Adaptive Advertising Algorithm

Ivan B Wong, MSEE University of Southern California

Motivation

The following algorithm was developed using ideas and fundamentals learned from data compression techniques including Huffman tree, quantization, and codebook utilization. The format of this paper is used to explain and demonstrate the adaptive advertising algorithm with respects to the Google Adsense™ advertising platform and to serve as the basis for constructing an online prototype.

Abstract

Optimizing online advertisement is a combination of serving contextually relevant ads and providing an optimized visual display of the ads on a webpage. Usually, contextual relevancy is controlled by the ad serving system which scans the page for keywords and displays ads related to the topic of the webpage. On the other hand, the visual display of ads is done manually by ad publishers. For example, ad publishers have to customize ad parameters and insert the ad codes into a given webpage. The goal of the adaptive advertising algorithm is to provide a method to optimize the visual display of an ad using a self-learning adaptive algorithm, thereby eliminating the need for the ad publisher to customize or 'tweak' the ads manually.

Keywords - Adaptive Advertising; Ad Optimization; Google Adsense™

I. Introduction

The online advertising industry has been growing aggressively with the increase of online budgets and a shift away from traditional advertising means. One of the current leaders of online advertisement sales is Google Inc. which offers a program called Google Adsense™ for webmasters to sell advertisements on their websites. The procedure to get ad codes to place on a website is quite trivial. A user will select the properties of an ad: type, format, colors, and then receive a snippet of ad code, typically JavaScript, to place on an area of a webpage (appendix fig. 1 for screenshot of setup page). Once the ad code has been placed, ads contextually relevant to the webpage will be served. Although relevant ads are displayed based on the keywords, the selection of visual ad properties (color, size, etc.), placement, and number of ads per given page is a manual process and typically cumbersome to repeatedly customize or 'tweak' for a given webpage. To this effect, an adaptive advertising algorithm is proposed to

automatically optimize the display variables to increase the effectiveness of ads on a webpage. The proposed algorithm can also apply to other multi-parameter optimization problems, but for conciseness will be explained with respects to the Google Adsense™ platform.

Since the final acceptance or 'click' of an ad is determined by a human individual, it is important to note the tuning of ad display parameters is not a stochastic process, but one that also touches base on human psychology and visual harmony (Appendix Figure 2 is a good example of various ad integration into a webpage). For instance, it does not make sense that black text is placed on a black background or have multiple banner ads displayed successively below each other. A set of forbidden variable combinations will have to be defined depending on parameters in the set. Moreover, favorable variable combinations can be defined into the algorithm to further optimize the adaptive process.

One of the difficulties of adjusting parameters such as the color of the ad's text, background, and border is the correlation that exists between them. If a parameter is tuned separately, it is not as efficient because the parameter's overall effectiveness is dependent on the assignments of its correlated parameters. To this effect it is important to establish a correlation between sets of parameters which are sensitive to each others settings. Moreover, parameter weights will need to be assigned to parameters from least to most effectiveness, so changes to higher weighted parameters can be performed first in the adaptive process. A random 'roulette wheel' process can also be used to choose the next parameter to adjust in which case the weights correspond to the size of the slots. In this case, all parameters will have the opportunity to change and possibly reach better overall optimization.

At a specified time interval, analytics and ad results from a webpage can be recorded. Analytics with correlation to CTR such as new/returning users, age group, geo-location can be entered into a database to provide more specific data for tracking and optimization. The three main items recorded will be the CTR, ad display parameters, and site analytics. A global database will store entries from all of the individual web pages and a local database will handle local entries for a given site. This way optimization can be performed both locally and globally, where local parameters can learn from the global database, such as in the case when an initial set of parameters is first needed. Since the correlation between analytics and ad display parameters may become more apparent or reach an equilibrium, groups of analytic entries can be grouped together to reduce the

overall size of the database and also be used to reduce search complexity and computation.

