Sponsored Search Ad Selection by Keyword Structure Analysis

Kai Hui[1], Bin Gao[2], Ben He[1], and Tie-jian Luo[1]

[1] University of Chinese Academy of Sciences, Beijing, P.R. China
huikai10@mails.ucas.ac.cn,
{benhe,tjluo}@ucas.ac.cn
[2] Microsoft Research Asia, Beijing, P.R. China
bingao@microsoft.com

Abstract. In sponsored search, the ad selection algorithm is used to pick out the best candidate ads for ranking, the bid keywords of which are best matched to the user queries. Existing ad selection methods mainly focus on the relevance between user query and selected ads, and consequently the monetization ability of the results is not necessarily maximized. To this end, instead of making selection based on keywords as a whole, our work takes advantages of the different impacts, as revealed in our data study, of different components inside the keywords on both relevance and monetization ability. In particular, we select keyword components and then maximize the relevance and revenue on the component level. Finally, we combine the selected components to generate the bid keywords. The experiments reveal that our method can significantly outperform two baseline algorithms on the metrics including recall, precision and the monetization ability.

Keywords: ad selection, entity relationship, sponsored search.

1 Introduction

Sponsored search is the main monetization source of the commercial search engines. The ads, which are generated by the advertisers, are selected by the sponsored search system and displayed along with the organic search results according to queries issued by the users. Specifically, the advertisers use a few (bid) keywords, short phrases with several terms (e.g., *used toyota sedan 2005*), with bid prices for the ads. When a query is issued, the ad selection algorithm picks out a relatively small group of keywords with which we can get a group of ads according to the existing bids. After the ad selection, the sponsored search system estimates the click probability of each selected ad, and then displays the top-ranked ads by descending order of the product of the estimated click probability of the ads and the bid price of the keywords triggering the ads [7]. If the user clicks an ad, the corresponding advertiser will pay the search engine a certain amount of money according to the generalized second price auction [6]. Thus, the input of ad selection is a query from user and the output is a group of

P. Serdyukov et al. (Eds.): ECIR 2013, LNCS 7814, pp. 230–241, 2013.

Table 1. An Example of a Clicked Ad in the Log

Query	toyota sedan review 2005
Bid Keyword	used toyota camry 2005
Ad Title	2005 Toyota for Sale
Ad Copy	Find a Toyota Near You. Compare 2005 Models Now!
Display url	www.AutoTrader.com/Toyota

candidate ads. Since the selected keywords can directly map to a group of ads, we focus on the selection of keywords in this paper.

Relevance between the issued query and a selected ad, how likely a user will click an ad triggered by a selected keyword, is the main concern of existing works on ad selection. Some approaches rely on the text relevance among several text streams like query, keyword, ad copy, or the landing pages [3][5]. Some employ the graph information from query logs or ad click logs [2][8]. Other works like Hillard *et al.* [9] import both text relevance and graph information into the learning model as features.

Nevertheless, relevance is far from enough for ad selection. As sponsored search is the main source of revenue in the commercial search engines, the monetization ability of the selected keywords should also be taken into consideration in the selection phase. All of the existing methods, however, only focus on the relevance where high relevance doesn't necessarily lead to high monetization ability since revenue is also influenced by the bid prize of the keywords. Besides, different from the relevance, which depends on both queries and keywords, the monetization ability of a keyword is query-independent. The ad selection algorithm should be able to pick out keywords with high relevance given query meanwhile with better monetization ability. Furthermore, our data study in Section 2 indicates that different components inside keywords have different impacts on relevance and monetization ability. Thus, it seems infeasible to take the keyword as a whole in the selection when considering both relevance and monetization ability, which are component-based instead of keyword-based. In other words, to take both relevance and monetization ability into consideration, we shall try to select keywords on a component basis. Accordingly, we propose a novel ad selection method by co-analyzing the relationship among different components inside the user queries and advertiser keywords. Our novel method make it possible to take advantages of the different impacts of different components so that we can optimize the relevance and monetization ability based on components.

