Using Landing Pages for Sponsored Search Ad Selection

Yejin Choi†, Marcus Fontoura‡, Evgeniy Gabrilovich‡,
Vanja Josifovski‡, Mauricio Mediano‡, and Bo Pang‡

† Department of Computer Science, Cornell University, 4162 Upson Hall, Ithaca, NY 14853
‡ Yahoo! Research, 4301 Great America Parkway, Santa Clara, CA 95054
ychoi@cs.cornell.edu | {marcusf | gabr | vanjaj | mmediano | bopang}@yahoo-inc.com

ABSTRACT

We explore the use of the landing page content in sponsored search ad selection. Specifically, we compare the use of the ad's intrinsic content to augmenting the ad with the whole, or parts, of the landing page. We explore two types of extractive summarization techniques to select useful regions from the landing pages: *out-of-context* and *in-context* methods. Out-of-context methods select salient regions from the landing page by analyzing the content alone, without taking into account the ad associated with the landing page. In-context methods use the ad context (including its title, creative, and bid phrases) to help identify regions of the landing page that should be used by the ad selection engine. In addition, we introduce a simple yet effective unsupervised algorithm to enrich the ad context to further improve the ad selection. Experimental evaluation confirms that the use of landing pages can significantly improve the quality of ad selection. We also find that our extractive summarization techniques reduce the size of landing pages substantially, while retaining or even improving the performance of ad retrieval over the method that utilize the entire landing page.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Retrieval Models and Selection Process

General Terms

Algorithms, Experimentation

Keywords

Sponsored search, landing pages, extractive summarization, compositional semantics

1. INTRODUCTION

Web search is the gateway to the Internet for billions of users daily. When the user issues a query to the search engine, two separate searches are evaluated: the search over the corpus of pre-crawled web pages is called *web search*; the advertisements that are displayed at the top and the side of the web search results are retrieved by *sponsored search*. Sponsored search provides revenue for the search engine and brings users to numerous advertiser sites.

WWW 2010, April 26–30, 2010, Raleigh, North Carolina, USA.
ACM 978-1-60558-799-8/10/04.

Web search and sponsored search differ in a few key aspects. Sponsored search is evaluated over a set of ads that promote products and services. As it is customary in the advertising world, the textual content visible to the user (*ad creative*), is generated by the advertiser to maximize the response of the target audience. In web search on the other hand, the snippet shown on the search result page is generated automatically by the summarization mechanism of the search engine. Another important difference is in the way the ads and the web results are selected. While the web pages are selected based on their content, the ad selection depends heavily on the use of the ad *bid phrase* – a query that the advertiser has specified as suitable for the ad. In the early days of the sponsored search marketplace, this mechanism allowed for simple ad selection where the whole burden (and control) is shifted to the advertiser. However, with the development of the sponsored search market, it become quickly apparent that the advertisers cannot possibly find all the queries that could be relevant to their advertisements.

To alleviate this problem the search engines allow for *advanced match* where ad can be selected even if their bid phrase does not match the query. The advanced match problem corresponds closer to the web search problem. Recent advanced match approaches use search techniques for ad selection by evaluating the query over a corpus of documents that are created from the ads [7, 25]. One of the key difficulties in this *ad retrieval* approach is that the ads are much shorter than documents in most other search applications [25].

In this paper, we explore the use of the landing page in ad retrieval for sponsored search. We contrast the use of the content of the ad creative with the use of the whole, or parts, of the landing page. Our study was partly motivated by a preliminary examination of a set of textual ads and their landing pages, which indicated that over 30% of the landing pages are not at all, or very remotely, related to the ads. Thus our intuition suggests that indiscriminate use the content of such landing pages in the ad selection would decrease the precision of the ad retrieval.

We explore two types of extractive summarization techniques to select useful regions from the landing pages: *out-of-context* and *in-context methods*. Out-of-context methods select regions from the landing page by simply analyzing the landing page itself, without taking the *ad context* into account. The ad context is composed of the creative, bid phrase, title and any other information about the ad that can be computed offline (i.e., prior to query processing). In-

context methods use the ad context to help select which regions of the landing page should be used by the ad selection engine. In addition, we introduce a simple yet effective unsupervised algorithm motivated by compositional vector space models [21, 22, 14] based on compositional semantics [23, 21] in order to enrich the ad context and enhance the ad selection. Experimental results demonstrate that selective use of landing pages can significantly improve the quality of ad selection. We also find that our extractive summarization techniques reduce the size of landing pages substantially, thereby reducing the amount of data that needs to be indexed, while retaining or even improving the performance of ad retrieval over the method that utilize the entire landing page.

The contributions of this paper are threefold:

- We quantify the benefit of using the landing page for ad selection in sponsored search. In particular, we examine a number of different extractive summarization techniques to make the best use of landing pages.
- We propose a simple yet effective unsupervised algorithm using compositional vector space models to enrich the ad context. We then present two different ways in which the enriched ad context can be utilized to enhance the ad selection.
- We report experimental results that show that our proposed methods for selecting landing page regions have up to 8.5% performance improvement in Discounted Cumulative Gain (DCG), measured over a production-level ad selection system.

2. BACKGROUND

A large part of the $30 billion Web advertising market consists of *textual ads*, the ubiquitous short text messages usually marked as "sponsored links". There are two main channels for distributing such ads. *Sponsored search* places ads on the result pages of a Web search engine, where ads are selected to be relevant to the search query (see [11] for a brief history of the subject). All major Web search engines (Google, Microsoft, Yahoo!) support sponsored ads and act simultaneously as a Web search engine and an ad search engine. *Content match* (or *contextual advertising*) places ads on third-party Web pages. Today, almost all of the for-profit non-transactional Web sites (without direct sales) rely at least to some extent on contextual advertising revenue. Content match supports sites that range from individual bloggers and small niche communities, to large publishers such as major newspapers.

In this paper we focus on sponsored search. However, content match ads are identical to the sponsored search ads and we believe that using landing page content for ad selection would be applicable to content match as well.

