On the Benefits of Keyword Spreading in Sponsored Search Auctions: An Experimental Analysis

M. Budinich[1] B. Codenotti[2] F. Geraci[2] M. Pellegrini[2]

[1] Institute for Advanced Studies, Lucca, Italy,m.budinich@imtlucca.it
[2] Istituto di Informatica e Telematica, CNR – Consiglio Nazionale delle Ricerche, 56100 Pisa, Italy, bruno.codenotti,filippo.geraci,marco.pellegrini@iit.cnr.it

Abstract. Sellers of goods or services wishing to participate in sponsored search auctions (SSA) must define a pool of keywords that are matched on-line to the queries submitted by the users to a search engine. Sellers must also define the value of their bid to the search engine for showing their advertisements in case of a query-keyword match. In order to optimize its revenue a seller might decide to substitute a keyword with a high cost, thus likely to be the object of intense competition, with sets of related keywords that collectively have lower cost while capturing an equivalent volume of user clicks. This technique is called *keyword spreading* and has recently attracted the attention of several researchers in the area of sponsored search auctions. In this paper we describe an experimental benchmark that, through large scale realistic simulations, allows us to pin-point the potential benefits/drawbacks of keyword spreading for the players using this technique, for those not using it, and for the search engine itself. Experimental results reveal that keyword spreading is generally convenient (or non-damaging) to all parties involved.

1 Introduction

A very large fraction of consumers use search engines to find information on the web about goods and services before deciding whether to purchase them in the online markets. Search engines take advantage of their key position on the Web to sell advertising space to economic players on search result pages. Indeed, over the last few years, sponsored search advertising has become the dominant source of profits for search engines. Typically sponsored search results appear in two separate parts of the page above and to the right of the results returned by a search engine. Sponsored search results include a title, a short text, and a link referring to a Website. Advertising space comes in the form of slots, which are sold by auctions. When a user submits a given keyword in a query to a search engine, an auction is run among all the advertisers submitting bids for that keyword. The advertisers who wish to display their ads against the search for a keyword participate in the auction by specifying their valuation and a daily budget to the search engine. The search engine could use various mechanisms for determining winners and payments, the most popular mechanism being the generalized second price (GSP) auction.

Although GSP looks similar to the classical Vickrey-Clarke-Groves (VCG) mechanism [31, 10, 18], its properties are very different, i.e., truth-telling is not an equilibrium in GSP [12, 30]. Over the last years, several papers of computational flavor have appeared, touching in different ways this paradigm of online advertising, see, e.g., [5, 6, 12, 20, 21]. From the viewpoint of a search engine, the *adword problem* consists of assigning a sequence of search keywords to a set of competing bidders, each with a daily spending limit, with the goal of maximizing the revenue generated by these keyword sales. This problem generalizes on-line matching, and this connection has been exploited in [23]. A central problem in adword markets from the point of view of a seller of goods and services is the generation of keywords. Advertisers typically prefer to bid for keywords that have high search volumes; however they may be very expensive, so that it might be reasonable to bid instead for several related and low volume, inexpensive terms that generate roughly the same amount of traffic altogether. Some preliminary work exploring this idea has been done in [1], where however, the emphasis is on the algorithmic aspects of keyword generation, not on the global market phenomena as in the present work.

In this paper we describe a large scale simulator for analyzing the effect of using synonyms for keyword spreading in sponsored search auctions (SSA), and collect a number of evidences about the effects of this strategy. Our simulations involve up to 2M agents bidding for words from a pool of 36K words and 3M queries per experiment (more details in Sections 2 and 3). Our experiments point to the following conclusions:

– using synonyms increases the revenues for all players in the market (Figs. 6a, 7a, 7b); in particular the early adopter agents benefit the most (Figs. 7a, 7b)
– using a VCG payment scheme decreases the agents' benefits with respect to using GSP while not much changes for the search engine (data omitted for lack of space, see [8])
– as the fraction of agents using synonyms increases, the search engine revenues are not significantly affected (Fig. 6b) as well as the costs for the agents not using them (Fig. 8a) while the agents using synonyms have decreasing gains (Fig. 8b)
– budget depletion strategies are shown to rarely be beneficial for the agents, while always increasing the revenue for the search engine, even in presence of keyword spreading (data omitted for lack of space, see [8]).

