Temporal Query Log Profiling to Improve Web Search Ranking

Alexander Kotov*
University of Illinois at Urbana-Champaign
Urbana, IL
akotov2@illinois.edu

Pranam Kolari
Yahoo! Labs
Sunnyvale, CA
pranam@yahoo-inc.com

Lei Duan
Microsoft
Mountain View, CA
lei.duan@microsoft.com

Yi Chang
Yahoo! Labs
Sunnyvale, CA
yichang@yahoo-inc.com

ABSTRACT

Temporal information can be leveraged and incorporated to improve web search ranking. In this work, we propose a method to improve the ranking of search results by identifying the fundamental properties of temporal behavior of *low-quality hosts* and *spam-prone* queries in search logs and modeling those properties as quantifiable features. In particular, we introduce the concepts of *host churn*, a measure of changes in host visibility for user queries, and *query volatility*, a measure of semantic instability of query results, and propose the methods for construction of temporal profiles from search query logs that can be used for estimation of a set of features based on the introduced concepts. The utility of the proposed concepts has been experimentally demonstrated for two language-independent search tasks: the regression-based ranking of search results and a novel classification problem of detecting *spam-prone* queries introduced in this work.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*search engine spam*; I.7.5 [**Document Capture**]: Document analysis—*document classification,spam filtering*

General Terms

Experimentation

Keywords

Search Spam, Search Logs Analysis, Temporal Data Mining

*Contributed while the author was an intern at Yahoo!

CIKM'10, October 26–30, 2010, Toronto, Ontario, Canada.

1. INTRODUCTION

Search engines are widely used tools for effectively exploring information on the Web. One of the core components of a search engine is its ranking function: when a search engine receives a user query, this function determines the order of presentation of retrieved results (documents or URLs). The main goal of the ranking process is to promote high-quality and relevant content to the top of the result list, which is an important and challenging problem by itself. In addition to that, since search engines are a highly trafficked and high-revenue generating Web resource, influencing the ranking process by promoting abusive and irrelevant yet monetizable content in search results becomes a highly sought-after capability for spammers and so-called search engine optimizers (SEOs). Consequently, improving the quality of ranking is a multi-faceted problem. While, on one hand, improvements can be achieved by proposing better methods for ranking high-quality content, on the other hand, efficient and accurate identification, demotion, and filtering of artificially promoted adversarial content is critically important to the overall quality of search results as well [20]. In this work, we propose a method that addresses these two important aspects of ranking of search results through temporal analysis of search logs.

In particular, our method focuses on quantifying the temporal changes in ranking of search results with respect to the two main concepts that we introduce, each of which is focused on two orthogonal dimensions. The first concept is *host churn*, which is aimed at quantifying the changes in temporal behavior of hosts in search results for different queries. The second concept is *query volatility*, which is a measure of semantic stability of search results for a query over time. The introduced concepts have different interpretations. When viewed from the perspective of eliminating adversarial content, *host churn* can be considered as a measure of the likelihood of in-organic host behavior and *query volatility* as a measure of the likelihood of a query being compromised by search spammers.

Application of temporal profiling to adversarial information retrieval (AIR) [22, 8, 23] is based on the observation that spammers target specific query verticals (separable subsets of all queries), that are both highly monetizable (commercial and adult queries) and have a low barrier to entry

(e.g. misspellings, tail queries), with the goal of altering the ranking of results returned for those verticals by promoting specific URLs or hosts within a short period of time. The main intuition behind the proposed method is that when a particular query is compromised by spammers, it typically results in two types of unnatural changes that can be captured through temporal analysis of search logs. Firstly, the two sets of search results, returned for a query before and after it was compromised, are likely to be different. Secondly, a successful attack on a vertical typically results in abnormal increases in search results position, query coverage, click-through rate and number of impressions for spam hosts in search results returned for queries in a compromised vertical. Therefore, by constructing the temporal profiles for hosts and queries from search logs it becomes possible to identify attempts to alter the natural ranking of search results, regardless of the specific method to achieve it. In addition to that, temporal behavior of queries in search logs can be used to characterize the properties of queries, which are often targeted by SEOs and spammers. In particular, it becomes possible to identify *spam-prone* queries, which is a novel classification problem introduced in this work.

The use of temporal profiling to improve the ranking of search results, however, is not limited by the detection of adversarial content. When viewed from the perspective of improving search results presentation, *host churn* can be considered an indicator of certain properties of documents, belonging to the host. Significant and frequent temporal changes in impressions of a host in search results of different queries may indicate that its content is either of low quality or highly temporally correlated (e.g. news). Queries with high volatility (i.e. queries, which retrieve semantically different sets of results at several distinct time points) should receive special attention, as they may also reveal potential problems with the ranking function.

The proposed concepts of *host churn* and *query volatility* are represented as a set of quantifiable features in the classification framework, which are estimated from the statistical temporal profiles constructed for the queries and hosts from the search logs. These features can be effectively used both within the traditional learning-to-rank framework to directly improve the performance of a ranking function or within the classification problem of identifying *spam-prone* queries.

We enumerate our primary contributions as follows: (i) we investigate the use of fundamental properties of temporal behavior of queries and hosts to improve the ranking of search results from traditional and adversarial perspectives; (ii) we introduce and formalize the notion of *spam-prone* queries and the problem of identification of *spam-prone* queries; (iii) we propose the first method for detection of artificially promoted adversarial content that utilizes temporal information from the search logs, and (iv) finally, our method is among the first efforts to approach search spam detection independent of the specific spamming techniques.