The paper is explained in the following order. Section II will explain the CTR, ad display parameters, and analytics in more detail. Section III will explain the ad parameter tree which will be utilized to systematically sort and search for the best ad parameters. Section IV will explain the local and database layout and how updates and grouping are performed. Section V explores the importance of ad parameter choices for optimization. Finally, Section VI provides a conclusion to the adaptive advertising algorithm.

II. CTR, Ad Display Parameters, Analytics

CTR, Ad Display Parameters, and Analytics will be recorded into the database at specified time intervals. A sample of the ad display parameters and web analytics, which will be further discussed in detail, are provided in Table 1. The sample intervals can be determined based on the traffic of the webpage where a higher trafficked website can sample at a higher rate. The sampling can be reduced once an optimization threshold has been achieved. In some ways, the sampling rate will also help define the granularity and aggressiveness of the adaptation as changes in the display parameters will be altered depending on the analytic and CTR results from the sampled interval.

Table 1: Example of Ad Display Parameters and Web Analytics

Ad Display Parameters	Web Analytics
Placement	New/Returning Users
Number of Ads	Age Group
Size Format	Page Details (content, colors, etc.)
Type of Ad – text, image, or both	Reference Sources
Color – text, border, background	Geo Location
Various advertisers / advertisements	Time of Day
	Other demographic information

Typically, advertisement effectiveness can be measured by the optimization of the eCPM, which is calculated by dividing total earnings by the number of impressions in thousands. However, since earnings are heavily dependent on keywords, which are already determined by the webpage and ad serving system, the adaptive advertising algorithm will use the CTR, click through rate, instead as the measurement to optimize.

The web analytics for the discussion of this paper will include basic web analytic statistics that can be attained from the Google Analytics™ program, which retrieves web analytics such as geo-location, new/returning users,

referring source, unique visitors etc. for a given webpage. The addition of web analytics helps to refine the algorithm by associating similar analytics groups with optimized parameters. Since there can be infinite number of analytic groups, the analytics will be weighted based on statistical research information and grouped into finite sets. Moreover, analytic groups can be further reduced based on a best match criterion. Initial ad parameters can be retrieved from the global database based on a web page's web analytics. A best match algorithm based on a MSE, mean squared error, approach will be used to determine the closest set to the given web analytics parameters. This will be used when ad parameters based on web analytics are retrieved from the global database and when grouping is performed.

Table 2: Example Weights of Analytics, Sample set of Web Analytics

Weights	
New/Returning Visitors	5
Age Group	3
Reference Source	2
Geo-Location	1

Web Analytics				
Index	New/Returning Visitors	Age Group	Reference Source	Geo-Location
1	40%	19-24	Google Search	Canada
2	25%	35-44	Yahoo Search	Canada
3	2%	19-24	Digg.com	United States
4	1.75%	25-34	Google Search	Switzerland

Table 2 shows the weights, which are ordered by significance with respect to CTR and a sample set of web analytics to show grouping and the best match method. The grouping of web analytics can be useful for truncating the database when a global set of maxima or convergence have been achieved. A threshold can be set accordingly so similar sets can be grouped together based on the closeness of the match and influence on the CTR results. In this example, index 3 and 4 could be grouped since the dominant weighted item, New/Returning

visitor ratio, has little change despite further changes in the other statistics which are weighed less. Likewise, other web analytic sets can be matched to a given database of sets via the best match algorithm. The organization of the weighting can also provide a way to implement a fast search algorithm that will reduce the complexity of searches in a large database.

Ad parameters are slightly different from the analytics as they are variables which are adjusted by the algorithm. Ad parameters include all aspects which influence the visual display of an ad including the placement and number of ads per page. Some ad parameter settings can be seen in Appendix Figure 1 which shows a typical Google Adsense™ setup page. Moreover, ad parameters can also be expanded to include ads from other advertisers or from a pool of advertisements which target a given audience or analytic group. To provide the quickest results from ad parameter changes, ad parameters can be weighted, so adjustments can first be performed on attributes which have larger effects on the CTR. Weights can be derived from human visual perception studies on online advertisements or interpolated from the global set of results. An alternative could be to use a roulette wheel based randomization approach where larger weights correspond to larger slots as seen in Figure 1. This randomized approach may take longer time duration for optimization, but could potentially reach a better optimization maximum since it also allows for adjustments of lower weighted parameters that might not otherwise be changed if a CTR threshold has been met.