Sponsored search is an interplay of three entities: The **advertiser** provides the supply of ads. As in traditional advertising, the goal of the advertisers can be broadly defined as promotion of products or services. The **search engine** provides "real estate" for placing ads (i.e., allocates space on search results pages), and selects ads that are relevant to the user's query. **Users** visit the Web pages and interact with the ads.

The prevalent pricing model for textual ads is that the advertisers pay for every click on the advertisement (pay-per-click or PPC). The amount paid by the advertiser for each sponsored search click is usually determined by an auction process [9]. The advertisers place *bids* on a search phrase, and their position in the column of ads displayed on the search results page is determined by their bid. Thus, each ad is annotated with one or more *bid phrases*. In addition to the bid phrase, an ad also contains a *title* usually displayed in bold font, and a *creative*, which is the few lines of text, usually shorter than 120 characters, displayed on the page. Naturally, each ad contains a URL to the advertised Web page, called the *landing page*. A recent study, analyzed the types of landing pages [3] and we classified the landing pages into three main categories: *homepage*, which are top-level pages on the advertisers' Web site; *search transfer*, which are dynamically generated search result pages on the advertiser's site; and *category browse*, which are subsections of the advertiser's site, generally related to the user query.

In this work we explore the use of landing pages in the context of an ad retrieval system that is based on information retrieval principles, as reported in [7]. The input to our system is a search (or "Web") query, and the output is a set of ads that are relevant to this query. We represent the ads and the queries in a vector space model using their unigrams and phrases as features. The query-ads similarity is computed using cosine between the angle between their vectors. Assuming that the query vector and the ad vectors are pre-normalized using L_2 norm, the scoring function is a simple dot product:

$$score(query, ad) = \sum_{f \in ad \cap query} w_{f,ad} w_{f,query}$$

where $w_{f,ad}$ and $w_{f,query}$ are the weights of the feature f in the ad and the query accordingly. For weighting of the features we use a section-aware variant of tf-idf where each region of the ad is given a tf multiplier. This weighting scheme can be naturally extended to incorporate new regions of the ad and the query, as the one we explore in this paper – the landing page. The ads are processed and indexed in an inverted index of ads that at runtime is used to evaluate similarity queries by a document-at-the-time algorithm. For more details we refer the reader to [7].

3. SUMMARIZING LANDING PAGES FOR AD RETRIEVAL

We explore a number of different ways to extract information from landing pages that can be used to augment ad indexing and eventually help with ad selection. Since the extracted information should be a succinct representation of the most useful information in the landing page, it can be viewed as a "summary" of the given page. A fundamental question we face here is what constitutes a good summary for our ad retrieval task?

We start by noting that a landing page can contain many different *regions*, each focusing on one type of information about the subject matter of the page. For instance, in the example shown in Figure 1, the page is mainly about "Canon EOS Digital Camera". It contains several regions that are directly related to the product: a product-description region, a customer-review region, and a transaction region with information for purchase. It also contains a region with a list of related products — information that is tangentially related to the product being advertised. In addition, there

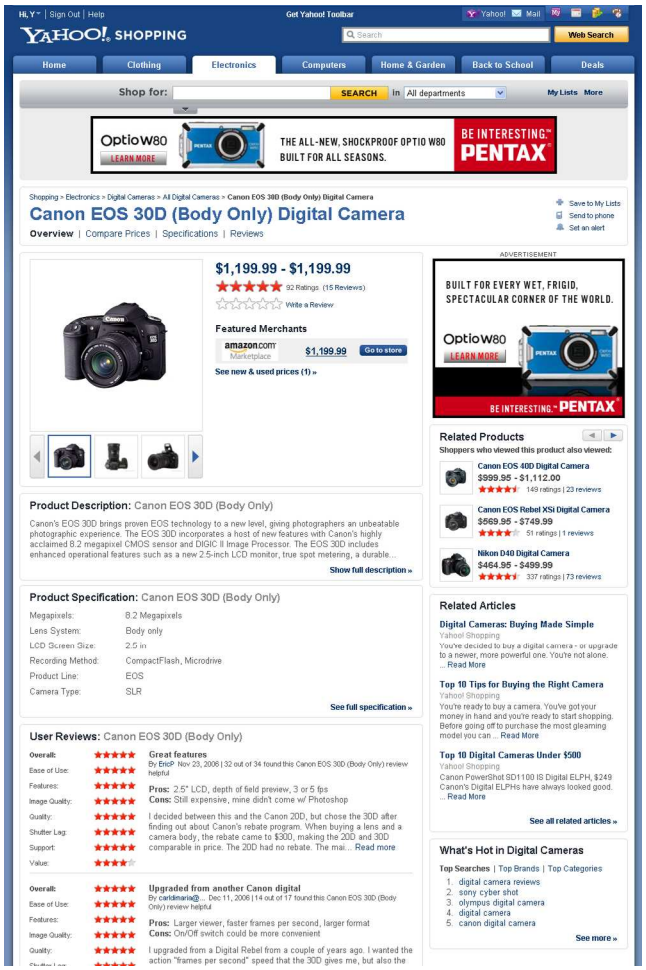

Figure 1: An example of a landing page.

are regions with navigational information that carry no information about the product at all. Clearly, not all of these regions are equally important for ad retrieval. Our goal is to investigate what is the best way to select information from the right regions that could help improve ad retrieval.

In this paper, we explore two different hypotheses. The first hypothesis is that a good summary should be defined in the context of the advertisement intent. For instance, if the intent of a given ad is to provide customer reviews, then a good summary should likewise focus on the customer reviews in the page rather than, say, the transaction information. In contrast, the second hypothesis assumes that a good summary for a landing page can be defined solely based on information available in the landing page itself, without referring to the content of the ad creative for the advertisement intent.

In what follows, *in-context term selection* refers to summarization of landing pages based on the first hypothesis, and *out-of-context term selection* refers to approaches based on the second hypothesis. Note that our end goal is not to produce a human readable summary of a web page. Rather, we plan to extract those terms from the landing pages that can assist advertisement selection.

TITLE	Machine Learning
BID PHRASE	machine learning
SHORT DESCRIPTION	Compare Prices in 101+ stores. Find cheap book prices every time.