The rest of this paper is organized as follows. In Section 2, we summarize the previous efforts in IR and AIR that are related to present work. Methods for constructing the temporal profiles and computing the associated features for hosts and queries are presented in detail in Sections 3 and 4 respectively. In Section 5, we discuss how temporal profiles can be used for improving the quality of search ranking. In Sections 6 and 7, we present the evaluation results of the proposed method in two different contexts. Finally, Section 8 concludes the paper with a discussion of limitations of the proposed method and future work.

2. RELATED WORK

In this section, we provide an overview of the previous efforts to improve the ranking of search results by introducing a better ranking function or a method to detect and eliminate adversarial content, the two major research directions highly relevant to present work.

In recent years, the ranking problem is frequently formulated as a supervised machine learning problem [4, 30, 32, 17]. The learning-to-rank approaches are capable of combining different types of features to train the ranking function. A number of previous works have also focused on exploring the methods to obtain useful information from click-through data to benefit search relevance [27]. This information can be expressed as pair-wise preferences [7, 11], or sequential data [18], or represented as ranking features [1]. Until recently, however, only a few attempts have been made to explore the idea of leveraging temporal information to improve web search ranking. Diaz [9] proposed a solution to integrate search results from the news vertical into web search results, where the news intent is detected by either inspecting the query dynamics or using the click feedback. Zhang et al. [31] proposed a ranking score adjustment method on year qualified queries, for which a few simple but effective adjustment rules are applied to the ranking results based on the time stamps extracted from the documents.

There are several ways to approach the problem of detection of artificially promoted adversarial content. One way is to address each individual technique [19] as it appears by proposing a detection method and a counter-measure. Most of the previous work in AIR has followed this highly reactive and ad hoc path, which can be explained by the fact that spamming techniques have a very transient nature and need to be addressed within a short period of time. Since spam filtering is traditionally viewed as as a two-class classification problem, the main line of research efforts in AIR is primarily focused on designing new effective features for classifiers. Typically, most features are simple statistical measures, which, depending on the nature of captured signals, can be grouped into two major categories. The first category is topological features [28, 6], which are designed to detect irregularities in the link structure of Web pages. In particular, [6] used the topology of the Web to identify spam, based on the intuition that spam pages tend to form clusters in the Web graph, primarily due to rank boosting techniques such as link farms [29]. The other category is content-based features [14], designed to identify content irregularities, indicative of certain spamming techniques, such as content stuffing [26]. Several other ideas from traditional IR, such as applying methods for deeper analysis of web page content at different granularity levels, using either language modeling [24] or LDA [2] have been applied in the spam domain as well. In [5] search log data is represented in the form of the click-view and anti-click graphs, which combine both the query and document nodes. Nodes are assigned the distribution of category labels by propagation from a set of manually categorized pages. Within this approach, queries and documents are characterized as either spam or non-spam by the entropy of the distribution of the inferred semantic categories for graph nodes. Our work is conceptually different from the previous work in AIR in that, to the

best of our knowledge, there have been no attempts to study the fundamental properties of spam pages and queries, independent of the particular spamming methods. In this sense, temporal profiling of hosts and queries can be considered as a general pro-active method to indicate abnormal temporal behavior of hosts and queries and improve the quality of Web search ranking by eliminating the artificially promoted adversarial content.

There are several other notable directions beyond feature engineering in the general classification setting of AIR that are related to our work as well. One particularly notable direction is focused on using temporal information to improve the quality of retrieval results. Viewing search from the temporal perspective was arguably first proposed in [10]. They identify queries that require temporal tuning by correlating them with temporally relevant documents. Temporal analysis of data has also been actively explored in the context of social media [16]. The more recent research efforts in AIR [22, 8, 23] also indicate the increasing importance of temporal inferences. In particular, one of the earliest methods that used temporal information in the context of spam detection was proposed in [28]. This method is based on the observation that spam and legitimate pages exhibit different patterns of link evolution.

As follows from the above discussion of the previous efforts, our approach unifies and complements the previously proposed approaches along two different research directions. In the next section, we move on to a detailed discussion of the proposed approach.

3. TEMPORAL HOST PROFILE

Our first hypothesis is that, unlike normal hosts, temporal profiles of most low quality hosts exhibit higher host churn. This is tied to the fact that spammers compete among themselves to increase referrals from the search engines, which is the main incentive behind abusive advertisement. This increase in referrals for low quality hosts is abnormal and can be captured by quantifying churn across four key metrics, (a) the number of queries a host appears in (nQ), (b) the number of impressions for a host (nI), (c) the number of referrals/clicks from the search engines ($nClk$), and (d) the average position of a host in the queries it appears (pos). For each of these metrics normal hosts show an organic and controlled pattern of growth or decay, as opposed to low quality hosts.

Organic and inorganic temporal nature of hosts can be illustrated through an example. Table 1 shows a normal host exhibiting an organic growth. The table depicts the four metrics measured at three different months on a large sub-sample of search logs, as measured across the top ten results. We use the click-through rate $nClk/nI$ for better clarity. The normal host shows a gradual increase in impressions nI and query coverage nQ, while maintaining stable click-through and positional attributes. Table 2 shows inorganic changes for a host involved in abusive advertisements. The host in question is a popular social networking site. Illegitimate profiles were created on this host and promoted through the search engines, which was noticed at the beginning of the search log analysis. As can be seen from Table 2, over time there has been a significant drop in query coverage and the number of impressions. This churn could occur due to multiple reasons, either improved abnormal page detection by the search engines or identification and removal of these profiles by the social networking site. Independent of the mechanism of removal, the host subsequently returned to its natural level of search visibility. Therefore, it would be interesting to capture such inorganic host behavior.

nQ	nI	nClk/nI	pos
2106533	5.05E8	0.06	4.38
2065249	5.04E8	0.06	4.3
2308792	5.35E8	0.063	4.16

Table 1: Temporal host attributes (query coverage, impressions, click-through rate, average position) of a normal host during three different months. The normal host shows an organic improvement in search visibility with a fairly stable click-through rate and average search result position.

nQ	nI	nClk/nI	pos
148025	2.03E7	0.059	5.82
104005	1.32E7	0.05	5.98
257	7.1E4	0.13	2.30

Table 2: Temporal host attributes (query coverage, impressions, click-through rate, average position) of a low quality host during three different months. The spam host shows an in-organic change in search visibility, with a sharp drop in visibility once the abnormal page detection problem is resolved.