In particular, we decompose queries and keywords into entities and modifiers separately. We build a global bipartite graph between query entities and keyword entities; at the same time, we build a local bipartite graph between the corresponding query modifiers and keyword modifiers for every entities. The local bipartite graphs are regarded as the expression for the entities. In the on-line system, when a new query comes, we decompose it into entities and modifiers, and mine the candidate entities and modifiers from the built two-layer graph separately where both the relevance and the monetization ability are considered.

Finally, we generate the combinations of the suggested entities and modifiers and match them in the bid keywords database to make the selection.

We evaluate the proposed methods from two aspects: relevance, which is represented by recall and precision, and the monetization ability. We compare the results with two baseline ad selection methods, namely the classical *Tf-Idf* model and the *Random Walk* algorithm [1]. On all the metrics used, our method significantly outperforms the baselines. We employ the ad click data of the sponsored search log from a commercial search engine in our work. In each entry of the log, there is plenty of information including original query, matched keyword, ad description (ad title, ad copy, display url, landing page url), etc., and Table 1 shows an example of the record, in which a user issued a query *toyota sedan review 2005* and clicked an ad of an automobile trader.

The rest of the paper is organized as the following. We investigate the different impacts of entities and modifiers on relevance and monetization ability in Section 2. In Section 3, we employ these impacts in our novel proposed ad selection method. We present the experimental results in Section 4, and summarize the related work in Section 5. Finally, we conclude the paper in Section 6.

2 Data Study

In this Section, we investigate the impacts of different components, entities and modifiers, inside the keywords on the effectiveness of sponsored search system.

2.1 Extracting Entities and Modifiers from Queries or Keywords

Entity recognition has been well studied in the literatures [4] *etc.*. Since it is not our main concern, we adopt similar methods employed in [12] to identify entities with a pre-defined entity list, which contains over 30 thousands entities, and the list can be updated with many specialized methods. For a query, we remove the stop words and some irregular characters and then identify the entities according to the entity list. The remaining terms of the query are regarded as the modifiers, similar setting with the methods in work [12]. For the example in Table 1, the query contains one entity ("toyota sedan") and two modifiers ("review", "2005"). For the keyword, the recognition methods and entity list are the same with the queries. In the former example, the keyword contains one entity ("toyota camry") and two modifiers ("used", "2005").

2.2 Statistical Test Methods

Relevance between the selected keywords and query and the monetization ability are two of the most important aspects of the effectiveness of the sponsored search system. In this section, we take click-through rate (abbreviated as CTR) and historical revenue (abbreviated as revenue) as metrics for relevance and monetization ability respectively. Here the keyword-level CTR is calculated as the ratio between the counts of ad clicks and the counts of ad impressions within the given period of time, both of which are triggered by the corresponding keyword. The keyword-level revenue is calculated as the sum of the revenue from the ad clicks triggered by the corresponding keyword within the given period of time.

Further, keywords containing same entity (modifier) are clustered in one entity (modifier) groups. If keywords contain more than one entities (modifiers), it will be put into all the corresponding entity (modifier) groups. For example, in Table 1, the keyword *used toyota camry 2005* is included in one entity group *toyota camry* and two modifier groups *used* and *2005*.

The keywords used in this study were uniformly sampled from the sponsored search log of a commercial search engine. There are in total 0.9 million unique keywords covering two months ad click records. For each keyword, we calculated its CTR and revenue within the two months.[1] We extracted 7,400 unique entities and 2300 unique modifiers from the keywords with the methods mentioned in Section 2.1. There are 250 keywords in each entity group on the average, meanwhile 167 keywords in each modifier group.

Our method is to compare the mean value among all these entity (modifier) groups. If the entities (modifiers) have impacts on the tested variable (CTR or revenue), there should be significant differences among the group mean values. Specifically, we first conduct the one-way analysis of variance (ANOVA) [10] on CTR and revenue where the null hypothesis (H_0) is the mean values of all groups are equal. As there are thousands of different entities or modifiers, which means thousands of groups, H_0 is actually a very strong statement. If we cannot reject the H_0, we may have confidence to claim the entities (modifiers) have no impact on the tested variable. Otherwise, however, able to reject H_0 is far from enough to support the exists of the impacts, since the sources and the magnitudes of the differences are not clarified. Therefore, we further conduct Tukey's HSD test [10] to compare each group with every other entity (modifier) groups, and for each entity (modifier) group we count the numbers of groups ($GNum$) with significantly different mean value (one pair is only counted once) comparing with its group mean value. If the $GNum$ of most of the entity (modifier) groups are larger than 1, we may conclude entities (modifiers) have impacts on the tested variable.