Figure 2: An example of an ad

a := TF-logIDF representation of an ad
CR := {[-5,+5] landing page text around any word in a}
$RR := \emptyset$
For each candidate region $r_i \in CR$,
  If $cosine\_similarity(a, r_i) > \delta$,
  Then $RR \leftarrow RR \cup r_i$,
Return RR as relevant regions for the given ad

Figure 3: Extracting relevant regions

3.1 *In*-Context Term Selection

We investigate the first hypothesis by introducing two variants of an algorithm that select *relevant regions* in the landing page with in the context of the ad intent. Both use "seed words" representing the ad intent to help select relevant regions. In the first (and simpler) variant, we use only words from the ad content to represent the ad intent; in the second variant, we use an extended set of words. We start by describing the simpler variant.

3.1.1 *Extracting Relevant Regions*

Figure 3 shows the procedure to select terms from a landing page by extracting the relevant regions based on the content of the ad.

We use the tf-idf model to represent ads. The textual information we utilize in an advertisement consists of three components: a title, a short description, and a bid phrase. As we can see in Figure 2, terms that are repeated across different components (e.g., "machine" and "learning") tend to be more important than words that are repeated inside only one component (e.g., "prices"). Thus, for the computation of term frequency, we count the number of components that term appears in. Similarly, we treat each ad component as a separate document in an ad corpus for the computation of document frequency. We refer to the resulting vector as *ad vector*.

Next, we locate candidate regions in a given landing page in the context of the ad vector. For any word in the landing page that also appears in the ad vector, we consider the text span in [-5, +5] window as a candidate region. For each candidate region, we compute the cosine similarity between the candidate region and the ad vector. We merge all candidate regions whose similarity scores are above a certain threshold δ. The resulting regions are relevant regions for a given advertisement.

One natural concern regarding this approach is that some good regions might not be selected as relevant due to the vocabulary mismatch between the ad and the corresponding landing page, because textual information given in an ad has to be very succinct.

In order to address such concern, we next introduce an algorithm that extends the ad vector into a richer context. We start by building co-occurrence vectors of words appearing

Central word	Co-occurring words(PMI)
mattress	futon(6.4), king(2.95), pillow(4.92) queen(5.64), shopping(2.2), brand(2.5) tempur-pedic(6.66), bunk(5.28), mite(5.79) serta(7.64), sealy(7.79), visco(7.75) platform(4.74), products(1.94), store(2.44) cover(4.1), outlet(3.46), directory(2.4) savings(1.37), topper(5.71), allergen(6.63)

Figure 4: An example of a co-occurrence vector

in an ad corpus (Section 3.1.2). Using the co-occurrence vectors for all words in a given ad, we then *compose* a *semantic vector* that represent the collective semantic meaning of the advertisement intent (Section 3.1.3). Finally, the resulting semantic vector, in conjunction with the original ad vector, is used to assist extracting relevant regions from the landing page (Section 3.1.4).

3.1.2 Building Co-occurrence Vectors from Ad Corpus

In order to overcome the vocabulary mismatch problem, we built co-occurrence vectors from an advertisement corpus that contained over half million ads. Again, each of the three textual ad components (title, short description, and bid phrase) was treated as a separate pseudo-document d. We define the co-occurrence count for a pair of words u and w as the number of pseudo-documents they co-appeared in:

$$cooc_{cnt}(u, w) = |\{d \mid u \in d \wedge w \in d\}|$$

We discarded stop-words and infrequent words (those that appeared in the corpus fewer than 4 times). We then formed the co-occurrence vector for each word u as

$$cooc_{vec}(u) = \{w \mid cooc_{cnt}(u, w) > 0\}$$

We kept only those with $|cooc_{vec}(u)| \geq 3$.

For all $w \in cooc_{vec}(u)$ we computed its point-wise mutual information (PMI) to u. The definition of PMI is given as follows:

$$\text{PMI}(u, w) = log \frac{\frac{c_{uw}}{N}}{\frac{\sum_{i=1}^{n} c_{iw}}{N} \times \frac{\sum_{j=1}^{n} c_{uj}}{N}}$$

where c_{uw} is the number of times u and w co-occured, n is the number of unique words, and N is the total word occurrences.

As an example, the co-occurrence vector for u = "mattress" is given in Figure 4. The PMI scores were shown in the parentheses. PMI scores reflect how informative a co-occurring word is for u. That is, those with higher PMI scores (e.g., "futon", "tempur-pedic", "serta") are in general more informative than words with lower PMI scores (e.g., "shopping", "products", "savings").

For each u, we also computed the average PMI score for it as

$$avg_{PMI}(u) = \frac{\sum_{w \in cooc_{vec}(u)} \text{PMI}(u, w)}{|cooc_{vec}(u)|}$$

$avg_{PMI}(u)$ represented how "specific" u was. That is, if u co-occurred with many words with low PMI scores, then u was likely to have appeared in many different contexts and domains. In other words, they tended to act like stop words in the advertisement corpus. We added the 50 words with the lowest average PMI scores to our existing stop-word list. Examples of such words included "find", "search", "save", "free", etc. We then rebuilt the co-occurrence vectors using the extended stop-word list.

3.1.3 Computing Compositional Semantic Vectors to Enrich Ad Context

Having constructed co-occurrence vectors for each word u in a given ad, the next question was how to combine them into one vector that captured the ad intent. Let $\{u_i\}$ be the bag-of-words representation of an ad, and $V = \{v_1, ..., v_n\}$ be the set of PMI based co-occurrence vectors for this ad, such that $v_i = \{v_{ij} | j \in cooc_{vec}(u_i)\}$ and v_{ij} is set to the PMI value between the ad word u_i and j. We then investigated different ways to *compose* these vectors into one single vector, which we refer to as the *compositional semantic vector* (csv) for the given ad:

$$csv = f(v_1, ..., v_n) \tag{1}$$

The need for vector composition arises often in information retrieval (IR) and national language processing (NLP). However, it has rarely been the main focus of research until recently. As such, the choice of composition function has been rather arbitrary. The popular choices have been component-wise vector averaging or component-wise vector addition (e.g., [12, 19]). Mitchell and Lapata [21] specifically addressed this issue by viewing the vector composition in light of compositional semantics [23] where the meaning of the whole is a function of the meaning of its parts. This principle of compositionality [13] has been a fundamental presupposition in some of the branches of mathematics, linguistics and philosophy. Researchers extended this insight by comparing various compositional operations in broader NLP applications (e.g., [22, 14]).