Figure 1: 'Roulette Wheel' Randomization with Sample Weights

Weight 5 Ad Parameter 4	Weight 3 Para. 2	Weight 2 Para. 3	Weight 1 Para. 1

Interdependence of ad parameter settings and their corresponding changes during the adaptive process need to adjust according to their sensitivity. For instance, the color parameters of an ad such as text, background, and border should be changed as a set rather than individual parameters. [*note: more study will have to be performed to fine tune and refine the multi-parameter optimization aspect of the algorithm, possibly with testing and simulation, since the computational complexities may or may not lead to any benefits that justify the extra cost*]. The basic idea is to establish a sensitivity relation between the CTRs and Ad parameter groups such as the color settings and to generate a tuning matrix to determine the tuning index. A correlation is first established to correlate the CTR results with the parameter values. In addition, the variation of the CTR

needs to be taken into account. This is to isolate cases in which the correlation is high but the CTR movement is low, ie. Variation is low. Accordingly, sensitivity, S, can be defined as the product of correlation coefficient C and the variation V as:

$$S = C*V \quad (1)$$

$$C = \rho_{xy} = cov(x,y) / \sigma_x \sigma_y \quad (2)$$

$$V_x = (X_{Max} - X_{Min}) / X_{Max} \quad (3)$$

Where x, y represents the CTR x and parameter y. X_{Max} and X_{Min} represents the maximum and minimum values of the CTR x's results in the set of measurement rounds while tuning parameter y. Alternatively, the maximum and minimum values can be attained by interpolating existing values from the global database. To optimize the tuning of multiple parameters simultaneously, a sensitivity-matrix-based optimization approach is designed. A tuning matrix is built with one example show in Table 3.

Table 3: Example of Tuning Matrix

5	10	Group 1	0.3	-0.4
1	-2	Group 2	-0.1	0.2
5	1	Group 3	0.1	0
1	-1	Group 4	0.2	0
		Tuning Index	0.5	-0.4

This sensitivity matrix is calculated by taking different proportions from both the heuristic sensitivity matrix and the current sensitivity matrix. Here the heuristic matrix consists of the sensitivity relations between the CTRs and parameters, whose relations are obtained from the global database. Current sensitivity matrix is calculated by using the current local or global CTR variation and parameters. In the sensitivity matrix, row represents the sensitivity relations of a CTR with multiple parameters.

$$CTR_i = g_i(parameter_1, parameter_2, \ldots, parameter_n) \quad (4)$$

The columns represent the sensitivity relations of one parameter and various CTR characterized by the parameter group.

$$Parameter_i = f_i(CTR\ group_1, CTR\ group_2, \ldots, CTR\ group_n) \quad (5)$$

Since the CTRs from group to group may deviate from the expected threshold CTR results a deficiency function of current CTR and target CTR is defined:

$$CTR_{Deficiency} = e^{(CTR_{current}-CTR_{target})/CTR_{target}} \quad (6)$$

In case of CTR deficiency, parameters relevant to the problematic CTR group will be tuned so that it can be improved. The tuning index can be defined as the product of the CTR group deficiency, preferences, and sensitivities, shown as follows:

$$Index = \sum perference_i * deficiency_i * sensitivity_i$$

Where i represents the row i of the tuning matrix. The index determines whether and in which direction to tune the relevant parameters. The optimization ends where there is not much improvement after several iteration steps.

III. Ad Parameter Tree

A tree based approach will serve to systematically improve the chosen ad parameter, and provide a basis for further adjustments and recording of the parameter optimization path. An ad parameter tree is constructed for a given analytic group on the local level. Depending on the variance and effects of the analytics on the CTR and ad parameters, multiple parameter trees can be defined for a given webpage.