Our method proceeds as follows. First, we assume that host profiles $H = \{H^1, H^2, \ldots, H^n\}$ are available across each of the n contiguous time points. Each H^i is a $m \times 4$ matrix, with an entry H^i_{jk} representing the value of a host j on property k at the time point i. Overall, for any specific host we compute the following four temporal attributes $(nQ, nI, nClk, nI \cdot pos)$. Next, for any host i, across a temporal attribute j, the *host churn* is computed as a sum of values of the churn metric φ on $n-1$ adjacent pairs of time points as follows:

$$\phi(H_{ij}) = \sum_{k=1}^{n-1} \varphi(H^k_{ij}, H^{k+1}_{ij}) \tag{1}$$

The score is a function of the chosen churn metric φ. We use two candidate metrics to quantify churn, the first of which is a logarithmic ratio across the two time points:

$$\varphi(H^m_{ij}, H^n_{ij}) = \log \frac{H^m_{ij}}{H^n_{ij}} \tag{2}$$

Since the above metric does not fully take into account the size of a host, we use a second metric, the log-likelihood (LL) test, which has been successfully used to compare two language models to quantify surprise and select representative terms in [25]. In contrast to language models, where an n-gram is compared across the two distributions, we compare a temporal property of a host across the two time points. Using this metric, the churn for a host i on a temporal attribute j across the two time points m and n can be computed as follows:

$$\varphi(H^m_{ij}, H^n_{ij}) = 2\left(H^m_{ij} \log \frac{H^m_{ij}}{E_m} + H^n_{ij} \log \frac{H^n_{ij}}{E_n}\right) \tag{3}$$

where the normalizing term E_m w.r.t E_n, common across

Temporal Measure	Description
HQ, HQLL	*host churn* on queries
HI, HILL	*host churn* on impressions
HP, HPLL	positional *host churn*
HC, HCLL	referral/click *host churn*

Table 3: Host churn profiles extracted from the search logs across four temporal properties. Logarithmic ratio and the log-likelihood test are used to independently quantify the churn.

all hosts, (and similarly E_n w.r.t E_m) is defined as:

$$E_m = \frac{\sum_{j=1}^{m} H_{ij}^m \cdot (H_{ij}^m + H_{ij}^n)}{\sum_j H_{ij}^m + \sum_j H_{ij}^n} \quad (4)$$

We use the above two metrics of churn when comparing hosts along each of the four dimensions. Clearly, these metrics are not designed to be exclusive for low quality host detection and will surface many upcoming popular hosts as having high churn. However, such upcoming normal hosts do not typically appear in *spam-prone* verticals and tend to be outliers in the respective verticals they appear. Within *spam-prone* verticals, however, most of the hosts show abnormal *host churn*, which is a key differentiator from the less *spam-prone* verticals. Depending on the used temporal attribute, we term the derived host churn profiles as **Host Query Churn**, **Host Impression Churn**, **Host Click Churn**, and **Host Positional Churn**. The eight host churn profiles used in present work are summarized in Table 3.

4. TEMPORAL QUERY PROFILE

We next turn our attention to methods for constructing the temporal profiles of queries, based on the results they serve and the user behavior observed on such results.

We assume the availability of search logs in the following format. The input is a temporally ordered collection of n document result sets $\mathcal{R}_q = \{R^1, R^2, \ldots, R^n\}$ for a query q with query frequencies $N = \{n^1, n^2, \ldots, n^n\}$ over a set T of n discrete time points $T = \{t_1, t_2, \ldots, t_n\}$. Each result set R is an ordered set of m URLs $R = \{u_1, u_2, \ldots, u_m\}$, returned by the search engine in response to a query q. At each time slice, we also have available the user feedback information for the query in the form of clicks at position i, $CLK = \{clk_1, clk_2, \ldots, clk_m\}$, and skips at position i, $SKP = \{skp_1, skp_2, \ldots, skp_m\}$, where each skip skp_i corresponds to the number of times all other URLs but u_i are clicked. The goal of constructing the temporal query profiles is to surface highly variable queries with low user satisfaction. We propose to quantify the query variability along several major dimensions: query result set, query impressions, clicks on query results and user query session behavior. We begin by characterizing the results shuffling behavior common to many *spam-prone* verticals.

4.1 Query Result Volatility

Queries affected by abusive advertisement show results that are highly volatile. A result at a particular position at one point in time is unlikely to be at the same position in a subsequent time point. Given a collection of result sets $\mathcal{R}$ for a query q, we calculate $\mu(q)$, the result-set volatility function of q, based on the distance measure δ between individual result sets in $\mathcal{R}$. The volatility score over n time intervals is calculated in general form as:

$$\mu(q) = \sum_{i=1}^{n-1} \delta(R^i, R^{i+1}) \quad (5)$$

The chosen distance measure δ should reflect the difference between individual result sets, with lager values indicating higher difference. Therefore, the more dissimilar are the result sets, the greater the value of μ. There exists a variety of distance measures [12], that can take into account only the elements of the two sets by themselves or other factors, such as position of the elements in an ordered set. Next we describe the distance measures that we use in this work.