A problem related to *keyword spreading* is that of *keyword selection* [29], where the economic players try to select at fixed rounds the subset of keywords that maximize revenues while trying to learn basic parameters (such as keyword click-through rates) during the repeated bidding processes. Note that here the viewpoint is that of a single player and that the market, as seen by the seller, is modeled via (known or unknown) time varying probability distributions. In contrast, in our simulations keywords are selected by the agents off-line. We simulate directly the market and the auctions by using a large number of atomic agents each performing simple actions. Previous research on agent-based simulation of adwords markets by Mizuta and Steiglitz [24] was centered on studying the interaction of different classes of players according to their bidding time profiles, e.g. early vs late bidders. Kitts and LeBlanc [19] describe a large scale simulator for adwords markets to investigate several bidding strategies, e.g. random bidding vs. bid to keep relative position, which however do not involve keyword spreading. The architecture of a large scale SSA is described in [4], where it is applied to compare several ranking, pricing and budgeting policies. To the best of our knowledge this is the first large-scale agent-based simulation of the market effects of keyword spreading. The time horizon of our simulations is one bidding day. For this reason we consider as fixed all features whose rate of change is so slow so that it can be approximated by a constant within the time span of one day (e.g. the number of bidder is fixed at the beginning of the day, and their number decreases only when they run out of budget). Other features that can vary with a faster dynamic are modeled as distributions (or with an adaptive behavior), but the parameters of the distribution itself are considered as having a much slower dynamics, therefore such parameters are fixed within the one day time frame.

The rest of this paper is organized as follows. In Section 2 we briefly describe the clustering technique used to build a dictionary of synonyms. In Section 3 we highlight the architecture of our market simulator. In Section 4 we show the main outcomes of our experiments.

2 Keyword Spreading

We explore two alternative ways of performing keyword spreading. One uses the well known Wordnet ontology, the second is based on clustering web pages related to a query as found by a generalist search engine (in our case Google). The two resulting word distributions are different but the measured trends are consistent for both data sets, thus giving high confidence in the robustness of the experimental benchmark.

Wordnet The most important project for ontologies of words is *WordNet* [26]. Originally proposed by the Cognitive Science Laboratory at Princeton University only for the English language, WordNet has become a reference for the whole information retrieval community, and similar projects are now available in many other languages. WordNet is a handmade semantic lexicon that groups words into sets of synonyms called *synsets*. Intuitively one can replace a word in a text with another from the same synset without changing its semantics. A word can appear in more than one synset if it has more than one meaning. Synsets are arranged as nodes in a graph such that there is an edge to connect two nodes if there is a relation between the two synsets. There are different types of possible relations, an exhaustive list of them can be found in the WordNet web site [25]. Given two synsets X and Y, the most common types of relations in WordNet are: *hypernym* if every X is a "kind of" Y, *hyponym* if Y is a "kind of" X, *holonym* if X is a part of Y and *meronym* if Y is a part of X. In our experiments we took into account only hypernym. Note that this relation is *asymmetric*.

Clustering Google Data Given a query word, our goal is to find a set of semantically related words whose cost is lower than those of the query. We are not only interested in paradigmatic similarity, i.e., when two words may be mutually exchanged without effects on the semantics of the text, but also to syntagmatic similarity, i.e., when two words significantly co-occur in the same context. To achieve this goal we approach the problem as a *word clustering* [11, 22] task. Given a set of objects, clustering attempts to create a partition such that the objects in a cluster are related among them, while objects in different clusters are unrelated. Word clustering requires a corpus of documents related to the query word. To set up such a corpus we redirect the query to *Google* and download pages related to the first 100 results. Each page is later parsed and split to extract a set of sentences. Under the well established hypothesis that co-related words are more likely to stand in the same sentence, all the sentences not containing the query are discarded. We remove from each sentence over-represented words (stop words) that are often "syntactic sugar" and their removal does not affect the semantic content of the sentence. We added to the standard stop word list, a set of words that normally can not be considered stop words, but in the Web environment are considered generic (e.g. "download"). Once filtered, all the sentences are arranged in a term-document matrix whose rows correspond to sentences and whose columns to terms of the corpus. We tested different weighting schemes for terms, and we found that for our purpose a simple binary weighting scheme suffice. For clustering we employed a fast implementation of the FPF clustering algorithm [16] because of its good trade off between speed and accuracy [14]. As distance between pairs of words, i.e., columns of the term-document matrix, we used the well known cosine similarity. FPF is an iterative algorithm. It makes a new cluster at each iteration and populates it by extracting from the other clusters all the elements that are more related to the new cluster. The procedure stops when a given number k of clusters is reached. For word clustering it is impossible to predict in advance a good value for k. The typical approach, with methods such as k-means, is to make a certain number of independent clusterings with different choices of k and select the most appropriate a posteriori. Instead, the iterative nature of FPF allows us to not feeding the number of clusters in advance but check a more appropriate termination condition at each iteration. In our case, at the end of iteration t, FPF checks the cluster $C_t(q)$ containing the query. When the number of elements of this cluster gets below a certain threshold (10 in our case) the algorithm stops and returns, among $C_t(q)$ and $C_{t-1}(q)$, the set whose cardinality is closest to the threshold. This procedure ensures that we find a coherent cluster of words even if the query is not central in that cluster. Note that this relation may be *asymmetric*, although in a subtle way, since different sets of snippets are processed for each query word. Stretching the terminology, we will call words for which the relations defined above hold (over WordNet and Google data) "synonyms", for lack of a better name, however one should keep in mind that the relations we model is more complex.