In this paper, we investigate this problem in the context of ad retrieval. In particular, we explore different vector compositions in order to compose a semantic vector representation for a given ad. One that has been used often is component-wise vector addition:

$$csv_j = \sum_i v_{ij} \tag{2}$$

where, csv_j and v_{ij} are jth components of vector csv and v_i respectively. Another compositional vector operation can be component-wise vector multiplication as shown below.

$$csv_j = \prod_i v_{ij} \tag{3}$$

Mitchell and Lapata [21] argues that a component-wise vector multiplication is an operation that has been rarely used, but it is conceptually more desirable for meaning composition because multiplication picks out the content relevant to the combination by scaling each component more explicitly.

As noted in [21], it might be desirable that each word should contribute differently to the overall meaning. This is particularly the case in our task. For instance, in Figure 2, the words "Compare" and "Find" are not as informative as the word "book" when distinguishing the given ad from others. In fact, words such as "Compare" or "Find"

might appear in almost all advertisement, regardless of the type of object being advertised. Therefore, such uninformative words should make relatively smaller contribution when composing the semantic meaning of the overall advertisement. To address this issue, we weigh the contribution of each co-occurrence vector by its average PMI scores (as defined in Section 3.1.2). Equation 2 and 3 are then modified as follows:

$$csv_j = \sum_i avg_{PMI}(u_i)v_{ij} \quad (4)$$

$$csv_j = \prod_i avg_{PMI}(u_i)v_{ij} \quad (5)$$

Note that the resulting vector *csv* of Equation 5 is equivalent to that of Equation 3 modulo normalization.

It is worth noting that the weighting scheme used by [21] is different from the one shown in this paper; in [21], weights are defined based on the syntax and semantic role of each word in a given sentence. However, such weighting scheme is not suitable for advertisement retrieval for two reasons. First, languages used in advertisement are succinct and often are not complete or valid sentences. Therefore, it can be hard to determine the semantic role of each word reliably in an advertisement. Second, we often need to weight words in the same syntactic category differently. For instance, in Figure 2, both "prices" and "book" are nouns, and used as objects of verbs. However, the word "prices" is not as informative as "book".

One aspect of composition that previous work (e.g., [21, 14]) did not discuss is the effect of zeros in the multiplications. This is less of a problem if composing only two vectors, as was the case in [21, 14]. However, when composing more than two vectors, if a word did not appear in all of the vectors, its value in the *csv* is zero. To address this problem, we adopt a simple smoothing scheme:

$$csv_j = \prod_i^{smoo} avg_{PMI}(u_i)v_{ij} \quad (6)$$

where the operation $\prod_i^{smoo}$ replaces v_{ij} with a smoothing factor μ whenever $v_{ij} = 0$.

There are other compositional operations that have been explored in literature. For simplicity, consider two vectors v_1 and v_2, where the length of each vector is given as m_1 and m_2 respectively. One example is a tensor product [26], where the resulting vector is a matrix U with dimensionality $m_1 * m_2$, and the component $U_{i,j}$ of the matrix is given as $U_{i,j} = v_{1i} * v_{1j}$. Tensor product is not practically useful for advertisement retrieval, as the dimensionality of the composed vector explodes exponentially. Another compositional operation is circular convolution [27], where the resulting vector u is given as $u_i = \sum_j v_{1j}v_{2i-j}$. In this case, the dimensionality of the resulting vector is manageable, but the computational cost is much heavier than component-wise operations such as Equation 2–6. Also, [14] reports that the performance of convolution is not better than other simpler alternatives. Therefore, we experiment only with component-wise operations.

a := TF-logIDF representation of an ad
csv := compositional semantic vector of an ad
c := TF-logIDF representation of N best entries of csv
CR_+ := {[-5,+5] landing page text around any word in a or c}
$RR_+ := \emptyset$
For each candidate region $r_i \in CR_+$,
  If $cosine\_similarity(a, r_i) + cosine\_similarity(c, r_i) > \delta_+$,
  Then $RR_+ \leftarrow RR_+ \cup r_i$,
Return RR_+ as relevant regions for the given ad

Figure 5: Extracting relevant regions with compositional semantic vectors for ads

3.1.4 Extracting Relevant Regions with Compositional Semantic Vectors

Finally, Figure 5 shows the procedure to extract relevant regions with the enriched context. First, we represent the content of the ad with the *ad vector*, as described in Section 3.1.1). We then compute the *compositional semantic vector*(*csv*) of the given ad, as described in Section 3.1.3. We keep the top N entries with highest scores in the *csv* in order to keep the size of *csv* similar to that of the ad vector. This is to ensure that the extended vector will not be dominated by *csv*, which could potentially introduce topic shift.

Note that some of the compositional operations we consider involve component-wise multiplications among multiple vectors (Equation 5-6). As a result, the distribution of scores across different entries can be undesirably skewed. Thus, before combining the *csv* with the ad vector, we compute the tf-idf score for each of these N entries in *csv* in the same way tf-idf scores are computed for ad vectors. Terms that do not appear in the ad receive a tf score of 1.

Once we compute the converted compositional semantic vector c, we determine the candidate regions in the landing page in a way similar to what we described in Section 3.1.1. For any word in the landing page that also appears in either a or c, we consider the text span in [-5, +5] window as a candidate region. For each candidate region, we compute the cosine similarity between the candidate region and the ad vector, as well as the cosine similarity between the candidate region and the converted compositional semantic vector. If the sum of the two cosine similarity scores is above a certain threshold δ_+, then the candidate region is selected as a relevant region. As we can see, the overall procedure given in Figure 5 is similar to the one given in Figure 3, except the former incorporates the compositional semantic vector of the given ad in order to complement the succinct language of ad.