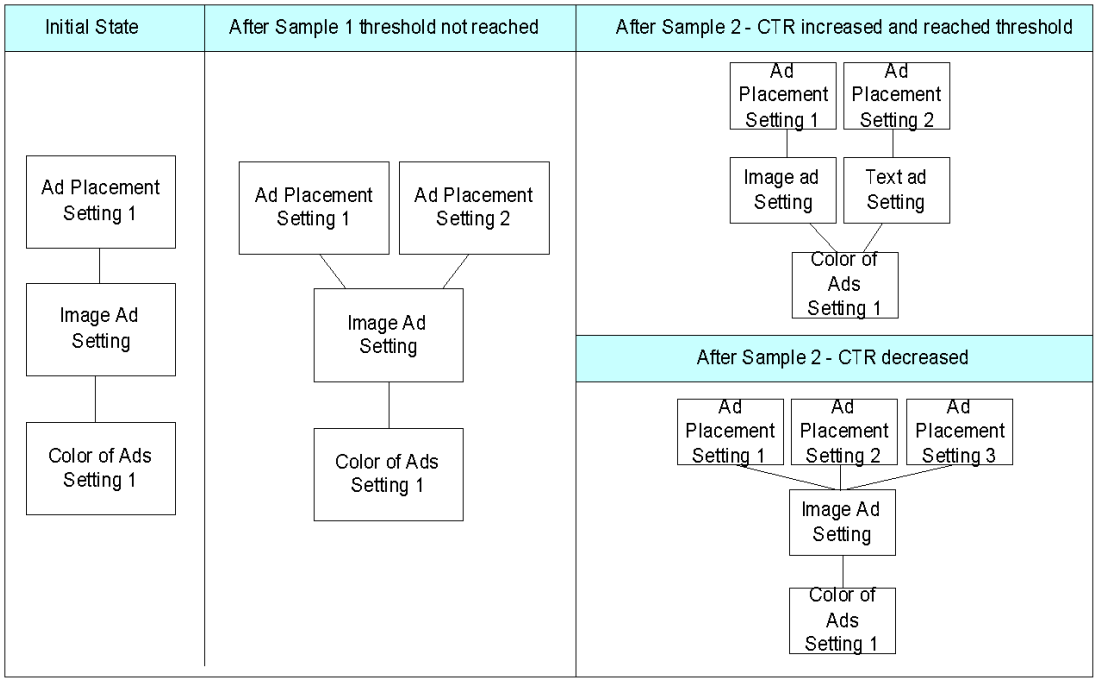
Higher weighted parameters will be positioned higher on the tree and adjusted first assuming the 'Roulette Wheel' method is not used. Since parameters can be obtained from the global database, the iterations for the tree optimization process can be relatively low and serve to fine tune the ad parameters for a given webpage. Moreover, the higher or more optimized ad settings will be moved to the right of the tree branch after each sample time period. Ad parameter groups can also be characterized as a single parameter with multiple sub-parameters.

Figure 2 shows an example of the steps of how the tree works. Initially, a set of ad parameters are chosen for a given webpage. After a sample period, the CTR results of the ad parameters are recorded. Depending on a threshold determined by the convergence of the global set of data, the algorithm will decide to either adjust the first, most significant parameter or the next parameter. Moreover, the number of iterations can be specified to set a limit to the number of

attempts made by the algorithm. In this case, since the threshold was not reached during the first sample period, the first parameter will then be adjusted again with 'Ad Placement Setting 2'. After the second sample, two possible scenarios are shown. The first is assuming the CTR increased and the threshold has been reached. In this case, going into the 3rd sampling period, the second parameter will switch from 'Image ad setting' to 'Text ad setting'. The second case is if the CTR decreased as a result of the 'Ad Placement Setting'. In this case, a third change to the ad placement parameter will occur.

The tree based approach allows for a systematic approach to optimizing a set of ad parameters while recording the steps of the optimization process. The data can further be used to help optimization on the local and global levels as a set of ad parameters, analytics, and CTR results will be recorded into the database.