3 The Simulator

The starting point in designing the simulator was the collection of some publicly available data on ad auctions, including:

– a large and representative set of words,
– an estimate of the cost of each word,
– an estimate of the number of clicks received by each word.

The Word List The simulator uses a finite set of words; these words represent all the possible queries that a user can make to the search engine and also all the possible keywords an advertiser can bid on. The core of the word list has been taken from the SCOWL (`http://wordlist.sourceforge.net/`) project (an open source project that maintains a set of word lists for use by spell checkers), and consists of 35867 entries.

The Traffic Estimator Google maintains an on-line tool (the AdWords Traffic Estimator Sandbox,[1]) developed to aid advertisers in their campaigns. The Traffic Estimator, given a keyword, displays its estimated cost per click (CPC) and the estimated number of clicks per day. The simulator uses this data to estimate some quantities that would otherwise be difficult to generate realistically. Although, as Google itself warns, the data is to be considered only as a guideline, it is of great help for our purposes. The estimated CPC is used in the simulator (averaging the two values given by the Traffic Estimator) as a basis to assign a "real" value to each keyword. The simulator successively employs these values as parameters to generate the agents' bids and valuations. Clearly the estimated CPC of a term is different from its "real" value. If we were to measure the estimated CPCs in the simulator at the end of a run they would certainly be different from the ones supplied by the Traffic Estimator. Nonetheless their distributions and main features would be similar, and that is enough for the use we make of it.

[1] `https://adwords.google.com/select/TrafficEstimatorSandbox`

The other parameter that is central to the simulator is the estimated number of clicks per day of each word. Since the simulation considers only the queries that give rise to a click, we can simply consider the estimated number of clicks per day as the distribution of the queries in the simulator. We collected such data for each of the 35867 entries in our dictionary, building a small database that constitutes our initial data set. Table 1a summarizes the main characteristics of the data set. For completeness, we plotted the data collected from Google's Traffic Estimator. Figure 1a is the distribution of the estimated clicks per day, while Figure 1b shows how estimated average costs per click are distributed.

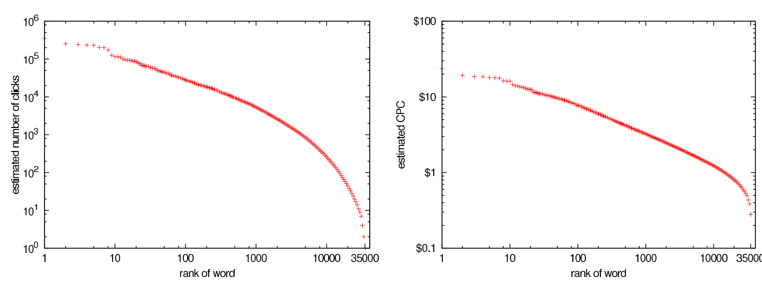
Number of words	35867
Max. clicks per day	349216
Min. clicks per day	1
Max. CPC	\$23.6
Min. CPC	\$0.05

(a) Statistics from the CPC and Click Volumes

	clustering	Wordnet
Words with synonyms	18660	12271
Max. synonyms for a word	13	441
Max. terms a word is synonym of	668	146

(b) Statistics from the two synonyms databases: Wordnet and clustering of Google snippets

Table 1: Statistics of the click and synonym databases.

