consecutive 2/3 words repeatedly use in a web page in same pattern. Then the result is - that consecutive 2/3 words, as a 2-word or 3-word phrase and how many times it repeatedly used, is its frequency. But this type of 2-word and 3-word phrases will not much helpful for my tool as we already know the cause. On the other hand, I have read so many articles from the web about keyword density analysis and SEO as there is no text or arranged resources, every where they preferred or discussed about 1-word-density rather than 2-word-density and 3-word-density. As keyword density measurement in terms of SEO is a search engine copyright technique, there are many debates and many predictions.

### **Solution:**

I have set a simple example to present my proposed solution. Let's take this webpage ([http://www.digiwor.com/k\\_tool/home.html](http://www.digiwor.com/k_tool/home.html)) as input URL. First, I will take all parsed texts in an array where every words and punctuation marks will store in separate indexes. So the array will be like this –

- |                   |                |               |
|-------------------|----------------|---------------|
| 1. The            | 24. some       | 47. Prices    |
| 2. best           | 25. of         | 48. and       |
| 3. Hosting        | 26. the        | 49. Hosting   |
| 4. Prices         | 27. world      | 50. Offers    |
| 5. .              | 28. top        | 51. .         |
| 6. It's           | 29. webhosting | 52. SQL       |
| 7. time           | 30. companies  | 53. ,         |
| 8. for            | 31. .          | 54. PHP       |
| 9. a              | 32. You        | 55. ,         |
| 10. change        | 33. will       | 56. CGI       |
| 11. ?             | 34. find       | 57. ,         |
| 12. HostingPrices | 35. a          | 58. FrontPage |
| 13. .             | 36. Top        | 59. ,         |
| 14. com           | 37. 10         | 60. Perl      |
| 15. selects       | 38. selection  | 61. ,         |
| 16. daily         | 39. with       | 62. free      |
| 17. the           | 40. the        | 63. domain    |
| 18. 10            | 41. best       | 64. ,         |
| 19. best          | 42. hosting    | 65. email     |
| 20. web           | 43. companies  | 66. accounts  |
| 21. hosting       | 44. including  | 67. ,         |
| 22. companies     | 45. the        | 68. Etc       |
| 23. from          | 46. Hosting    | 69. .         |

70. #	118. ,	166. would
71. reliable	119. cheap	167. require
72. hosting	120. hosting	168. a
73. ,	121. What	169. high
74. good	122. is	170. speed
75. service	123. web	171. connection
76. SQL	124. hosting	172. ,
77. ,	125. ?	173. the
78. PHP	126. Web	174. Server
79. ,	127. hosting	175. hardware
80. CGI	128. is	176. and
81. ,	129. the	177. the
82. FrontPage	130. business	178. expertise
83. ,	131. of	179. to
84. Perl	132. housing	180. set-up
85. ,	133. ,	181. and
86. free	134. Serving	182. configure
87. domain	135. ,	183. a
88. ,	136. and	184. web
89. email	137. maintaining	185. Server
90. accounts	138. files	186. .
91. ,	139. for	187. Using
92. Etc	140. a	188. a
93. .	141. website	189. hosting
94. #	142. .	190. service
95. professional	143. Hosting	191. allows
96. hosting	144. removes	192. companies
97. ,	145. the	193. and
98. low	146. need	194. individuals
99. price	147. of	195. to
100. PHP	148. you	196. get
101. ,	149. having	197. on
102. CGI	150. to	198. to
103. ,	151. provide	199. the
104. FrontPage	152. and	200. Internet
105. ,	153. maintain	201. without
106. Perl	154. expensive	202. the
107. ,	155. and	203. expense
108. free	156. complicated	204. or
109. domain	157. Servers	205. effort
110. ,	158. .	206. of
111. email	159. An	207. maintaining
112. accounts	160. individual	208. their
113. ,	161. business	209. own
114. Etc	162. hosting	210. Servers
115. .	163. its	211. .
116. #	164. own	
117. Good	165. site	

Table-4

(See over)