2.3 Impacts on Relevance

The ANOVA [10] on CTR for entity groups and modifier groups shows that they both reject the null hypothesis at 0.01 level, indicating that there are significant differences among the 7400 entities and 2300 modifiers on CTR (the ANOVA graph is omitted for space reasons). We further conduct Tukey's HSD test [10].

For entities, the results show that each entity is at least significantly different from other 16 entities meanwhile the average value is 244. As to the modifiers, the minimum number of $GNum$ is 3 and the average is 38. The results indicate that both entities and modifiers have impacts on CTR. The left-most 6 columns

[1] Note that all the revenue related values are multiplied by a specific number due to the confidentiality requirements of the search engine.

Table 2. Top 5 Entities and Modifiers with Best Distinguish Ability on CTR

Entity	*GNum*	CTR	Modifier	*GNum*	CTR	Entity	*GNum*	Revenue
iTunes	7341	1.69	chase	2262	0.50	online college	7339	43085
HSBC	7341	1.62	speck	2251	0.58	state farm	7326	33018
green dot	7341	1.79	download	1728	0.35	flower delivery	7324	30910
P&G	7340	1.55	login	615	0.28	auto insurance	7323	26720
Citibank	7339	1.17	pay	477	0.25	home secure	7317	27187

of Table 2 show the top 5 entities (modifiers) that can distinguish most of other entities (modifiers) on CTR. In this part, we confirm both the entities and the modifiers have impacts on relevance of the bid keywords.

2.4 Impacts on Monetization Ability

The ANOVA [10] on revenue for entity groups rejects the null hypothesis at 0.01 level, indicating that there are significant differences among the 7400 entity groups on revenue. Meanwhile, the test on modifiers fails to reject the null hypothesis. Due to the null hypothesis is a strict statement which supposes all of the 2,300 modifier groups have the same mean revenue, we have strong confidence to claim that the modifiers do not have impacts on monetization ability (the ANOVA graph is omitted for space reasons).

We conduct Tukey's HSD test [10] on entity groups. The tests on revenue show that for each entity the minimum value of $GNum$ is 6 and the average number is 55. The results indicate that entities have impacts on monetization ability of keywords. The right-most three columns of Table 2 show the top 5 entities with maximum $GNum$. In this part, we confirm the entities have impacts on monetization ability of keywords while modifiers not.

2.5 Highlight from Data Study

To sum up, we can conclude that both entities and modifiers have impacts on sponsored search effectiveness. Data study indicates entities play an important role on both relevance and monetization ability, meanwhile the modifiers only have impacts on relevance. In our proposed system, inspired by the different impacts, we mine entities and modifiers separately, where optimization on both relevance and monetization ability are made, and then we make combinations of these two parts to pick out keywords.

3 Entity Relationship Mining for Ad Selection

In this section, we describe the proposed entity relationship mining method for ad selection. We first introduce the off-line system of building the knowledge base, and then we explain the on-line procedures of applying the knowledge base to generate the selected keywords (Figure 1).

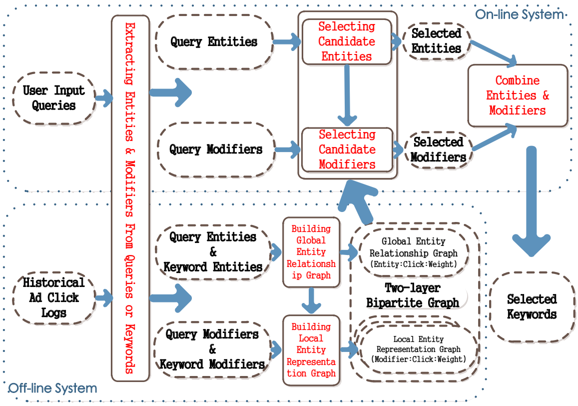

Fig. 1. Entity relationship mining system for ad selection. There are on-line system and off-line system. The dotted circles in this graph represent various kinds of data streams, and the solid rectangles represent important procedures.