Jaccard distance is defined on two sets R^i and R^j as:

$$\delta(R^i, R^j) = \frac{|R^i \cup R^j| - |R^i \cap R^j|}{|R^i \cup R^j|} \quad (6)$$

Jaccard distance measures dissimilarity between two sets, with larger values corresponding to more dissimilar sets. It does not take into account either the specific positions or the ordering of elements in both sets.

KL-divergence is a measure of distance between two language models. A language model for a result set Θ_R is constructed at each time point by tokenizing the URLs in the result set on all non-alphabetic symbols. Formally, KL-divergence between the language models of the result sets, Θ_R^i and Θ_R^j is defined as:

$$\delta(\Theta_R^i \parallel \Theta_R^j) = \sum_w p(w|\Theta_R^i) \log \frac{p(w|\Theta_R^i)}{p(w|\Theta_R^j)} \quad (7)$$

We use both Jaccard distance and KL-divergence as distance measures δ in evaluating the volatility score $\mu(q)$ for a query q.

We compute the Jaccard-based measure up to different positions and refer to them as SHUF_n i.e. SHUF_5 refers to temporal Jaccard-based similarity up to the result number five. The KL-divergence based measure will be referred to as KLTEMPORAL. The most temporally unstable queries are either in the adult domain, highly monetizable or wrongly formulated by the user.

4.2 Query Impression Volatility

The number of times a query is served can also be a good indicator of whether a query is *spam-prone* or not. For instance, popular queries, like *myspace* or *facebook*, for which the normal pages have accumulated a large number of in-links, are more difficult to be compromised and hence are less targeted by abusive advertisers. Furthermore, buzzy queries are also less likely to be *spam-prone*, since buzz is not a trivial prediction. To capture such classes of queries, we elicit various temporal query properties based on the query frequency across different time points.

Specifically, we compute the mean, standard deviation, kurtosis, and Pearson coefficients. While mean and standard deviation are well-known measures, kurtosis is a measure of buzz, and Pearson coefficient is a measure of the skewness of a distribution. Given the frequency of a query across k time points $N = \{n_1, n_2, \ldots, n_k\}$ with mean $\bar{n}$, kurtosis is

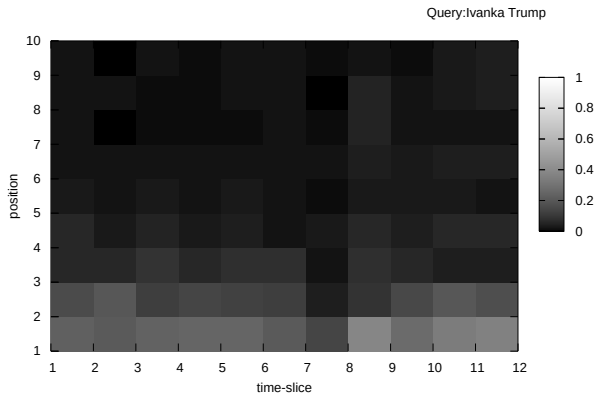

Figure 1: Temporal click pattern for an informational, non *spam-prone* query "ivanka trump". The click pattern for a non *spam-prone* query appears to be concentrated towards the top few results and is consistent across the time points.

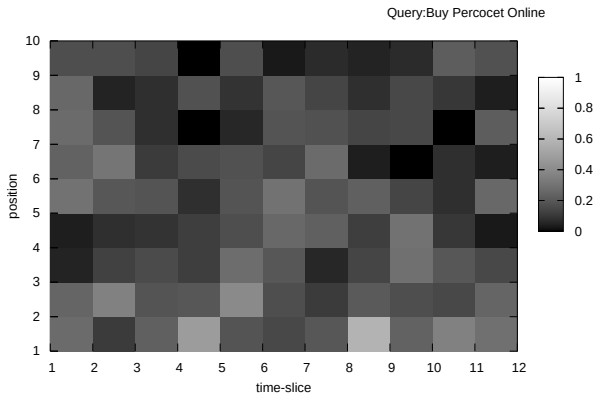

Figure 2: Temporal click pattern for a *spam-prone* query "buy percocet online". The click-pattern for *spam-prone* query lacks the uniformity seen on a non *spam-prone* query.

defined as:

$$Kurtosis(N) = \frac{\sum_{i=1}^{k}(n_i - \bar{n})^4}{\left(\frac{1}{n}\sum_{i=1}^{k}(n_i - \bar{n})^2\right)^2} - 3 \quad (8)$$

And Pearson coefficient is defined as:

$$Pearson(N) = \frac{\sum_{i=1}^{k}(n_i - \bar{n})^3}{\left(\frac{1}{n}\sum_{i=1}^{k}(n_i - \bar{n})^2\right)^{\frac{3}{2}}} \quad (9)$$

We refer to these profiles as IMP_MEAN, IMP_SD, IMP_KURTOSIS, and IMP_PEARSON.

4.3 Query Click Volatility

Unlike normal queries, *spam-prone* queries typically exhibit higher click volatility as well. Figure 1 shows a temporal click profile of an informational query not prone to be abused, while figure 2 shows a similar click profile for a *spam-prone* query. The x-axis corresponds to the time points and the y-axis corresponds to the position in search results. The intensity signifies click-through, with white shade corresponding to the click-through of 1 and black shade corresponding to the click-through of 0. As follows from the click intensity plots in Figures 1 and 2, a click-through matrix, corresponding to a *spam-prone* query, indicates a higher degree of confusion among users. The informational query shows clicks typical to queries providing good user experience with higher click density on the first few results.