3.2 Term Selection *Out-of*-Context

We next explore algorithms that extract a summary-like representation of a landing page without consulting the advertisement associated with the given landing page.

3.2.1 First N unique words

One popular strategy is taking the top portion of the landing page as a summary (e.g., [1]). Albeit the simplicity, this method is known to be very effective, and often a hard baseline to beat (e.g., [6, 10]). We take up to N unique words that appear first in the given landing page.

Method	Context	DCG@1	DCG@2	DCG@3	NDCG@2	NDCG@3
Baseline	n/a	0.563	0.840	1.054	0.515	0.495
First	out	*0.594 (↑ 5.5%)	+*0.871 (↑ 3.7%)	1.072 (↑ 1.7%)	+*0.534 (↑ 3.7%)	0.503 (↑ 1.6%)
Best	out	0.591 (↑ 5.0%)	*0.866 (↑ 3.1%)	1.061 (↑ 0.7%)	*0.531 (↑ 3.1%)	0.498 (↑ 0.6%)
All	out	*0.600 (↑ 6.6%)	+0.868 (↑ 3.3%)	1.060 (↑ 0.6%)	+0.532 (↑ 3.3%)	0.498 (↑ 0.6%)
Overlap	in	0.563 (↑ 0.0%)	0.847 (↑ 0.8%)	1.052 (↓ -0.2%)	0.519 (↑ 0.8%)	0.494 (↓ -0.2%)
RR	in	0.594 (↑ 5.5%)	*0.877 (↑ 4.4%)	1.074 (↑ 1.9%)	*0.537 (↑ 4.3%)	0.504 (↑ 1.8%)
RR$-csv(\sum)$	in	*0.603 (↑ 7.1%)	+*0.877 (↑ 4.4%)	1.075 (↑ 2.0%)	+*0.538 (↑ 4.5%)	0.504 (↑ 1.8%)
RR$-csv(\prod)$	in	+*0.604 (↑ 7.3%)	+*0.884 (↑ 5.2%)	1.078 (↑ 2.3%)	+*0.542 (↑ 5.2%)	0.506 (↑ 2.2%)
RR$-csv(\prod^{smoo})$	in	+***0.611 (↑ 8.5%)**	+***0.892 (↑ 6.2%)**	***1.087 (↑ 3.1%)**	+***0.547 (↑ 6.2%)**	+***0.510 (↑ 3.0%)**

Table 1: Evaluation of different term selection strategies for landing pages.

3.2.2 *Best N unique words*

Next strategy to consider is taking up to N words that are the most representative of the landing page. We use TF-IDF weighting to extract such words.

3.2.3 *All Words*

Finally, we also try with all words from the entire landing page as an extreme case. This option is not practically as attractive however, for it does not reduce the amount of data that needs to be indexed.

4. EXPERIMENTS

To validate and compare the proposed approaches, we use a data set sampled from the sponsored search traffic of a major search engine. The query-ad pairs in this sample were evaluated for relevance by professional editorial staff. We evaluate our methods by augmenting the existing ad selection mechanisms to use the landing page features and rerank the judged pairs. In the following we first give more details about the data set, and then present the evaluation results.

4.1 Data Description

The development data consists of about 3600 query-ad pairs, and the test data consists of about 22500 query-ad pairs. In order to measure the effect of landing pages on ad selection more directly, we evaluate on only those query-ad pairs that have valid landing pages. We consider landing pages with less than 50 content words as invalid, as most of such landing pages were pages with error messages, such as a page that says the clicked link is no longer valid, or the searched item no longer exists. We also exclude URL queries from the evaluation, as the relevance of an ad for a URL query has little to do with the content of landing pages. For each query-ad pair, human editors judged the quality of ad into five different values: perfect (10.0), excellent (7.0), good (3.0), fair (0.5), bad (0.0). In advertisement retrieval, the quality of top few results is the most important. Therefore, we report the performance in terms of Discounted Cumulative Gain (DCG) and Normalized DCG (NDCG) at $k \in \{1, 2, 3\}$. The definition of DCG and NDCG are given as follows:

$$\mathrm{DCG}_k := rel_1 + \sum_{i=2}^{k} \frac{rel_i}{\log_2 i}$$

$$\mathrm{NDCG}_k := \frac{\mathrm{DCG}_k}{\mathrm{IDCG}_k}$$

where rel_i is the human graded relevant score ($rel_i \in \{10.0, 7.0, 3.0, 0.5, 0.0\}$) for the result at position i, and IDCG_k is the ideal DCG at position k.

4.2 Evaluation of Landing Page Summarization

Table 1 shows the performance of different term selection strategies for landing pages. The table omits NDCG at 1 as it is the same as DCG at 1. The brief description of each method in Table 1 is as follows:

- **Baseline:** This method corresponds to our ad retrieval system without utilizing landing pages.

Next three methods extract features from landing pages without considering the context of the ad.

- **First:** This method corresponds to the approach described in Section 3.2.1. In particular, it utilizes first $N = 100$ unique words in landing pages.
- **Best:** This method corresponds to the approach described in Section 3.2.2. In particular, it utilizes best $N = 100$ unique words in landing pages.
- **All:** This method corresponds to the approach described in Section 3.2.3. In particular, it includes all words in the landing pages.

Next five methods extract features from landing pages based on the context of the ad.

- **Overlap:** This method includes only those words in the landing pages that also appeared in the given ad. The purpose of this method is to indirectly contrast the relative performance gain resulted from words in landing pages that did not appear in the ad.
- **RR:** This method corresponds to the approach introduced in Section 3.1.1. In particular, it determines **R**elevant **R**egions in landing pages based only on ad vectors. In the procedure described in Figure 3, we set $\delta = 0.03$ to select relevant regions based on the cosine similarity. We choose this value so that the size of resulting relevant regions is approximately one third of the size of all words in the landing page.
- **RR**$-csv(\sum)$**:** This method corresponds to the approach introduced in Section 3.1.4 in conjunction with the compositional operation specified by Equation 4.