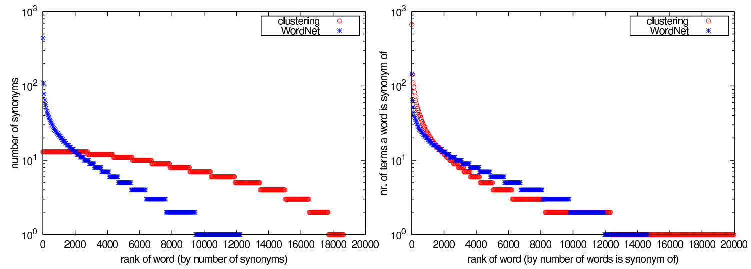


(a) estimated number of clicks (b) estimated CPC

Fig. 1: Data gathered from Google's Traffic Estimator, in log-log scale.

We want to use our simulator to investigate the behavior of the ad auction mechanism in the presence of agents who make use of keyword spreading. To model such behaviors we need a set of synonyms for each word. The clustering algorithms described in Section 2 produced a list of synonyms for each word. As a reference we have also created a similar list by querying the Wordnet[2] database. Table 1b gives some basic figures on the two resulting data sets, while Figure 2a and Figure 2b show the distribution of the number of synonyms per word and the distribution of the number of terms a word is synonym of.

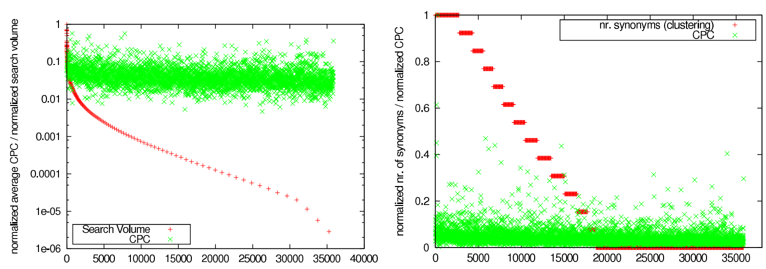
[2] `http://wordnet.princeton.edu/`



(a) number of synonyms per word (b) number of words a term is a synonym of

Fig. 2: Comparison of the two synonyms databases in log-scale.

There is a big difference in the boundary values of the databases: for example, there is a term for which Wordnet gives 441 synonyms; but more important is the difference in the rank distribution (see Fig. 2a). Due to limitations in the computational resources, the clustering imposed a hard limit of 13 on the maximum number of synonyms per word. Nonetheless, as shown clearly by Figure 2a, the majority of the words have more synonyms in the clustering database than in the Wordnet one. Overall we can consider the databases comparable for our purposes, and the experimentally detected trends are consistent in both databases. Starting from a list of words, we have expanded it with various information: prices, number of clicks and synonyms. It seems now a natural question to ask if there is any correlation between these quantities. As a first guess it might seem reasonable to expect at least some correlation. That is, we might expect that some "popular" words receive many clicks and have a high price. Or that words that receive a lot of clicks also happen to have many synonyms. Somewhat surprisingly, an empirical analysis gives a negative result. At a first glance the data set exhibits virtually no correlation between the different values. To give a rough idea of this result we present just two plots, all the other ones being extremely similar. Figure 3a ranks words by estimated number of clicks, and shows these values along the CPCs (both normalized). It looks like there is no order in the CPC values; they appear as if uniformly distributed.



(a) nr. of clicks and CPC (log-scale) (b) nr. of synonyms and CPC

Fig. 3: Comparison of the two synonyms databases used.

Figure 3b, instead, ranks the words by the number of synonyms they possess, using the Wordnet database, and displays this value along the estimated CPC (again, normalized). As it seems apparent there is no correlation between these quantities. All the other comparisons, e.g. CPCs versus click volume, synonyms versus CPC using the clustering database, give similar results.

All the simulations were carried out using the same static set of agents. To this end, the set of agents was generated once and for all and saved in a file. Its main characteristics are presented in Table 2a. In what follows we will refer to this fixed set of agents, words and synonyms as our data set.