To capture click discrepancies, we aggregate these metrics temporally as the mean, standard deviation, Pearson correlation and kurtosis coefficients for both the clicks clk_m and skips skp_m at each position.

4.4 Query Session Volatility

We next discuss features based on the aggregate user session behavior on the search results page. Note that the search results page typically consists of organic search results, sponsored search results, reformulation suggestions and, sometimes, news results. The primary intuition is that in *spam-prone* verticals users are seldom satisfied with the presented organic results and, therefore, are less likely to click on any of them. On the other side of the spectrum, high user satisfaction generally maps to a single click on an organic result before the end of a user session. This is generally the case for all navigational queries, i.e. queries where the user's intent is to navigate to another site (e.g. the query "facebook" leading to "facebook.com").

For our purpose, a user session is bounded by 30 minutes of user inactivity. To capture typical and atypical session behavior, we identify four distinct, but related characteristics, as measured by user clicks on the search results page. (i) *ONLYCTR* is the ratio of the number of sessions, in which the users click only on a single organic result, to the number of all sessions, in which a query appears (ii) *NOCTR* is the ratio of the number of sessions, in which the users do not click on any of the presented content (organic and sponsored) to the number of all sessions, in which a query appears (iii) *NOWCTR* is the ratio of the number of sessions, in which there are no user clicks on any of the presented organic results, to the number of all sessions, in which a query appears, and, finally, (iv) *REFCTR* is the ratio of the number of sessions, in which the users click on a query reformulation, to the number of all sessions, in which a query appears. To capture query session volatility, we compute the mean and standard deviation for the above metrics across all the time points.

5. OUR APPROACH

Having introduced the concept of temporal profiles that can help identify abnormal behavior of hosts and queries, we can now focus our attention on using such temporal profiles in two tasks: the classification problem of identifying *spam-prone* queries and the problem of using regression to

rank search results. We approach both tasks by learning a classifier on a catalog of features.

First, for the *spam-prone* query classification task our method is based on the assumption that one of the distinguishing properties of *spam-prone* queries is volatility in their result sets, clicks, as well as the underlying churn of the hosts providing such results, taken over a period of time. The query level properties can be directly used as features in the query classification task. However, churn is a property of a host, not a query. We map individual *host churns* into queries by applying the aggregate metrics (HQ_MEAN, HQLL_MEAN, HQ_SD, HQLL_SD) to the number of impressions (HI, HILL), referrals (HC, HCLL), and positional attributes (HP, HPLL) over all the hosts appearing in the results for a specific query. The main intuition is that *spam-prone* queries show a higher incidence of volatile hosts involved in abusive advertisement techniques.

Second, we use the *spam-prone* query classification results for search results ranking. The *spam-prone* query score is treated as a new query-level feature to be used by the ranking function. The baseline constitutes all existing features used by a production-level ranking function of a major search engine. The intuition is that identifying *spam-prone* queries can support new feature interactions in *spam-prone* verticals, interactions that are more robust to abusive advertisement techniques.

We use the gradient boosted decision tree (GBDT) [15] as a classifier for both problems. GBDT is an additive regression algorithm consisting of an ensemble of trees, fitted to current residuals, gradients of the loss function, in a forward step-wise manner. It iteratively fits an additive model as

$$f_t(x) = T_t(x;\Theta) + \lambda \sum_{t=1}^{T} \beta_t T_t(x;\Theta_t)$$

such that the loss function $L(y_i, f_T(x+i))$ is minimized, where $T_t(x;\Theta_t)$ is a tree at iteration t, weighted by parameter β_t, with a finite number of parameters, Θ_t and λ is the learning rate. At iteration t, tree $T_t(x;\beta)$ is induced to fit the negative gradient by least squares:

$$\hat{\Theta} := \text{argmin}_\beta \sum_{i}^{N} (-G_{it} - \beta_t T_t(x_i);\Theta)^2$$

where G_{it} is the gradient over current prediction function

$$G_{it} = \left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f=f_{t-1}}$$

The optimal weights of trees β_t are determined by

$$\beta_t = \text{argmin}_\beta \sum_{i}^{N} L(y_i, f_{t-1}(x_i) + \beta T(x_i, \theta))$$

Each node in the trees represents a split on a feature. The tuneable parameters in this model include the number of leaf nodes in each tree, the relative contribution of score from each tree, called the shrinkage, and the total number of shallow decision trees.

The relative importance of a feature S_i in such forests of decision trees is aggregated over all the m shallow decision trees [3] as follows:

$$S_i^2 = \frac{1}{M} \sum_{m=1}^{M} \sum_{n=1}^{L-1} \frac{w_l * w_r}{w_l + w_r} (y_l - y_r)^2 I(v_t = i) \qquad (10)$$

where v_t is the feature on which a split occurs, y_l and y_r are the mean regression responses from the right and left sub-trees, and w_l and w_r are the corresponding weights to the means, as measured by the number of training examples traversing the left and right sub-trees. We report on the relative importance of the extracted temporal features in the results section.

In the following sections, we discuss the use of features based on temporal profiles in the classification (Section 6) and regression (Section 7) settings.

6. SPAM-PRONE QUERY CLASSIFICATION

In this section, we report the results of an experimental evaluation of using temporal profiling to classify the queries as spam-prone and non spam-prone. A *spam-prone* query is any query frequently targeted by abusive advertisements. However, since this definition is not quantifiable, we consider a stricter definition. A query is considered to be *spam-prone* if it is targeted by either (a) multiple abusive advertisers at any given snap-shot of search results, or (b) one or more abusive advertisers at two different snapshots of search results, separated by a month. Intuitively, requirement (a) enforces that more than one result in the top ten is spam, and requirement (b) enforces that the query is consistently prone to spam. In either case, we are more interested in the class of queries repeatedly targeted by an abusive advertisement, or a group of abusive advertisements. To account for the latent causes of search engine result changes, we require that the snapshots be separated by at least one month. Although shorter time intervals could be potentially more interesting, in this work we limit ourselves to monthly intervals.