3.1 Off-Line System: Building Knowledge Base for Entity Relationship

Building *Global Entity Relationship Graph.* After decomposing the queries and keywords in all records in the log according to the methods in the Section 2.1, we build a weighted bipartite graph between query entities and keyword entities. The nodes on the two sides of the graph are respectively the query entities and keyword entities extracted from the historical ad click logs. When there are clicked ads associated with a query and a keyword, there should be an edge between them, meanwhile the weight of the edge will equal the total number of ad clicks. In the example in Table 1, if there is no existing edge, we would build an edge between the query entity node "toyota sedan" and the keyword entity node "toyota camry" on which the weight is set one, otherwise, the weight of the edge would equal the total number of clicks. If the query or keyword contains more than one entities, we will create edges between every query entities and every keyword entities. We call this graph the *global entity relationship graph* due to it encodes the historical ad click information among all entities appeared.

Building *Local Entity Representation Graph.* We notice that the correlations among the modifiers can provide information of characteristics of the entities. In the aforementioned example, we may regard that the correlation between "2005" and "used" is strengthened by the ad click. In another words, it is verified by the user ad click that "used" is related to the past year "2005", and this knowledge is associated with the entities "toyota sedan" and "toyota camry". This kind of knowledge is entity-specified and can be used to enrich the entity relationships. In our method, we generate the modifier pairs from each click record and attach the pairs to every corresponding entities in the record.

In the example, we have four pairs across the query modifiers and the keyword modifiers, i.e., "review – used", "review – 2005", "2005 – used", and "2005 – 2005". These pairs are attached to both the query entity "toyota sedan" and the keyword entity "toyota camry". For each entity in the *global entity relationship graph*, it might be attached with such pairs from multiple records. We collect these pairs and build a bipartite graph between the query modifiers and keyword modifiers for each entity. We call this graph as the *local entity representation graph* as it describes each entity with a specific modifier graph generated from the ad clicks, which might also be regarded as the entity expression in the modifier space. These graphs contain the information about the characteristics of each entity and are used to catch the similarities among the entities.

3.2 On-Line System: Making Combination of Selected Entities and Modifiers

When a new query is submitted to the sponsored search system, we split it into query entities and query modifiers accordingly (Section 2.1). After that, we use the off-line built graphs to mine suitable entities and modifiers of keywords separately and then we combine these components to generate selected keywords.

Selecting Candidate Entities. When mining the keyword entities, there are two steps. First, we run a random walk with restart algorithm [11] on the *global entity relationship graph*. In the initial entity node distribution, the elements corresponding to the extracted query entities are set to non-zero and equal values, and the rest elements are set to zeros. After several steps of graph propagation, we check the element values of the keyword entities and select the top ranked entities (as many as two times of the numbers of entities need in the combination). These selected keyword entities are preliminary candidates, which are all to some extent related to the query entities, for our entity selection phase.

After that, we take both relevance and the monetization ability into consideration in identifying the final keyword entity candidates with the following formulae. Here we take advantages of relationship between *local entity representation graphs* to compute relevance among entities. Comparing with the element values from the random walk in the global graph, local graph, where entities have similar characteristics share similar modifiers, can describe the similarity between query entities and keyword entities more precisely (and perform better in our small-scale preliminary tests). The historical revenue of entities is used as metrics for monetization ability. The historical revenue of entity (modifier) is the average of revenue of keywords within the entity (modifier) groups mentioned in Section 2.2. Then we rank the entities according to the multiplication of the entity similarity and monetization ability, which can be regarded as expectation of the entity's revenue in this search.

$$Score(E_q, E_x, HR_x) = SimScore(E_q, E_x) * HR_x \tag{1}$$

$$SimScore(E_q, E_x) = \frac{|LS(E_q) \cap LS(E_x)|}{|LS(E_q) \cup LS(E_x)|} \tag{2}$$

In the above equations, E_q is the entity from the query, E_x is the keyword entity to be judged, and HR_x is the historical revenue of E_x. $SimScore(E_q, E_x)$ is the entity similarity score. We employ the normalized overlap ratio computed in (2) between the two graphs as the similarity score. In this formula, $LS(A)$ means the respective set of the *local entity representation graph* of A. The entity similarity score is in the interval $[0, 1]$ and it indicates the probability of E_q and E_x being similar.