And Google's AdWords Keyword Tool's result (a portion of main output) for the above URL is –

Keywords	Estimated Ad Position	Estimated Avg. CPC	Advertiser Competition	Local Search Volume: June	Global Mo
hosting	2	\$11.04		1	7480000
web hosting	2	\$13.62		1	3350000
business hosting	2	\$16.73		1	246000
free web hosting	2	\$7.30		1	450000
business web hosting	2	\$16.50		1	201000
reseller hosting	2	\$10.80		1	201000
vps hosting	2	\$9.90		1	40500
windows web hosting	2	\$14.89		1	40500
asp net 2.0 hosting	2	\$17.88	0.93		2400
asp net hosting	2	\$17.09		1	49500
asp web hosting	2	\$18.43		1	90500
dedicated hosting	2	\$19.58		1	246000
domain hosting	2	\$10.26		1	450000
email hosting	2	\$9.28		1	165000
net hosting	2	\$15.84		1	110000
asp hosting	2	\$16.01		1	165000
cpanel hosting	2	\$6.98		1	40500
top 10 web hosting	2	\$15.26		1	8100
top web hosting	2	\$14.83		1	33100
web hosting companies	2	\$16.56		1	33100
best web hosting	2	\$15.24		1	135000
cheap web hosting	2	\$11.17		1	301000
web hosting review	2	\$16.60		1	40500
free hosting	2	\$4.11		1	1000000

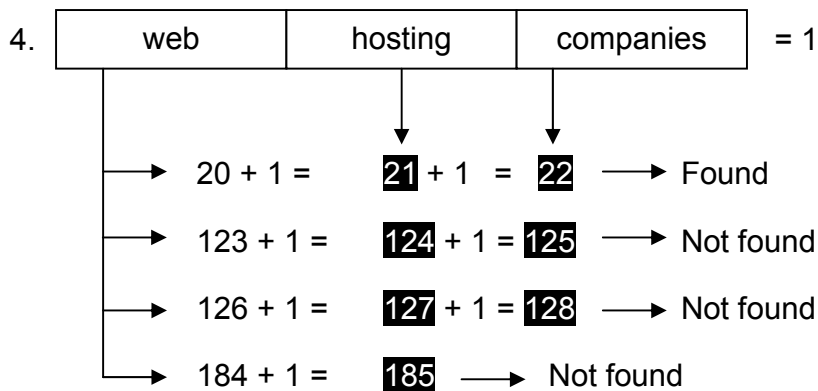
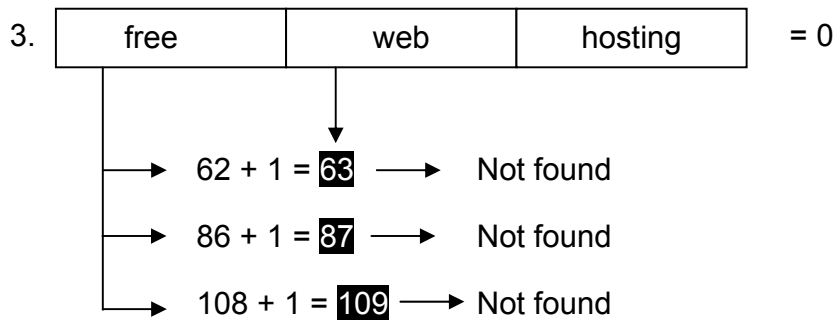
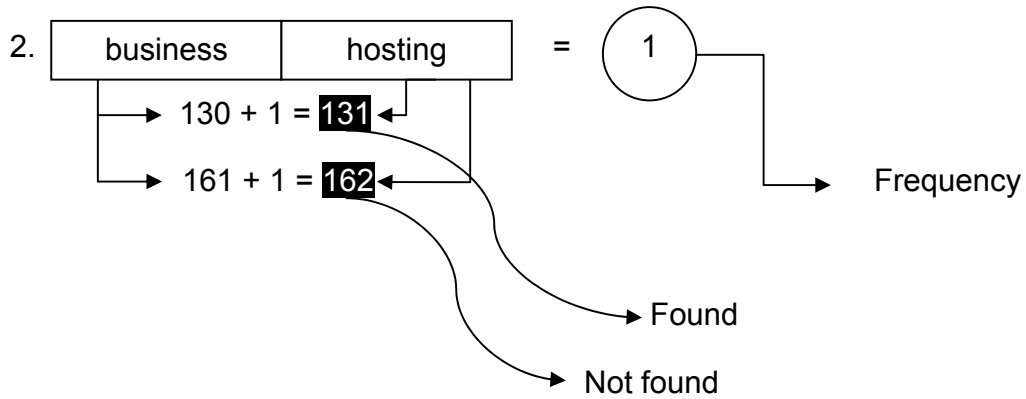
Table-5