That is, it determines **R**elevant **R**egions in the landing page based on both the ad vector as well as the **c**ompositional **s**emantic **v**ector(csv) of the given ad, and csv is computed by applying weighted component-wise summation of co-occurrence vectors. In the procedure described in Figure 5, we set $\delta_+ = 0.05$ to select relevant regions based on the cosine similarity scores. We choose this value so that the size of resulting relevant regions by this method is close to the size of resulting relevant regions by **RR** method described above.

- **RR**$-csv(\prod)$**:** This method corresponds to the approach introduced in Section 3.1.4 in conjunction with the compositional operation specified by Equation 5. That is, csv is computed by applying weighted component-wise multiplications of co-occurrence vectors. In the procedure described in Figure 5, we use the same $\delta_+ = 0.05$ value as above.
- **RR**$-csv(\prod^{smoo})$**:** This method corresponds to the approach introduced in Section 3.1.4 in conjunction with the compositional operation specified by Equation 6. That is, csv is computed by applying smoothed weighted component-wise multiplications of co-occurrence vectors. In the procedure described in Figure 5, we use the same $\delta_+ = 0.05$ value as above. We set the smoothing factor μ to 0.01.

In Table 1, the relative performance gain of each method with respect to the baseline is given in parentheses. We perform two statistical significant tests: Wilcoxon signed-rank test and paired Student's t-test. Numbers marked with '+' indicate the performance gain is statistically significant by Wilcoxon test, and '*' indicate the gain is statistically significant by paired t-test. Numbers in bold indicate the best performing method for each evaluation metric.

We first discuss the performance of approaches that do not take into account the context of ad: First, Best, and All. Among the three, First performs the best for most of the evaluation metrics, achieving 5.5% relative gain for DCG@1, 3.7% for DCG@2, and 1.7% for DCG@3. The performance gain with respect to the baseline is not always statistically significant however. It is interesting that First performs slightly better than Best, albeit its simplicity. This result confirms the observation from previous research that first N bytes of a web page often make a very strong baseline as a summary (e.g., [6, 10]). All three approaches make substantial improvement over the baseline for DCG/NDCG at 1 and 2. This result indicates that landing pages contain valuable information that can improve ad selection. One of our initial goal was to reduce the amount of data we need to index from the landing pages. Therefore, it is a good news that First performs at least as good as All for most metrics. The performance difference between First and All is not statistically significant.

Next we discuss the performance of approaches that extract features from landing pages based on the context of ad. The best performing approach is RR$-csv(\prod^{smoo})$, achieving 8.5% relative gain for DCG@1, 6.2% for DCG@2, and 3.1% for DCG@3. The improvement is statistically significant for all evaluation metrics. In fact, RR$-csv(\prod^{smoo})$ is the only approach that makes a statistically significant improvement for DCG/NDCG at 3. We also make the following observations:

Method	Size of selected terms
All	30.1 MB
First	12.7 MB
Best	12.4 MB
Overlap	0.8 MB
RR	9.9 MB
RR$-csv(\sum)$	9.9 MB
RR$-csv(\prod)$	9.6 MB
RR$-csv(\prod^{smoo})$	10.1 MB
$csv(\prod^{smoo})$	13.1 MB

Table 3: Size of selected terms in landing pages.

(1) Among the approaches that select relevant regions using the compositional semantic vector(csv), those that are based on multiplicative vector composition perform better than the one based on additive vector composition. This result echoes the empirical results reported by [21].

(2) Notice that all three approaches that utilizes compositional semantic vector(csv) of the ad perform better than RR, which select relevant regions only based on the ad.

(3) In general, approaches that consider the context of the ad perform better than the approaches that do not.

(4) Finally, the worst performing approach is Overlap, which indicates that the performance gain of other approaches comes from utilizing terms in landing pages that did not appear in the ad. In other words, the language used in advertisement is often too succinct, and it might not match any term used in web queries. Utilizing landing pages enables to bridge the vocabulary gap between the advertisement and the web queries.

4.3 Quality of Compositional Semantic Vectors

Having seen that RR$-csv(\prod^{smoo})$ is the best performing method in Table 1, and that it performs better than RR that does not utilize the compositional semantic vector(csv), we conduct an indirect evaluation of the quality of compositional semantic vectors. In particular, we extract 100 best entries in the compositional semantic vector in Figure 5, and use those words as features for our ad retrieval system, without extracting any relevant region from the landing pages. We denote this approach as $csv(\prod^{smoo})$.

As shown in Table 2, we find that RR-$csv(\prod^{smoo})$, the method utilizing extractive summarization of landing pages achieves overall a better performance with the exception of DCG/NDCG at 3, where $csv(\prod^{smoo})$ performs slightly better. We draw following two conclusions from this experiment: First, utilizing the compositional semantic vectors without consulting landing pages brings out a performance gain that is close to the gain of a method that uses extractive summarization of landing pages. (The difference between the two approaches is not statistically significant.) This implies that compositional semantic vectors proposed in this paper have strong utility on their own. Second, we conjecture that the performance gain from either method comes from reducing the vocabulary mismatch between the

Method	DCG@1	DCG@2	DCG@3	NDCG@2	NDCG@3
Baseline	0.563	0.840	1.054	0.515	0.495
$csv(\prod^{smoo})$	$^{+}$*0.603 (↑ 7.1%)	$^{+}$*0.882 (↑ 5.0%)	$^{+}$***1.092 (↑ 3.6%)**	$^{+}$*0.541 (↑ 5.0%)	$^{+}$***0.513 (↑ 3.6%)**
RR$-csv(\prod^{smoo})$	$^{+}$***0.611 (↑ 8.5%)**	$^{+}$***0.892 (↑ 6.2%)**	*1.087 (↑ 3.1%)	$^{+}$***0.547 (↑ 6.2%)**	$^{+}$*0.510 (↑ 3.0%)

Table 2: Evaluation of the quality of compositional semantic vectors for ads.

ad and the query by enriching the context of the ad. The fact RR-$csv(\prod^{smoo})$ achieves a better performance in general suggests that landing pages provide extra information that is not available in compositional semantic vectors.