Number of Agents	$2 \cdot 10^6$
Max. bids on a single word	21446
Min. bids on a single word	21
Max. bids per agent	3000
Min. bids per agent	3
Budget Range	[\$1 − \$max. avail.]
Bid Range	[\$0.01 − \$200]
Nr. of slots	4
Clickthrough probabilities	$0.6, 0.25, 0.10, 0.05$

(a) Statistics on the pool of bidding agents

Keyword	"reviews"
"Real" value	\$1.045
Estimated nr. of clicks	12029
Nr. of interested agents	11023
Max. bid	\$3.064
Min. bid	\$0.010
Max. difference $v_i - b_i$	12.5% of v_i

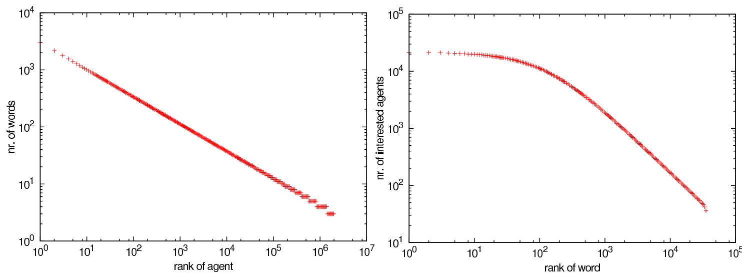
(b) Statistics on the bids for the keyword "review"

Table 2: Statistics for the bidding agents and bids on a typical keyword.

Each agent bids on a number of different keywords. If we consider all the agents, these numbers of allocated keywords are distributed as a power law, whose parameters are based on the number of agents, such as to keep a fixed maximum and minimum (to avoid cases in which an agent bids on all of the words, or cases in which there are agents that have not bid on any word at all). Figure 4a plots these values for the data set. The choice of a power law distribution to model the keyword-to-agent distribution is justified by an analogy with real data in the version 1.0 of *Yahoo! Search Marketing advertising bidding data*[3], used also in [7]. The distribution described in [7] fits qualitatively a power law. The real data come for an anonymised log of bids for 1000 keywords with about 10,000 bidders collected in the period 2002/2003, where data was truncated at 50 keywords-per-agent. In order to perform a larger simulation ($2 \cdot 10^6$ bidders, $36 \cdot 10^3$ keywords) we have correspondingly scaled up the power-law curve so to have the number of words-per-agents in a range from a few units to about two thousand.

Another quantity characterizing agents is their budget. We have chosen a uniform distribution with budgets in the range [\$1 − \$100]. The choice of a uniform distribution for the budget-to-agent distribution comes from a series of rather indirect arguments. We could not find any such distribution described in literature, or in publicly available data sets, probably due to the sensitive nature of such data. A few papers that use such a distribution in simulation (e.g. [13] [3]) give no clue as to its shape. Anecdotic remarks [15] report typical budgets are in the orders of hundreds of dollars. A theory of sponsored search

[3] http://webscope.sandbox.yahoo.com/



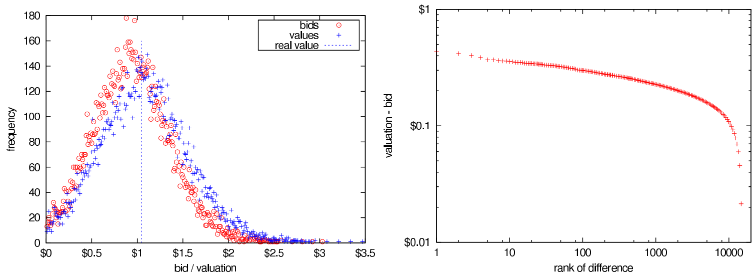
(a) words per agent (b) agents per word

Fig. 4: Agents per word and words per agent in log-log scale.

auctions for markets with budget constraints has been developed in recent years, and often the budget distribution among bidders is left as a free parameter of the theory. An interesting paper of Z. Abrams [2] describes a theory for revenue maximization where a critical parameter is the *bidder budget dominance*, that is the fraction of the total market budget that is allocated to the bidder with the highest budget. The use of a uniform budget distribution in the range $[1, .., 100]$ is consistent with the above considerations. While the ratio of the highest to the lowest bidder can be as high as 100, the bidder budget dominance is very low (at most 10^{-6}). Experiments with a power law distribution give almost identical results. We conjecture that the results we present are qualitatively analogous for any other distribution that has a similar low budget dominance and maximum budget ratio, even if not uniform.