Figure 2: Ad Parameter Tree Example



IV. Global and Local Database

The main purpose of the global and local database is to keep track of the ad parameters with respect to the analytics and ctr. Since there can effectively be an infinite number of entries, a central codebook corresponding to the analytics can used to both reduce space, but serve as a way to consolidate similar analytics groups. Moreover, search algorithms that find matches based on the codebook entries can be improved. In addition, truncation can be performed by isolating ad parameter choices which do not reach a certain CTR

threshold. The global database will contain a set amount of data and retrieve optimized ad parameters from local databases. The purpose of the global database is to build a dictionary of optimized parameter settings which can be used during initialization of a campaign, finding the next optimized parameter setting, or interpolating statistics for further analysis.

Table 4 shows an example of a local database with codebook index reference to the central codebook of web analytics shown in Table 5.

Table 4: Example Local Database

Codebook Index	Ad Parameters		CTR
	placement	# of ads	
5	Setting 1	3ads	1.21%
2	Setting 4	3ads	1.11%
1	Setting 2	2 ads	.90%

Table 5: Central Codebook for Web Analytics

Index	Web Analytics	
	New/ Returning Visitors	Age
1	35%	19-24
2	32%	13-15
3	33%	25-34
4	48%	25-34
5	44%	25-34

Since the web analytics with respect to the ad parameters might have little variance to the CTR results after many iterations, it makes sense that they can be grouped to further optimize the database and algorithm. For example, we will use the example in Table 6 of the global database. Initially, the web analytics contains codebook index 5, 4, and 3. During an update of the database, a close match algorithm that takes into account the Ad Parameters and CTR will scan through the database to see if it can find any close matches. In this case, the analytic results from index 5 and 4 are similar. As a result, the codebook in Table 8 will update by merging the index parameters 5 and 4 together. Variance in the

two index parameters can be specified as a range or a mean average, which is the case shown. This update can further be recorded for statistical relevance or tracing back. Table 7 shows the results after the codebook and global database update. The CTR will eventually update itself accordingly after more sample points are recorded.

Table 6: Example Global Database

Codebook Index	Ad Parameters		CTR
	placement	# of ads	
5	Setting 1	3ads	1.21%
5	Setting 4	3ads	1.11%
4	Setting 1	3 ads	1.20%
4	Setting 4	2 ads	1.11%
3	Setting 3	2 ads	.87%

Table 7: Example Global Database after grouping

Codebook Index	Ad Parameters		CTR
	placement	# of ads	
5	Setting 1	3ads	1.21%
5	Setting 4	3ads	1.11%
1	Setting 3	2 ads	.87%

Table 8: Central Codebook for Web Analytics after Grouping

Index	Web Analytics	
	New/ Returning Visitors	Age
1	35%	19-24
2	32%	13-15
3	33%	25-34
5	46%	25-34

At the start, the algorithm will sample a wide range of analytic value sets and eventually group similar analytic value sets together based on a close match procedure. This method will further help to optimize the database both in size and search times, not to mention reduce the sampling redundancy and computation in the overall algorithm.

V. The Importance of Ad Parameter Settings

An actual simulation was not performed on the algorithm so simulation results are not discussed. However, optimization tips on the Google Adsense™ Blog, clearly reveal the importance of Ad parameter selection. Below is an excerpt from a posting that reveals tips for optimizing ads on a blog page for example:

1. Choose the right ad formats
   Because the typical blog layout uses a narrow section for posts, the medium rectangle fits nicely at the end of each post. The medium rectangle also supports image ads, increasing the competition and revenue potential for that ad space. For shorter posts, a banner may be a better fit.

2. Place ads where your readers will notice
   Blog main page:
   Whether the subject is gadgets, gossip, or a glimpse into the writer's personal life, readers are deeply engaged with the content of their favorite blogs. What could be more effective than displaying an ad precisely relevant to that content? An ad after each post can serve as a "commercial break" from reading or as an "action step" to take after reading.

   Individual blog entries:
   As individual post pages don't have a lot of content, the key is to place ads near your content without bombarding the page with ads. For long entries, try embedding a blended medium rectangle into your post and adding a banner at the end. For short entries, just place one medium rectangle or banner at the end.