Therefore, according to the combined scores in (1), we select the top N_e entities as the final keyword entity candidates.[2]

Selecting Candidate Modifiers. To mine the similar keyword modifiers, we merge the corresponding *local entity representation graphs* of the selected keyword entities and get an aggregated local entity representation graph. In particular, just like the methods used in building the local graph, the nodes in the new graph are the modifiers and the edges come from the graphs being merged, and weights of the edges are the sum of the weights of all edges linking the same modifier pairs. We then run the random walk with restart algorithm [11] on the graphs. After several steps of graph propagation, we check the element values of the keyword modifiers and select the top N_m modifiers as candidates.[3]

Combine the Entities and Modifiers. After we get the keyword entities and modifiers, we can select the matched keywords in a very efficient way. We first obtain keywords by generating all possible entity-modifier combinations, which have at least one bid in the log, from the selected entities and modifiers. After that, we sum up the entity score computed in (1) and the random walk score computed in selecting modifiers as the final score for each combination. We rank all the candidate combinations and select the top N_k.[4] Afterwards, as we described in Section 1, the ads associated with all these selected keywords are chosen as the candidate ads for the follow-up ad ranking algorithm [7].

In all above on-line phases, we desire to select enough candidate keywords whose number (N_k) are actually determined by the sponsored search system. To do so, we set the number of entities (N_e) and modifiers (N_m) by empirical experiments where the setting of these parameters have little influence on the performances and the current settings can guarantee we have enough keyword candidates to be selected with. Due to the space reasons, we omit the discussion about the selects of parameters in our work.

4 Experimental Evaluation

In this section, we evaluate our proposed method by comparing with two baseline ad selection algorithms on both relevance and monetization ability. We test two

[2] In our experiments, we set $N_e = 10$ by experiences.

[3] In our experiments, we set $N_m = 10$ by experiences.

[4] In our experiments, $N_k = 30$ according to the set of search engine.

settings of the proposed method: *OnlyEntity* (abbreviated as *OE*) and *EntityWithModifier* (abbreviated as *EWM*). *OE* employs the entity expansion results and matches the keywords with only entity parts. *EWM* further takes advantages of the modifiers and matches keywords with entity-modifier combinations. The experimental results show that the proposed *EntityWithModifier* method can significantly outperform the baselines on recall rate, precision rate, and monetization ability.

4.1 Dataset and Baselines

The historical ad click data building the off-line knowledge base is sampled from the sponsored search log of a commercial search engine within two months (different from data used in Section 2 where only keywords are extracted). It contains 3.5 million query-keyword pairs which are associated with ad clicks. There are in total 1.5 million unique queries and 0.51 million unique keywords. The evaluation data comes from the same log whose time frame is three days after the aforementioned two months. It contains 0.4 million records with 22.5 thousand unique queries and 12 thousand unique keywords. We decompose queries and keywords according to methods mentioned in Section 2.1. and can get at least one entity from 97% queries and keywords.

We implemented two baselines in our experiments: one is based on the classical text relevance method *Tf-Idf* with query expansion, and the other is the *Random Walk* algorithm on the historical click graph. In the *Tf-Idf* baseline, we use the *Tf-Idf* framework in which the queries are expanded with the top 10 snippets from the organic search results. The *Random Walk* baseline employs a similar method discussed in Antonellis *et al.* [1]. According to our knowledge, these two baselines are the foundations for all current ad selection algorithms.

4.2 Evaluation on Relevance: The Recall Rate

According to the setting of the commercial search engine, given a query, the ad selection algorithms will return at most top 30 ranked keywords. The goals for ad selection are to select "correct" keywords within a small set size (e.g., 30 keywords in total) which is quite different from information retrieval tasks where the rank of the results is the most important. In this section, we judge the recall rate at each position by comparing the results with the actual keywords with triggered ad clicks. In particular, for each query, if the top N returned keywords contain all of keywords triggering ad clicks in the log, the recall rate at N is 100%; otherwise the value is 0%. We compute the average recall rate at each position on the 22.5 thousand queries for test.