Words The words come from an open-source dictionary created for the spell-checkers. To each entry we have added the following two pieces of information: estimated number of clicks, and estimated CPC, which we obtained from the tool made available by Google. Given the dictionary, we assign the words to the bidders in such a way that both the number of bidders per keyword and the number of keywords per bidder be distributed according to a power-law. Having fixed the number of words an agent will bid on, the next step is to select them from the dictionary. The simulator does so, and the resulting values (i.e., the number of agents interested in every word) are again distributed as a (different) power law. The parameters controlling such distribution are chosen as to avoid unrealistic scenarios. Figure 4b shows the number of interested agents per word in our data set. As described above, each word is assigned a "real" value based on the data gathered from the Traffic Estimator. Based on this reference value, each agent i will then compute its personal valuation v_i for the keyword. The distribution of the valuations for each agent is a normal distribution whose mean is precisely the "real" value of the word. To increase the variety among agents, each agent has a different variance associated to this normal distribution. Figure 5a shows the distribution of valuations for different agents interested in the same keyword, i.e. "reviews". The agent valuation v_i represents the agents's *Return on Investment* for a click of his advertisement, and it is a private information not disclosed to any of the partners in the auction (either SE or other players), thus difficult to infer from any collected data set where only the bids are known.

As a final step each agent i must generate a bid b_i. Bids are generated according to the agent's valuation v_i. Only bids such that $b_i < v_i$ are considered, and such that they do not exceed the residual budget of agent i. The quantity $v_i - b_i$ for a given keyword and agent i is proportional to the payoff (or utility)



(a) bids and valuations (b) difference valuations - bids

Fig. 5: The bids and valuations for the word “review”, whose ”real” value is 1.045.

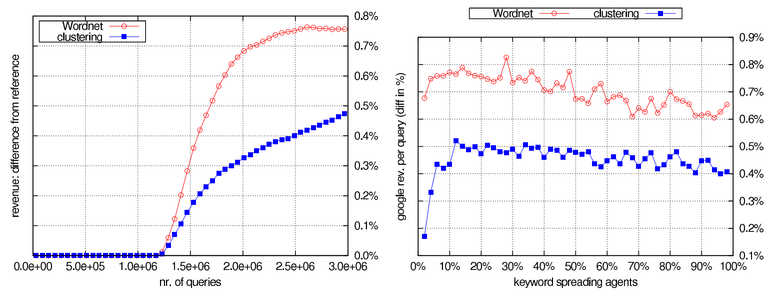
of agent i in a generalized first price auction (where each agent pays the amount of his own bid), thus it is an overestimate for the payoff in a GSP auction. Much research has tackled the scenario of a single agent optimizing the bid b_i in an adaptive manner so to maximize revenue, and the game-theoretic properties of such strategies. We simulate both adaptive and non-adaptive bidding strategies. In [17] two basic bidding strategies are described, the first strategy is to bid high enough so to increase the chances of securing a good rank, the second is to lower the bid so to increase the payoff in case the auction is won. For the non-adaptive case we sample the bid value from a power-law distribution that represents a probabilistic mixed strategy that pursue both goals at once in a balanced manner. Thus we choose to generate the differences $v_i - b_i$ according to a power law distribution. For the adaptive case, we have implemented the equilibrium converging strategy described in [9]. The two cases give similar outcomes and we report the non-adaptive results.

4 Experimental Results

The simulations in this section were all run under the GSP mechanism. Moreover, the keyword spreading mechanism is applied after the first 20% of the queries have been processed. Fig. 6a shows the increment in search engine revenue between a basic simulation (in which no agent ever changes keywords) and one where we allow 20% of the agents to apply keyword spreading, using both the Wordnet database and our clustering techniques. The difference levels-off with a gain of 0.5 to 0.8 of a point. Given the total value of the adwords market this difference is very significant in absolute terms.

In Fig. 7a we show the revenue increase for the agents that are allowed to change words (20% of total), and in Fig. 7b for those that are not allowed (80% of total). For both groups the revenue increase is positive and levels-off, with the agents in the first group having better performance (in the range 3.0%-4.5%).

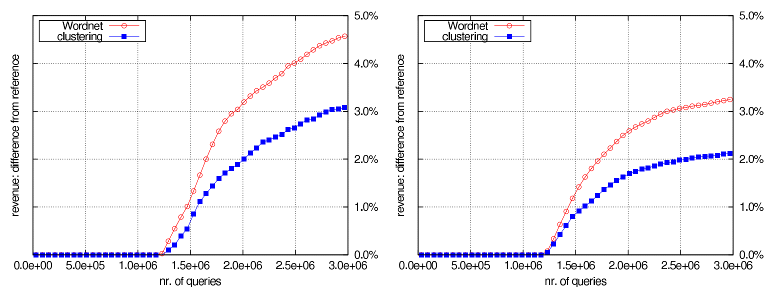
Figures 6b, 8a, 8b show the variation in revenue increase when we vary the fraction of users using keyword spreading from 5% to 95% of the total. While the increased revenue for the search engine and for non-keyword spreading agents is



(a) Search engine's revenue when 20% of the agents change their keywords with synonyms (Wordnet and clustering data).