6.1 Dataset

For all experiments in this work we used the search logs from a major search engine for the year 2008. The logs feature aggregate information for the queries themselves and for the results presented for such queries, including user clicks. We remove results beyond the top ten and divide the data by one month into twelve subsets. We eliminate the queries that were served less than ten times every month and queries that were not served repeatedly across the twelve months, thus eliminating the queries in the long-tail, i.e. rare queries. Many of these queries typically repeat less often, rendering our temporal features less useful. We leave out targeting such queries as future work and focus on identifying *spam-prone* queries that are in the head and torso of the query frequency distribution. Next, from each monthly subset we remove the results that appear less than ten times, to eliminate random noise URLs. All temporal features are extracted on this data, with a total number of 3.2 Million queries. For the SHUF_n features we collapse URLs to hosts.

The natural process to obtain labeled data (*spam-prone*, *non spam-prone*) would be as follows. First, the dataset is uniformly sub-sampled for a smaller set of candidate queries. All URLs appearing on these candidate queries are independently labeled by annotators as spam and non-spam.

Queries satisfying our definition of *spam-prone* query are labeled as a positive class and all other queries are labeled as a negative class. However, given the number of URLs to be judged in order to label a single query, this direct process is highly inefficient. Furthermore, since abusive advertisements affect only a small percentage of all queries, the resulting data will be highly biased towards the non-spam queries. To work around this, we generate the candidate labeled data using three approaches and use them in combination. (i) First, we randomly sample one thousand queries from the collection at a given instance, and label the results as spam/non-spam by independent annotators. From this query sample, we consider all queries with more than one spam page in the top-10 results as *spam-prone*. Queries, with no spam pages in the top-10 are treated as *non spam-prone*. (ii) Second, we use a dictionary of queries that repeatedly (more than once) received spam complaints at a major search engine and treat all such queries as *spam-prone*, and (iii) Finally, we active-learn on the SHUF_n feature i.e. we sample for queries with SHUF_{10} values in twelve equally sized ranges. The results of these queries are then judged by editors as spam/non-spam. All queries with more than two spam results are labeled *spam-prone* and all queries with no spam results are labeled as *non spam-prone*. Although this dataset may have some overall bias, we believe this is the closest possible approach to generating a reasonable labeled dataset, given the nature of this problem and the editorial constraints that any such effort has to face.

In order to train the classifier, we created and made available a training set of roughly 500 queries, labeled as *spam-prone* and non *spam-prone*, and evaluated the effectiveness of features, extracted by using temporal profiling, in a supervised classification setting. Since the overall percentage of *spam-prone* queries is temporally highly variable, the generated positive and negative candidate samples were uniformly sub-sampled to create a balanced dataset.

6.2 Results

We use a shrinkage value of 0.05, with 10 trees and 4 leaf nodes at each tree as a configuration of the gradient boosted decision tree. We report on the precision, recall, and F_1, at the best F_1 value, using a 20-fold cross-validation on training data.

Case	F1	Precision	Recall	AUC
SPAMMEAN (1)	77.46	71.38	84.67	0.73
TEMPORAL (2)	76.73	68.48	87.23	0.80
(1) + (2)	81.07	74.92	88.32	0.84

Table 4: Results depicted at the best F1. The content-independent temporal features together are as useful as SPAMMEAN developed over many years. In addition the combination of the two methods outperforms SPAMMEAN. The AUC score is significantly higher for the combination.

The results presented in Table 4 are summarized at the best F_1 value. The baseline is set by *SPAMMEAN*, a simple approach that quantifies *spam-prone* query as the average spam value for all the hosts, appearing in search results for this query. The spam value for each host is computed as the mean of the spam scores for all the pages, belonging to a host. This spam score for each individual page is computed

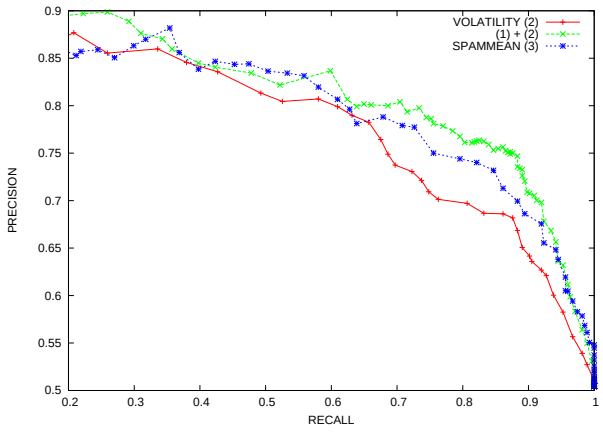

Figure 3: The precision-recall curves indicate that using the TEMPORAL features alone is close in performance to SPAMMEAN, whereas their combination (TEMPORAL + SPAMMEAN) outperforms SPAMMEAN.