The statistical test shows that the recall rate of *OE* and *EWM* are both significantly higher than those of the two baselines on top 30 keywords at 0.01 level. Furthermore, at each position, *EWM* outperforms the *OE* method by more than 10% units at all positions. It double confirms the conclusion in Section 2 that both entities and modifiers have impacts on relevance.

Table 3. Recall in Different Positions

Position	*OE*	*EWM*	*Tf-Idf*	*Random Walk*
10	48.44%	**59.71%**	57.44%	58.94%
15	53.79%	**65.13%**	62.60%	59.71%
20	57.11%	**68.86%**	66.33%	60.14%
25	60.13%	**72.11%**	69.01%	60.49%
30	62.25%	**74.24%**	71.60%	60.68%

Table 4. Precision on Manually Labeled Results

Label	*OE*	*EWM*	*Tf-Idf*	*Random Walk*
Relevant (3-5)	76.87%	**79.50%**	71.11%	78.59%
Irrelevant (2)	23.13%	20.50%	28.89%	21.41%

4.3 Evaluation on Relevance: The Manually Judged Precision

In this section, we employ manual judgement to evaluate the precision of the query-keyword pairs generated by the ad selection algorithms. In particular, given the same query set, we take top 20 returned keywords for each query with one of four algorithms, *OE*, *EWM*, *Tf-Idf*, and *Random Walk*, for manual judgement. Then we pool all the query-keyword pairs and each query-keyword pair is judged by an evaluator. The evaluators give a score for each query-keyword pair from 1 to 5 which stand for cannot judge, irrelevant, weak relevant, relevant, and strong relevant respectively. It's blind for the evaluators that which algorithms generate the specific given query-keyword pair and the judgment scores are solely based on the evaluators' knowledge. As the labeling task is expensive, in total there are 1,600 query-keyword pairs, including 600 overlap pairs among the four algorithms, are labeled. Since the numbers of query-keyword pairs being labeled as "Cannot Judge" are quite small (around 10) in all these four methods, we remove pairs with the label 1 in our further analysis.

We regard the label 2 as irrelevant, and merge the labels 3, 4, 5 as relevant to compute the precision scores. The results in Table 4 show that *EWM* can outperform *Tf-Idf* and *Random Walk* by 8.4% and 0.9% respectively at a 0.05 significance level. Comparing the *OE* and *EWM* methods, we can see that the combination of entities and modifiers improves the precision by considering the modifiers.

4.4 Evaluation on Monetization Ability

In this section, we employ a simulation system of sponsored search to evaluate the monetization ability of the four algorithms in the experiments. In the simulation system, given a keyword group from ad selection, the system can conduct the off-line simulated auctions and get the collection of winner ads to be displayed on the search result pages. As we do not know which ads the users may click, we use the sum of the cost per click (abbreviated as CPC, the amount of money the search engine will get when the ad is clicked [6][7]) of the top n returned ads

Table 5. Simulate Results on Revenue

Position	*OE*	*EWM*	*Tf-Idf*	*Random Walk*
1	230.76	**267.17**	255.03	257.23
5	207.69	**243.32**	237.03	228.24
10	190.23	**225.90**	219.03	204.87
15	178.15	**213.66**	202.93	188.06
20	175.43	**204.06**	193.45	175.43
25	168.94	**196.24**	185.87	165.34
30	161.46	**189.62**	179.56	156.93

triggered by each selected keyword as the metric for the monetization ability of the selected keywords.

Therefore, for each query and each algorithm, we compute the average values (among all tested keywords) of sum CPC of the returned top n ads and we set n as 1, 5, 10, 15, 20, 25, 30 here. From the results in Table 5, we can find that *EWM* outperforms all the other algorithms by about 5% units at all positions. With statistical tests, we confirm that all of the differences are significant at 0.05 level. Here, the revenues of *OE* are significantly less than *EWM*. That is because, though *OE* may select keywords with good monetization ability, some of its returned top-rank ads are less relevant with the users' queries and fail to attract the users' clicks.