Finally, it is worthwhile to highlight two potential utilities of extractive summarization of landing pages(RR-$csv(\prod^{smoo})$), that are not available if using only compositional semantic vectors without consulting landing pages($csv(\prod^{smoo})$).

(1) Extractive summarization of landing pages can be used to detect landing pages that are either spam or broken. That is, spam or broken pages might not contain any relevant region pertaining to the advertisement, and lack of relevant regions can signal bad landing pages.

(2) Unlike compositional semantic vectors, extractive summarization of landing pages can be potentially utilized as snippet of landing pages, when displaying the search advertising.

4.4 Data Reduction

In this section, we examine the effect of data reduction by different summarization strategies for landing pages. As shown in Table 3, the size of selected terms when using all words in the landing pages(All) amounts to 30.1MB. Most of summarization strategies reduces the data in the range of 32-43%. One exception is Overlap, which drastically reduces the data down to 3%, but Overlap performs poorly as shown in Table 1. The best performing method RR-$csv(\prod^{smoo})$ reduces the data approximately to a third in comparison to All, saving the time and space required for indexing substantially.

Notice that there are small differences in size of selected terms among different summarization strategies, even though we always set the equal upper bound (100) on the number of words selected for each landing page. The small difference between First and Best comes from the difference in lengths among different words. The difference between First (or Best) and $csv(\prod^{smoo})$ comes from the fact that landing pages might not always have as many words to hit the upper bound, while *csv* almost always have more than 100 words before truncation. Similarly, the difference between First (or Best) and some of the RR variants comes from the fact that selected regions are smaller than the original landing pages and might not have as many words to hit the upper bound.

5. RELATED WORK

We discuss the related work from three different aspects; sponsored search and content match (Section 5.1), web page summarization (Section 5.2), and applications of compositional vector space models (Section 5.3).

5.1 Sponsored Search and Content Match

Sponsored search in general and advanced match in particular have been an area of active research in the last few years. There are several approaches based on query rewriting techniques that are easy to implement on the top of exact match by mapping query to rewrites and then using the rewrites to fetch ads. [2, 28, 16]. In those approaches, ad selection is performed using a single feature, in that it is essentially based on exact match between the rewrites and the bid phrases of ads.

In contrast, a few recent approaches employ search based techniques to overcome the limited bid phrases supplied by advertisers (advanced match). For instance, Ribeiro-Neto et al. [25] examine the use of vector space model and cosine similarity as a ranking function for content match ad retrieval. To resolve the vocabulary mismatch, the *triggering page* (used as a query) is expanded by features from related pages. The proposed method in [25] is particularly suitable for content match, but the application to sponsored search is not straightforward due to lack of the triggering pages in sponsored search.

Although most of previous research for ad retrieval did not utilize landing pages for ad-side expansion, some (e.g., [25, 24]) experimented with augmenting the ad with the entire landing page to improve the content match. In contrast, we perform a more focused study of landing pages, contrasting a number of in-context and out-of-context extractive summarization techniques. In addition, our study is first to explore the use of landing pages in the context of sponsored search, where the query is much shorter than that of content match studied in [25, 24].

The ad-side expansion presented in this paper is complementary to the query side expansion in an ad retrieval system for sponsored search in [7], where queries are expanded using web search results. The sponsored search problem is then effectively mapped to contextual advertising on the search result page. The major difference from our work is that it does not consider the use of landing pages or employ any ad-side expansion technique.

The ad-side expansion can be viewed as document-side expansion, which has been examined extensively in the general IR community. An interesting study by Billerbeck and Zobel [5] demonstrates that document-side expansion is inferior to query-side expansion when the documents are long. It is worthwhile to point out that this conclusion does not extend to advertisement retrieval, since the ads are significantly shorter than the web documents used in [5]. There are several studies that show the benefit of document-side expansion by extracting features from similar documents based on the language models (e.g. [17, 20]). However, the particularity of the ad retrieval and the relationship between the landing page and the ads makes our problem significantly different than the setting explored there.

5.2 Web Page Summarization

There are a good deal of previous work investigating web page summarization (e.g., [4, 15]), but most of them are geared toward producing a human readable summaries. In contrast, our main thrust in this paper is to extract relevant regions from the landing pages in the context of sponsored search. Our empirical results demonstrate that extractive summarization of landing pages can improve the ad retrieval while reducing the amount of data that needs to be indexed. Another notable difference from the work of [4] is that we only consider summarization techniques that are unsupervised algorithms. In general, summarization techniques that require human annotated summaries are not easily applicable to our task, since such human annotations do not exist for advertisers' landing pages, and the typical web pages available for summarization tasks are much different from advertiser's landing pages.

Lam-Adesina and Jones [18] present a query expansion technique that share conceptual similarities to our work; they employ summarization techniques to make a better use of relevant documents (pseudo relevance feedback), and compare context-independent (*out-of-context*) summaries with query-biased (*in-context*) summaries. However, the actual task of focus is very different in that [18] studies query-side expansion for ad hoc information retrieval, while our work explores ad-side expansion in the context of sponsored search. Unlike [18], we investigate document summaries in the context of ads, not queries, because off-line processing of landing pages is much desirable for efficiency reasons.

5.3 Compositional Vector Space Models

The need for vector composition arises often in information retrieval (IR) and national language processing (NLP), but it has rarely been the main focus of research until recently. Researchers cast compositional vector operations in light of compositional semantics [23], and explored the utility of compositional vector space models in a number of NLP applications [14, 21, 22, 27]. Our work is the first to utilize compositional vector space models in the context of sponsored search.

Some might wonder the connection between Latent Semantic Analysis [8] and the compositional vector space models explored in this paper. Both are based on vectorial representation of semantic meaning, but the object of representation is inherently different; that is, the former analyzes the relationships between documents and terms, while the latter captures the relationships among terms. A well known problem of LSA is that it cannot represent polysemy, because each word represents a single point in the meaning space [8]. On the contrary, compositional vector space models naturally embody polysemous representation, as each co-occurrence vector may contain multiple implicit clusters of co-occurring words corresponding to different semantic meanings of the center word [21]. Therefore, a particular meaning of a polysemous word is chosen only as a result of compositional vector operations with other co-occurrence vectors.