So to find the frequency of 2<sup>nd</sup> keyword phrase (*web hosting*) of Table-5:

- **web hosting** → **web hosting**  
(Separate the words from keyword phrase and tag every word separately so that we can identify them separately.)
- Find all matches and their index numbers of 1<sup>st</sup> word “web” and Table-4. In this case try to make the Table-4 such a way that this searching would be faster. In this case –  
**web** → found in 20, 123, 126 and 184 positions
- Now search for **hosting** in 20+1=21, 123+1=124, 126+1=127 and 184+1=185 positions. For each successful finding count it for the whole keyword phrase's frequency.
- If the keyword phrase contains more than two words then repeat previous step for searching next words and count last word/part's successful finding for frequency.



Some more examples based on Table-4 and Table-5:



So this proposed solution might be working for finding keyword density of Google AdWords Keyword Tool's keyword result based on a webpage URL.

### 4.3 Standalone Keyword Density Checker

I also want to make my keyword tool as a standalone keyword density tool. So first I have to determine all words and their frequencies of that webpage. But all words of that page are not keywords or index able words. So I have to remove very common words that search engines do not consider as keywords. After study and observing so many search results from Google I have prepared a list of words ignored by search engines. I have to perform an analysis more than 500+ searches and take help from SEO sites to prepare this list. Then I will get filtered result that contains only index able words and their frequencies (Example - Table-2 for URL: <http://www.hostingprices.com>).

#### List of ignore able words:

a	ask	consequently	ex	help
able	asking	consider	exactly	hence
about	associated	considering	example	her
above	at	contain	except	here
abroad	available	containing	f	hereafter
according	away	contains	fairly	hereby
accordingly	awfully	corresponding	far	herein
across	b	could	farther	here's
actually	back	couldn't	few	hereupon
adj	backward	course	fewer	hers
after	backwards	c's	fifth	herself
afterwards	be	currently	first	he's
again	became	d	five	hi
against	because	dare	followed	him
ago	become	daren't	following	himself
ahead	becomes	definitely	follows	his
ain't	becoming	described	for	hither
all	been	despite	forever	hopefully
allow	before	did	former	how
allows	beforehand	didn't	formerly	howbeit
almost	begin	different	forth	however
alone	behind	directly	forward	hundred
along	being	do	found	i
alongside	believe	does	four	i'd
already	below	doesn't	from	ie
also	beside	doing	further	if
although	besides	done	furthermore	ignored
always	best	don't	g	i'll
am	better	down	get	i'm
amid	between	downwards	gets	immediate
amidst	beyond	during	getting	in
among	both	e	given	inasmuch
amongst	brief	each	gives	inc
an	but	edu	go	inc.
and	by	eg	goes	indeed
another	c	eight	going	indicate
any	came	eighty	gone	indicated
anybody	can	either	got	indicates
anyhow	cannot	else	gotten	inner
anyone	cant	elsewhere	greetings	inside
anything	can't	end	h	insofar
anyway	caption	ending	had	instead
anyways	cause	enough	hadn't	into
anywhere	causes	entirely	half	inward
apart	certain	especially	happens	is
appear	certainly	et	hardly	isn't
appreciate	changes	etc	has	it
appropriate	clearly	even	hasn't	it'd
are	c'mon	ever	have	it'll
aren't	co	evermore	haven't	its
around	co.	every	having	it's
as	com	everybody	he	itself
a's	come	everyone	he'd	i've
aside	comes	everything	he'll	j
	concerning	everywhere	hello	just