5 Related Work

Existing works on ad selection mainly try to improve the relevance of the selected keywords (or ads) given query. Some of them are relevance-based. Broder *et al.* [3] enriched both queries and ads with additional knowledge features. Broder *et al.* [2] proposed alternative approach of matching the ads against rare queries, and make the process be able to accomplished on-line. Choi *et al.* [5] explored the usage of the landing page content to expand the text stream of ads. Some other methods employ the historical click information to mine the relationship among queries and keywords (ads). Fuxman *et al.* [8] conducted the keyword suggestion by making use of the query logs of the search engine. In the work of Hillard *et al.* [9], the author introduced a machine learning approach based on translation models to predict the ad relevance.

In comparison with these works, our paper takes both relevance and monetization ability into consideration in the selection. Besides, in the above works, the text streams employed are based on queries or keywords as a whole and our work, on the other hand, tries to conduct ad selection based on components inside queries and keywords.

6 Conclusions and Future Work

In our work, we proposed a novel ad selection methodology in which both relevance and monetization ability of keywords are considered. In particular, we make ad selection by picking out, and then combining, the keyword components. Experimental results show that the proposed method outperforms two baseline ad selection algorithms on both relevance and monetization ability. For the future work, we would like to take the interests of advertisers, like conversion rate, into consideration in our ad selection algorithm.

References

1. Antonellis, I., Molina, H.G., Chang, C.C.: Simrank++: query rewriting through link analysis of the click graph. Proc. VLDB Endow. 1(1), 408–421 (2008)
2. Broder, A., Ciccolo, P., Gabrilovich, E., Josifovski, V., Metzler, D., Riedel, L., Yuan, J.: Online expansion of rare queries for sponsored search. In: Proceedings of the 18th International Conference on World wide Web, WWW 2009, pp. 511–520 (2009)
3. Broder, A.Z., Ciccolo, P., Fontoura, M., Gabrilovich, E., Josifovski, V., Riedel, L.: Search advertising using web relevance feedback. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM 2008, pp. 1013–1022 (2008)
4. Chandel, A., Nagesh, P.C., Sarawagi, S.: Efficient batch top-k search for dictionary-based entity recognition. In: Proceedings of the 22nd International Conference on Data Engineering, ICDE 2006, pp. 28–37 (2006)
5. Choi, Y., Fontoura, M., Gabrilovich, E., Josifovski, V., Mediano, M., Pang, B.: Using landing pages for sponsored search ad selection. In: Proceedings of the 19th International Conference on World Wide Web, WWW 2010, pp. 251–260 (2010)
6. Edelman, B., Ostrovsky, M., Schwarz, M.: Internet advertising and the generalized second price auction: Selling billions of dollars worth of keywords. Working Paper 11765, National Bureau of Economic Research (November 2005)
7. Feng, J., Bhargava, H.K., Pennock, D.M.: Implementing sponsored search in web search engines: Computational evaluation of alternative mechanisms. INFORMS J. on Computing 19(1), 137–148 (2007)
8. Fuxman, A., Tsaparas, P., Achan, K., Agrawal, R.: Using the wisdom of the crowds for keyword generation. In: Proceedings of the 17th International Conference on World Wide Web, WWW 2008, pp. 61–70 (2008)
9. Hillard, D., Schroedl, S., Manavoglu, E., Raghavan, H., Leggetter, C.: Improving ad relevance in sponsored search. In: Proceedings of the Third ACM International Conference on Web Search and Data Mining, WSDM 2010, pp. 361–370 (2010)
10. Schervish, M.J.: Theory of Statistics. Springer (1995)
11. Tong, H., Faloutsos, C., Pan, J.-Y.: Fast random walk with restart and its applications. In: Proceedings of the Sixth International Conference on Data Mining, ICDM 2006, pp. 613–622 (2006)
12. Yin, X., Shah, S.: Building taxonomy of web search intents for name entity queries. In: Proceedings of the 19th International Conference on World wide Web, WWW 2010, pp. 1001–1010 (2010)