6. CONCLUSION

In this paper, we explore a number of extractive summarization techniques for landing pages in order to enhance sponsored search ad retrieval. We contrast two hypotheses – in-context and out-of-context summarization of landing pages with respect to the advertisement intent, and find that in-context summarization techniques are more effective for improving sponsored search. Empirical results show that applying extractive summarization techniques to landing pages can reduce the amount of data that needs to be indexed significantly, while retaining or even improving the performance of ad retrieval over the method that utilize the entire landing page.

Our work is the first to utilize compositional vector space models in the context of ad retrieval. We explore a range of compositional vector operations that combine co-occurrence vectors to enrich the succinct advertisement. We then show two different ways in which the enriched ad context can be utilized. First, it helps to extract more useful regions in the landing page with respect to the ad intent. Second, the enriched ad context can be a useful resource on its own to reduce the vocabulary mismatch.

In the future, we plan to extend extractive summarization techniques presented in this paper in order to reliably detect the deceptive advertisements that link to spam or broken landing pages.

7. ACKNOWLEDGMENTS

We thank Andrei Broder, Peter Ciccolo, Donald Metzler, and Jeffrey Yuan for helpful discussions and technical assistance. We also thank the anonymous reviewers for their comments and suggestions.

8. REFERENCES

[1] A. Anagnostopoulos, A. Broder, E. Gabrilovich, V. Josifovski, and L. Riedel. Just-in-time contextual advertising. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, 2007.

[2] I. Antonellis, H. Garcia-Molina, and C.-C. Chang. Simrank++: Query rewriting through link analysis of the click graph. In *PVLDB '08: Proceeding of the International Conference on Very Large Databases*, 2008.

[3] H. Becker, A. Broder, E. Gabrilovich, V. Josifovski, and B. Pang. Context transfer in search advertising. In *SIGIR '09: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 656–657, New York, NY, USA, 2009. ACM.

[4] A. L. Berger and V. O. Mittal. Ocelot: a system for summarizing web pages. In *SIGIR '00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 144–151, New York, NY,USA, 2000. ACM.

[5] B. Billerbeck and J. Zobel. Document expansion versus query expansion for ad-hoc retrieval. In *Proc of the Tenth Australasian Document Computing Symposium*, pages 34–41, 2005.

[6] R. Brandow, K. Mitze, and L. F. Rau. Automatic condensation of electronic publications by sentence selection. *Inf. Process. Manage.*, 31(5):675–685, 1995.

[7] A. Z. Broder, P. Ciccolo, M. Fontoura, E. Gabrilovich, V. Josifovski, and L. Riedel. Search advertising using web relevance feedback. In *CIKM '08: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, 2008.

[8] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407, 1990.

[9] 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.

[10] G. Erkan and D. R. Radev. Lexpagerank: Prestige in multi-document text summarization. In *EMNLP*, Barcelona, Spain, 2004.

[11] D. Fain and J. Pedersen. Sponsored search: A brief history. In *Second Workshop on Sponsored Search Auctions*, 2006.

[12] P. W. Foltz, W. Kintsch, T. K. Landauer, and T. K. L. The measurement of textual coherence with latent semantic analysis. 1998.

[13] G. Frege. The foundations of arithmetic. Evanston, IL, 1884. Northwestern University Press.

[14] E. Giesbrecht. In search of semantic compositionality in vector spaces. In S. Rudolph, F. Dau, and S. O. Kuznetsov, editors, *ICCS*, volume 5662 of *Lecture Notes in Computer Science*, pages 173–184. Springer, 2009.

[15] A. Jatowt and M. Ishizuka. Web page summarization using dynamic content. In *WWW 2004: Proceedings of the 13th International World Wide Web Conference*, pages 344–345, New York, NY, USA, May 2004. ACM Press.

[16] R. Jones, B. Rey, O. Madani, and W. Greiner. Generating query substitutions. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 387–396, New York, NY, USA, 2006. ACM.

[17] O. Kurland and L. Lee. Corpus structure, language models, and ad hoc information retrieval. In *SIGIR*, pages 194–201, 2004.

[18] A. M. Lam-Adesina and G. J. F. Jones. Applying summarization techniques for term selection in relevance feedback. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 1–9, New York, NY, USA, 2001. ACM.

[19] T. K. Landauer and S. T. Dumais. Solution to plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, (104), 1997.

[20] X. Liu and W. B. Croft. Cluster-based retrieval using language models. In *SIGIR*, pages 186–193, 2004.

[21] J. Mitchell and M. Lapata. Vector-based models of semantic composition. In *Proceedings of ACL-08: HLT*, pages 236–244, Columbus, Ohio, June 2008. Association for Computational Linguistics.

[22] J. Mitchell and M. Lapata. Language models based on semantic composition. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 430–439, Singapore, August 2009. Association for Computational Linguistics.

[23] R. Montague. The proper treatment of quantification in ordinary English. In J. Hintikka, J. Moravcsik, and P. Suppes, editors, *Approaches to Natural Language*, pages 373–398. Reidel, Dordrecht, 1973.

[24] V. Murdock, M. Ciaramita, and V. Plachouras. A noisy-channel approach to contextual advertising. In *ADKDD '07: Proceedings of the 1st international workshop on Data mining and audience intelligence for advertising*, pages 21–27, New York, NY, USA, 2007. ACM.

[25] B. Ribeiro-Neto, M. Cristo, P. B. Golgher, and E. S. de Moura. Impedance coupling in content-targeted advertising. In *SIGIR'05*, 2005.

[26] P. Smolensky. Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artif. Intell.*, 46(1-2):159–216, 1990.

[27] D. Widdows and K. Ferraro. Semantic vectors: a scalable open source package and online technology management application. In *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, may 2008. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

[28] W. V. Zhang, X. He, B. Rey, and R. Jones. Query rewriting using active learning for sponsored search. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 853–854, New York, NY, USA, 2007. ACM.