(b) Search engine's revenue increase for varying fraction of keyword spreading agents.

Fig. 6: Revenue increase for the Search Engine.



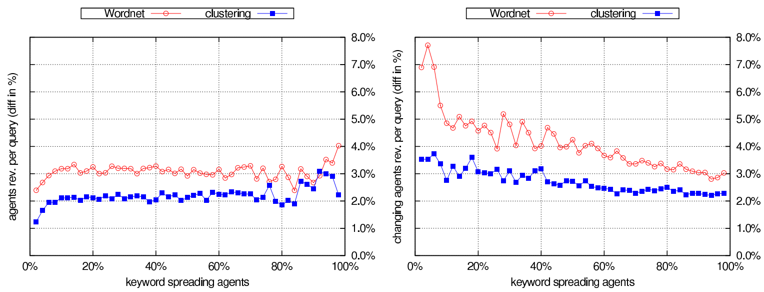
(a) keyword spreading agents

(b) non keyword spreading agents

Fig. 7: Revenue increase for agents.

hardly affected, we notice a clear effect of diminishing marginal gain for keyword-spreading agents, with initial gains up to 7% for early adopters, and just 2.5% when the practice is widespread.

We also explore the effect of a class of strategic behavior called *budget depletion strategies*. Here for 20% of the words the top bidder of each such word that does not obtain an advertisement slot will switch to a policy of increasing its bid so to deplete the competition's budget faster, without incurring any additional cost. We explore two cases (a) called "unrealistic" where the strategic agent knows the optimal new bid value, and (b) called "realistic" where the optimal new bid value is sought by small successive increments. SE do not seem to suffer in revenue from this type of strategic agents. In the realistic case, the strategic agents may have to bid above their own valuation, thus risking a negative payoff. In our simulation (data not shown, see [8]) this effect overweights the potential gain from the depletion of competitors' budget.



(a) non keyword-spreading agents (b) keyword-spreading agents

Fig. 8: Agent's revenue increase for varying fraction of keyword spreading agents.

5 Conclusions

Keyword spreading in sponsored search auction is a technique aiming at extracting more value from the long tail of the distribution of user queries volumes. We performed a simulation with a large number of bidding agents and keywords to expose the possible benefits of this technique for all players involved. We conclude that there are non-negligible economic benefits for the search engine running the auction, and for the bidding agents. There is also a competitive advantage for early adopters. Our simulations are based on publicly available data, on educated guesses as to the shape of some relevant distributions, and on a very large numbers of agents, keywords and queries involved. As a word of caution, we remark that the model for each single agent we employed is rather simple (both adaptive and non-adaptive). Thus, although our results are interesting for search engine companies and bidders, we view them a preliminary investigation. As future research we plan to use more sophisticated user and bidder behavior models (for example by replacing some distributions with "profiles" deducted from real data) so to confirm our findings in a scenario that is more complex in terms of the repertoire of possible individual behaviors. A second future line of research will use more complex market segmentation models and investigate how such models can influence the outcomes of the simulations.

References

1. V. Abhishek, K. Hosanagar. Keyword Generation for Search Engine Advertising using Semantic Similarity between Terms, Proc. ICEC 2007.
2. Abrams, Z. 2006. Revenue maximization when bidders have budgets. In Proceedings of the ACM-SIAM SODA '06. (2006).
3. Abrams, Z., Mendelevitch, O., and Tomlin, J. 2007. Optimal delivery of sponsored search advertisements subject to budget constraints. In Procof the 8th ACM EC, San Diego, California (2007). 272-278.
4. Acharya, S., Krishnamurthy,P., Deshpande, K. , Yan, T.W., and Chang. C.C.. A Simulation Framework for Evaluating Designs for Sponsored Search Markets. WWW2007 Sponsored Search Auctions Workshop.
5. G. Aggarwal, A. Goel, and R. Motwani. Truthful auctions for pricing search keywords. In Proceedings of the 7th ACM EC, pages 1-7, 2006.
6. S. Amer-Yahia, S. Lahaie, and D. M. Pennock. Towards a generic bidding standard for online advertising. In Fourth Workshop on Ad Auctions, 2008.