k	no-one	seem	they've	where
keep	nor	seemed	thing	whereafter
keeps	normally	seeming	things	whereas
kept	not	seems	think	whereby
know	nothing	seen	third	wherein
known	notwithstanding	self	thirty	where's
knows	novel	selves	this	whereupon
l	now	sensible	thorough	wherever
last	nowhere	sent	thoroughly	whether
lately	o	serious	those	which
later	obviously	seriously	though	whichever
latter	of	seven	through	while
latterly	off	several	throughout	whilst
least	often	shall	thru	whither
less	oh	shan't	thus	who
lest	ok	she	till	who'd
let	okay	she'd	to	whoever
let's	old	she'll	together	whole
like	on	she's	too	who'll
liked	once	should	took	whom
likely	one	shouldn't	toward	whomever
likewise	ones	since	towards	who's
little	one's	six	tried	whose
look	only	so	tries	why
looking	onto	some	truly	will
looks	opposite	somebody	try	willing
low	or	someday	trying	wish
lower	other	somehow	t's	with
ltd	others	someone	twice	within
m	otherwise	something	two	without
made	ought	sometime	u	wonder
mainly	oughtn't	sometimes	un	won't
make	our	somewhat	under	would
makes	ours	somewhere	underneath	wouldn't
many	ourselves	soon	undoing	x
may	out	sorry	unfortunately	y
maybe	outside	specified	unless	yes
mayn't	over	specify	unlike	yet
me	overall	specifying	unlikely	you
mean	own	still	until	you'd
meantime	p	sub	unto	you'll
meanwhile	particular	such	up	your
merely	particularly	sup	upon	you're
might	past	sure	upwards	yours
mightn't	per	t	us	yourself
mine	perhaps	take	use	yourselves
minus	placed	taken	used	you've
miss	please	taking	useful	z
more	plus	tell	uses	zer
moreover	possible	tends	using	
most	presumably	th	usually	
mostly	probably	than	v	
mr	provided	thank	value	
mrs	provides	thanks	various	
much	q	thanx	versus	
must	que	that	very	
mustn't	quite	that'll	via	
my	qv	thats	viz	
myself	r	that's	vs	
n	rather	that've	w	
name	rd	the	want	
namely	re	their	wants	
nd	really	theirs	was	
near	reasonably	them	wasn't	
nearly	recent	themselves	way	
necessary	recently	then	we	
need	regarding	thence	we'd	
needn't	regardless	there	welcome	
needs	regards	thereafter	well	
neither	relatively	thereby	we'll	
never	respectively	there'd	went	
neverf	right	therefore	were	
neverless	round	therein	we're	
nevertheless	s	there'll	weren't	
new	said	there're	we've	
next	same	theres	what	
nine	saw	there's	whatever	
ninety	say	thereupon	what'll	
no	saying	there've	what's	
nobody	says	these	what've	
non	second	they	when	
none	secondly	they'd	whence	
nonetheless	see	they'll	whenever	
noone	seeing	they're		

#### 4.4 Wrong Step

Due to **Problem-2** in section **4.2.4**, I thought that a keyword database can help me finding keyword phrases, which can ultimately help me integrating with AdWords Keyword Tool's result. So I started to build-up an initial keyword database (*keywords.xls*). I collected many keywords from the web, AdWords Keyword Tool, Wordtracker, Overture and many more. It already contains almost 10,000 keywords. But later on, I realized that it would not very helpful for me. I already explained it in **Problem-2** section.