Feature	Relative Importance
SKP_9_MEAN	100.00
SHUF_4	91.32
HQ_MEAN	90.33
HC_MEAN	76.14
IMP_SD	68.05
KLTEMPORAL	36.19
SKP_6_KURTOSIS	29.92
HP_MEAN	27.61

Table 5: The importance of temporal features within the *spam-prone* query classification model. A diverse range of temporal properties surface up among the top features, involving both *query volatility* and *host churn*.

using existing spam classifiers of a major search engine. This classifier has been developed over the years and makes use of both content and link-based features to evaluate a page. It is comparable in performance to the best spam classifiers in published literature and sets a challenging baseline. Interestingly, results obtained by using only the temporal features, which is completely content and language agnostic, are comparable in performance with the baseline. A combination of SPAMMEAN and temporal features, however, improves on all the metrics by around 5%, confirming the overall utility of temporal profiling. The improvements in AUC (Area Under the Curve) measure also support this conclusion. From Table 5, it follows that the temporal profiles, which are the most important for classification, include query results volatility (SHUF_4), skips (SKP_9_MEAN), query frequency, and host churn on query coverage (HQ_MEAN). However, in the combined model SPAMMEAN continues to rank as a top feature, indicating the strength of the baseline set by using this feature alone.

We next run the *spam-prone* query classifier on the entire corpus of yearly search logs, i.e. around 3.2 million queries satisfying our pre-processing criteria. The classifier was configured to provide the *spam-likelihood score* of a query in the

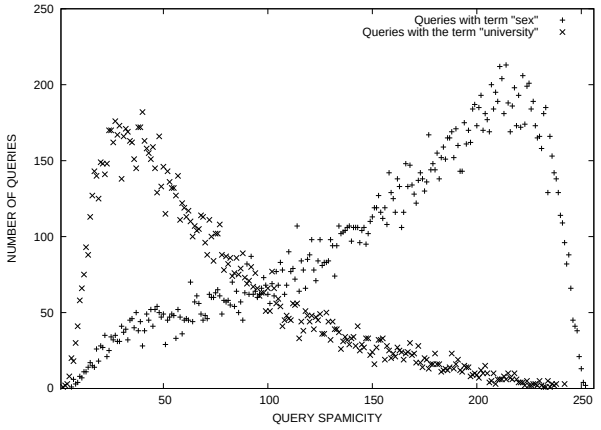

Figure 4: The plot shows a distribution of *spam-prone* query scores within the range of [0, 255] on a sample of queries. The left peak is for the queries featuring the token "university", the right peak is for the queries featuring the token "sex".

range $[0, 255]$, from the least *spam-prone* to the most *spam-prone*. We then identified some of the key terms occurring in the queries on the two boundaries of this interval, using the constructed language models and the log-likelihood test for individual terms in these language models. Both sets of terms are presented in Table 6. Adult themes, pharmacy, ringtones, gratis products etc. feature prominently in the *spam-prone* class. In Figure 4, we also plot the score distribution of all queries with the term "sex", against all queries with the term "university".

Having discussed the utility of temporal profiles for the problem of *spam-prone* query classification, we next turn our attention to their use in search ranking.

spam-prone terms	spam-free terms
sex	state
free	news
nude	school
cheap	university
escorts	department
ringtones	hospital
handbags	park
background	radio
viagra	usa

Table 6: Key indicator terms in *spam-prone* queries vs. non-spam prone queries. These terms were generated from the language models derived from running the *spam-prone* query classifier on a large corpus of queries.

7. SEARCH RESULT RANKING

The problem of search ranking is a well known application of regression modeling. In this setting, a model learns useful features and their interactions for ranking documents in response to a user query. The features are generally either (i) query-specific, i.e. an attribute of the query only (ii) document-specific, i.e. an attribute of the document only, or (iii) query-document specific, i.e. an attribute connecting a query to the document. The model is trained on a large set of labeled examples, where relevance labels are assigned to the documents for each query.

7.1 Dataset

The ranking models are trained using a large dataset of labeled examples. For each query, editors independently label five to thirty candidate URLs on a relevance scale of zero to four, where zero represents irrelevant document, and four represents highly relevant. These are considered equivalent to "Bad, Fair, Good, Excellent, Perfect" grades respectively. Using this approach we obtained editorial labels for approximately 1.8 Million documents across these five grades for a training set of around seventy thousand queries. A second independently sampled dataset of seven thousand queries with similar editorial relevance judgments was used as a validation set.

From this dataset we extracted all the features used by a popular search engine in production. These features are in the hundreds and exist in all three classes of features outlined above. Document-specific features include the spam classifiers pointed to earlier, as well as many other features of document and host authority. Query-document features include popular text-matching features used in information retrieval as well as user feedback features from the click logs. To this set of features we added a query-specific feature that measures the likelihood of a query to be compromised by spammers (*spam-likelihood score*). We hypothesize that a combination of existing features interacting with the *spam-likelihood score* will enable the search ranking function to better rank the results both in the *spam-prone* and *non spam-prone* query verticals.

7.2 Results

Different feature sets are compared using the popular Discounted Cumulative Gain (DCG) metric [21], defined up to position K as:

$$DCG(K) = \sum_{k=1}^{K} \frac{g_k}{\log_2(1+k)} \tag{11}$$

where g_k is the grade of the document at position k. We also use NDCG, defined as:

$$NDCG(K) = Z_n \sum_{k=1}^{K} \frac{g_k}{\log_2(1+k)} \tag{12}$$

where g_k is the grade of the document at position k and Z_n is a normalizing factor to ensure that the score of an ideally ranked list is one.

The baseline model is trained with all the features used by the current production of a popular search engine. The number of trees is set to 2500, with 20 terminal nodes per tree and shrinkage of 0.07. The challenger model is trained using the *spam-likelihood query classification score* ($[0, 1]$) in addition to the baseline features. For queries that do not have a *spam-likelihood score*, we set the value to "-1". The coverage of the queries by the *spam-likelihood score* is around 50%.