7. Auerbach, J., Galenson, J., and Sundararajan, M. 2008. An empirical analysis of return on investment maximization in sponsored search auctions. In Proc of ADKDD '08 Las Vegas, Nevada. pp 1-9.
8. M. Budinich and B. Codenotti and F. Geraci and M. Pellegrini. On the benefits of keyword spreading in sponsored search auctions: an experimental analysis,2009, Institute for Informatics and Telematics of CNR, TR IIT-10/2009.
9. M. Cary, A. Das, B. Edelman, I. Giotis, K. Heimerl, A. R. Karlin, C. Mathieu, M. Schwarz. On Best-Response Bidding in GSP Auctions National Bureau of Economic Research Working Paper No. 13788. February 2008.
10. E. H. Clarke. Multipart pricing of public goods. Public Choice, 11:17-33, 1971.
11. I. S. Dhillon, S. Mallela, R. Kumar, Enhanced word clustering for hierarchical text classification, Proc. of 8th ACM SIGKDD pp. 191-200 (2002).
12. B. Edelman, M. Ostrovsky, and M. Schwarz. Internet advertising and the Generalized Second Price auction: Selling billions of dollars worth of keywords. American Economic Review, 97(1):242-259, 2007.
13. Feldman, J., Muthukrishnan, S., Pal, M., and Stein, C. 2007. Budget optimization in search-based advertising auctions. In Proc. of the 8th ACM EC, San Diego, California. June. 40-49.
14. Geraci, F., Pellegrini, M., Pisati, P., and Sebastiani, F. 2006. A scalable algorithm for high-quality clustering of web snippets. In Proc. ACM SAC '06. Dijoin, France.
15. R. Gonen, E. Pavlov. 2007. An Adaptive Sponsored Search Mechanism *delta*-Gain Truthful in Valuation, Time, and Budget, Proc. of WINE 2007, pp. 341-346.
16. T. F. Gonzalez, Clustering to Minimize the Maximum Intercluster Distance, Theor. Comp. Sci. 38 (2/3), pp.293-306 (1985).
17. Ghose, A. and Yang, S. 2008. An empirical analysis of sponsored search performance in search engine advertising. In Proc. of the Intl. Conf. on Web Search and Web Data Mining WSDM '08. New York, NY, 241-250.
18. T. Groves. Efficient collective choice when compensation is possible. Review of Economic Studies, 46:227-241, 1979.
19. B. Kitts, B. J. LeBlanc: A Trading Agent and Simulator for Keyword Auctions. Proceedigns of AAMAS 2004: 228-235
20. S. Lahaie. An analysis of alternative slot auction designs for sponsored search. In Proc. of ACM EC, pages 218-227, 2006.
21. S. Lahaie, D. M. Pennock, A. Saberi, and R. V. Vohra. Sponsored search auctions. In Algorithmic Game Theory, chapter 28, pages 699-716. Cambridge University Press, 2007.
22. H. Li and N. Abe, Word clustering and disambiguation based on co-occurrence data, Proc. of the 17th Intl. Conf. on Computational Linguistics, pp. 749–755 (1998).
23. A. Mehta, A. Saberi and U. Vazirani. Adwords and Generalized Online Matching, J. of ACM 54(5), 2007.
24. Mizuta, H. and Steiglitz, K. 2000. Agent-based simulation of dynamic online auctions. In Proc. of the 32nd Conference on Winter Simulation (Orlando, Florida, December 10 - 13, 2000). San Diego, CA.
25. G. A. Miller, C. Fellbaum, R. Tengi, P. Wakefield, R. Poddar, H. Langone, B. Haskell, WordNet: A Lexical Database for English, Princeton University (2006).
26. G. A. Miller, WordNet: An On-line Lexical Database, Int. J. of Lexicography (1990)
27. M. Richardson, E. Dominowska, and R. Ragno. Predicting clicks: estimating the click-through rate for new ads. Proc. WWW 2007, pages 521-530, 2007.
28. Robu, V., Bohte, S. and La Poutre, J. A. The Complex Dynamics of Sponsored Search Markets. Agents and Data Mining Interaction, pp. 183-198, 2009
29. P. Rusmevichientong, D. P. Williamson: An adaptive algorithm for selecting profitable keywords for search-based advertising services. ACM EC 2006: 260-269
30. H. R. Varian. Position auctions. Int. J. of Industrial Organization, 25:1163-1178, 2007.
31. W. Vickrey. Counterspeculation, auctions and competitive sealed tenders. J. of Finance, 16:8-37, 1961.