## CHAPTER V

### COMPARE BETWEEN DIFFERENT METRICS

The tool has built provides eight different statistics or metrics such as *Keyword Density*, *Estimated Ad Position*, *Estimated Avg. CPC*, *Advertiser Competition*, *Local Search Volume [Last Available Month]*, *Global Monthly Search Volume*, *Search Volume Trends and Highest Volume Occurred In* with keyword variations. It is discussed earlier that *Google* keyword tool's result with keyword density always ends with a result that better matches search engine optimizer and publisher's needs. This approach will help to keep the publisher sites stable. For example –

1. When a publisher prepare his sites for *AdSense*, he first use the tool for keyword analysis for a webpage, then see for targeted keywords from the result and check their density if it matches proper density criteria or not. Then he will use those keywords to write his title tag, Meta tag, heading etc.
2. If a publisher wants to target high paying ads on his site, then he will see for keywords with highest *Estimated Avg. CPC* and prepare his site with those keywords with proper density.
3. If the *Advertiser Competition* is full or high then he will never suffer from *Google's Public Service Ads* or from blank ad space.
4. Publisher can target high traffic keywords from the analysis to keep an eye on *Global Monthly Search Volume or Local Search Volume*. But if the search volume is high then the competition might be too high. So other metrics should be considered in this case.

## CHAPTER VI

### SUMMARY

This thesis deals with keywords, which is the main building blocks of online advertising and search engine indexing. It's an industry term and this is why no book or arranged resource is available in the market, only some discussions in different forums and some industry materials available to study.

The main achievement of this thesis is to make a way to check the density of *Google's* keywords on a given webpage URL. No other keyword tools in the market done this before. Also this tool can do this in real time. So no keyword database is needed. We are free from the hassle of updating a keyword database regularly.

In my design, I have put two options to analysis a webpage. One is standalone density checker which checks 1-word density on a webpage. It removes non-index able words and shows only index able words with their density to us. When this works, no call have made to *Google API*.

The other option is details analysis which makes call to both my *Keyword Frequency Retriever* and *Google API*, then use the algorithm discussed in section 4.2.4 to produce results. It provides keyword variations and *AdWords Keyword Tool's* existing seven different metrics with a new metric **Keyword Density**.

This tool will help for those who are going to setup and optimize their sites' web pages for search engine and ad publishing. At this stage, the tool works well as an initial keyword optimization tool. Now it is considering only the HTML body for analysis. There is not tag analyzer present at this time. That's why; it gives same level of result for a non optimized page as well as for an already optimized page.

## CHAPTER VII

### FUTURE PLAN

In future I will replace the brute force method in my *Keyword Frequency Retriever* with dynamic approach for efficient searching. This will make the process time faster and reduce the memory usages. I have planned to add individual tag analyzer in my tool in future. So that it will be more helpful for keyword analysis.

## REFERENCES

- Google AdWords Keyword Tool : <https://adwords.google.com/select/KeywordToolExternal>
- Google AdWords API : <http://code.google.com/apis/adwords>
- Keyword Research Procedure : <http://www.redalkemi.com>
- Google Scholar : <http://scholar.google.com>
- Wikipedia : <http://en.wikipedia.org>
- APllity : <http://google-apility.sourceforge.net>



## KEY TERMS

**1** Keyword Prominence refers to how prominent keywords are within a web page such as at or near, the start of a web page, sentence, as bold, TITLE, META tags or within other tags etc.

**2** Keyword Proximity refers to the closeness between two or more keywords. In general, the closer the keywords are - the better.

**3** Contextual advertising is a form of targeted advertising where the advertisements themselves are selected and served by automated systems based on the content displayed to the user. A contextual advertising system scans the text of a website for keywords and returns advertisements to the webpage based on what the user is viewing.

**4** Keyword Effectiveness Indicator (KEI) helps to measure and quantify the quality and worth of a search term.

**5** Density of a keyword =  $(N/T) * 100$ , where N is how many times a specific keyword repeated and T is the total words in the analyzed text.

**6** Density of a keyword phrase =  $(N*M/T) * 100$ , where N is how many times a specific keyword phrase repeated, M is the number of words in the phrase and T is the total words in the analyzed text.