   If there's a comment section after the post, place a banner or a horizontal link unit just above the comment box. Rather than using the traditional Skyscraper in your sidebar, consider using a link unit, which will offer a wider range of topics and may appear less obtrusive.

3. Improve targeting
   How can you ensure that your ads are relevant to a specific post? With section targeting, you can target an ad unit to a specific section of the page, as well as block out irrelevant sections such as the navigational links.

4. Customize your ad colors
   For more seamless integration with the content, blend ad units into the background of your blog. Choose a bold color for the ad title to help draw attention to your ads while ensuring that users don't confuse ads with content.

It is evident from the optimization tips that ad parameters including the placement and number of ads per webpage play a dominant role as to effectiveness of the

ads for a given page. With that said, the adaptive algorithm will be able to increase the overall CTR of ads of a given webpage by varying the ad display parameters. Based on further blog postings, there are examples of where CTR dramatically improved for webmasters by having simply switched around several ad parameters. This again reaffirms the importance of ad parameters and the final CTR result.

VI. Conclusion

The adaptive advertising algorithm takes the components discussed above to figure out the most optimized ad parameters to serve to a given analytic group or audience to maximize CTR. The algorithm is self learning once initial weights and parameter sets are provided and can automatically adjust and update itself at set intervals. Ad parameters are dynamically adjusted according to relative sensitivities and by using a tuning matrix which is based on a multi-parameter optimization procedure. Routes to figure out the best ad parameter are traced by the ad parameter tree which provides a systematic approach to optimizing ad parameters for a given analytic group. Moreover, the global database can help to quickly seek the next best parameter set or provide interpreted statistics for other functions. By using a central codebook for the analytics, analytic groups that meet the best match specifications will be merged together; thus, allowing the database to operate more efficiently on the local and global level.

Recording web analytics with the ad parameters and CTR results provides a webpage running the adaptive algorithm a means of serving a different set of ad parameters to a different audience base or analytic group. For example, new users to a site may not be familiar with the site navigation and randomly click on a blue ad link. In contrast, a frequent user may have ‘ad blindness’ and more compelled to click on a bright green ad that is placed above the navigation fold. By using analytic results, the new user and returning user can be displayed a different set of ad parameters. Since the advertising algorithm will adaptively adjust ad parameters when a page is served, the webmaster will never have to manually ‘tweak’ the ad codes.

In conclusion, an adaptive advertising algorithm was proposed to maximize CTR by adaptively adjusting ad parameters based on web analytic statistics. The algorithm takes into account the various correlations and weights of various parameters and provides a self updating and optimizing database

scheme. The goal of providing Google Adsense™ users optimized ad display without any manual guess work can be accomplished.

Appendix

Bibliography:

Multi-parameter optimization in WCDMA radio networks
Houtao Zhu; Buot, T.
Vehicular Technology Conference, 2004. VTC 2004-Spring. 2004 IEEE 59th
Volume 4, Issue , 17-19 May 2004 Page(s): 2370 - 2374 Vol.4

An Adaptive Algorithm for Selecting Profitable Keywords for Search-Based Advertising Services
Rusmevichientong, P; Williamson, D
Cornell University
http://delivery.acm.org/10.1145/1140000/1134736/p260-rusmevichientong.pdf?key1=1134736&key2=8087457711&coll=GUIDE&dl=GUIDE&CFID=17368788&CFTOKEN=14025181

The Expectation Maximization Algorithm: A short tutorial
Borman, S.
Last updated October 14, 2006
http://www.csd.uwo.ca/courses/CS840a/Papers/EM_algorithm.pdf

Ad Blindness
Lawrence, A.P.
http://aplawrence.com/foo-web/ad-blindness.html

General Reference: http://www.wikipedia.com

Links:

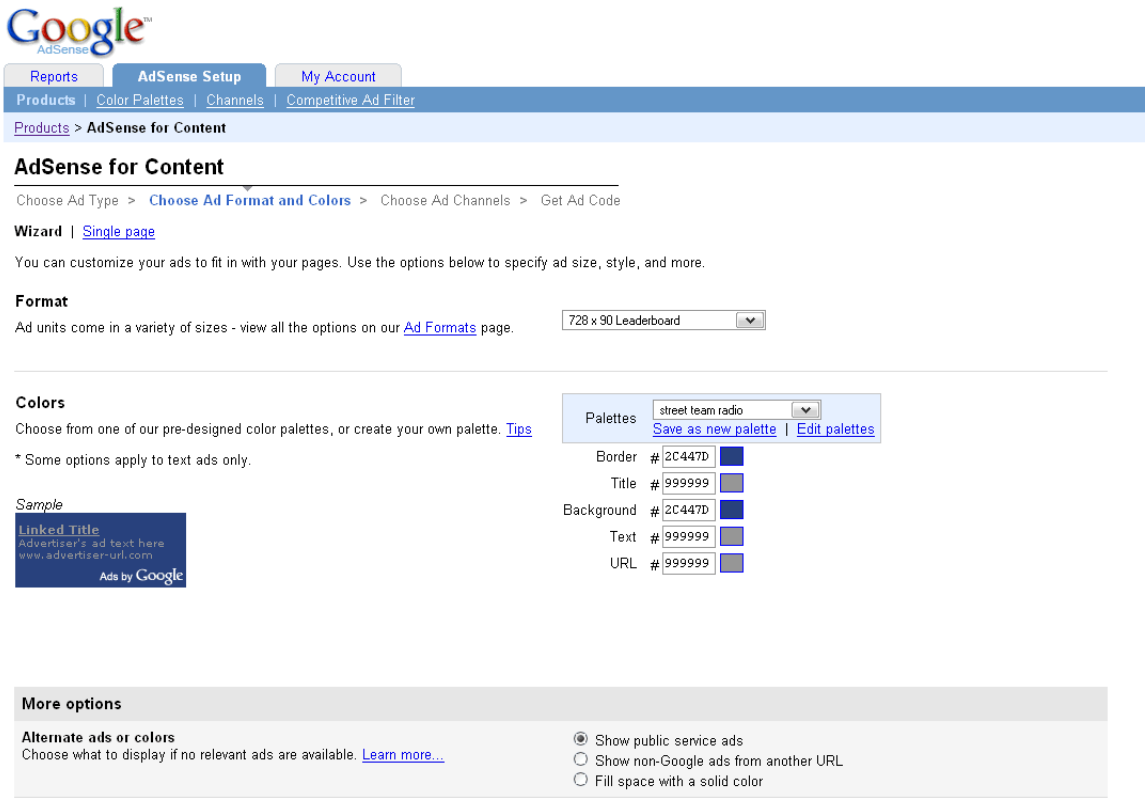
Google Adsense™ – http://www.google.com/adsense
Google Analytics™ – http://www.google.com/analytics
Google Adsense™ Blog w/ Optimization Tips - http://adsense.blogspot.com/

Definitions:

Click Through Ratio / CTR
The relationship between the number of views and clicks of an image or images in a campaign. The number of clicks is divided by the number of views and is expressed as a percentage. CTR is used as one gauge to judge the relative effectiveness of an ad campaign. For example, if an image is displayed 100 times and is clicked twice; the CPM is 2.00%. A 2% CPM is better than industry average.

CPM, CPC, CPA
CPM (Cost per 1,000), CPC (Cost per Click) and CPA (Cost per Action) are methods of calculating the charge for pages (advertisements) being served. CPM is a holdover from traditional media advertising, and does not take advantage of the Hypertext nature of the medium. It charges purely on the number of times the advertisement is served. It does account for branding effects, that are not accounted for in the other models. CPC is a cost associated with each click on the advertisement to the target page. CPA is a cost associated with each lead created from a click on the advertisement (CPL), or each sale (CPS).

Appendix – Figure 1: Google Adsense Advertisement Setup



Appendix – Figure 2: Visualization of Ads on Webpage