The results of comparing a new model against the baselines on a held-out dataset of 7000 randomly selected queries are shown in Table 7. The overall improvement across all

Metric	All Queries	Covered Queries
DCG@1 %	0.33	**0.44**
DCG@3 %	**0.42**	**0.51**
DCG@5 %	0.25	**0.28**
NDCG@1 %	0.25	**0.47**
NDCG@3 %	**0.38**	**0.50**
NDCG@5 %	0.19	0.21

Table 7: Knowledge of the *spam-likelihood score* can be useful to a search ranking function. Statistically significant results are in bold, as compared against a production baseline of a popular search engine.

queries is not statistically significant (at the 0.05 level), which is expected, given that the overall coverage of queries with the *spam-likelihood score* is around 50%. However, queries populated with the *spam-likelihood score* show significant improvement. Although these improvements appear small (less than 1%), they are targeted at the adversarial *spam-prone* vertical. We believe these improvements are a result of the *spam-prone* query feature interacting with other document level features more robust to spam. The *spam-likelihood score* based feature ranks among the top thirty features in the resulting model.

Two example queries from the validation set that show relevance improvement when using temporal profiling are shown in Tables 8 and 9. Both queries are frequently prone to spam and are generally highly optimized by search engine optimization firms. For the query "diet pill", an informational result from Wikipedia is promoted higher. For the query "baby gifts", stores more well-known in popular culture are promoted to the top. In both cases, the overall relevance, as measured by NDCG, is improved for the queries affected by temporal profiling.

8. DISCUSSION

In this work, we proposed a new approach for improving the ranking of search results by constructing the temporal profiles from the search logs that allow to quantitatively characterize the concepts of *host churn* and *query volatility*. We have experimentally shown that the features, which are based on these two concepts and are estimated from the temporal profiles, although being fully content-independent, outperform state-of-the-art baselines in two different tasks of regression-based ranking of search results and detection of *spam-prone* queries.

While the presented results are highly encouraging, the present work also opens up several directions for future work. The first direction is related to exploring other concepts, characterizing temporal behavior, besides the *host churn* and *query volatility*. Secondly, although our temporal profiling approach to the problem of classifying *spam-prone* queries outperforms existing baselines, there is still room for improvement. Specifically, there exist many other verticals that share similar traits with *spam-prone* verticals. For instance, popular trending queries on news and current affairs also often have temporally unstable search results. Many of these interactions with other verticals can be better quantified and incorporated into the ranking framework. One interesting and related direction is qualitative analysis of *spam-prone* queries to better understand the relative incidence of adult, misspelled or commercial queries along semantic and syntactic query dimensions.

In order to incorporate the dynamics of spam creation and the delays associated with the production search system, in this work we only consider monthly time intervals for construction of temporal profiles. Although we show the utility of temporal profiling using only this chosen time interval, questions about the utility of shorter time intervals, perhaps within the contexts other than spam detection, remain open.

Better understanding of *spam-prone* verticals can also enable focused ranking improvements in query classes that are highly *spam-prone*. Although we approached this problem by using an existing methodology and viewed it as an incremental addition of new features, the question of whether a specialized ranking function exclusively along this vertical is more suitable still remains unanswered. Such a function can also provide higher importance to spam-related features, thus improving the overall user satisfaction.

Even though many lines of future research remain open, we believe that the novel dimension of improving the quality of search results initiated in this paper can enable more principled and effective solutions to ranking search results and eliminating adversarial content, as compared to the state-of-the-art.

9. REFERENCES

[1] E. Agichtein, E. Brill, and S. Dumais. Improving web search ranking by incorporating user behavior information. In *Proceedings of ACM SIGIR*, pages 19–26, 2006.

[2] I. Bíró, J. Szabó, and A. A. Benczúr. Latent dirichlet allocation in web spam filtering. In *Proceedings of the 4th International Workshop on Adversarial Information Retrieval on the Web (AIRWeb'08)*, 2008.

[3] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA, 1984.

[4] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *Proceedings of ICML*, pages 89–96, 2005.

[5] C. Castillo, C. Corsi, D. Donato, P. Ferragina, and A. Gionis. Query-log mining for detecting spam. In *Proceedings of the 4th International Workshop on Adversarial Information Retrieval on the Web (AIRWeb'08)*, 2008.

[6] C. Castillo, D. Donato, A. Gionis, V. Murdock, and F. Silvestri. Know your neighbors: Web spam detection using the web topology. In *Proceedings of ACM SIGIR*, pages 423–430, 2008.

[7] O. Chapelle and Y. Zhang. A dynamic bayesian network click model for web search ranking. In *Proceedings of WWW*, pages 1–10, 2009.

[8] N. Dai, B. D. Davison, and X. Qi. Looking into the past to better classify web spam. In Fetterly and Gyöngyi [13], pages 1–8.

[9] F. Diaz. Integration of news content into web results. In *Proceedings of WSDM*, pages 182–191, 2009.

[10] F. Diaz and R. Jones. Using temporal profiles of queries for precision prediction. In *Proceedings of*

Position	Result from Improvement (Grade)	Result from Baseline (Grade)
1	http://www.dietpills.com/ (Fair)	http://www.lab88.com (Fair)
2	http://en.wikipedia.org/wiki/Diet_pill (Excellent)	http://lab88uk.com/ (Fair)
3	http://www.pricesexposed.net/ (Fair)	http://en.wikipedia.org/wiki/Diet_pill (Excellent)

Table 8: Improvements in the ranking of the query "diet pill". An informational host is promoted in the new ranking.

Position	Result from Improvement (Grade)	Result from Baseline (Grade)
1	http://babiesrus.com/ (Excellent)	http://store.babycenter.com/category... (Fair)
2	http://www.toysrus.com/shop/index.jsp?category... (Excellent)	http://www.gifts.com/ideas/baby (Bad)
3	http://www.babygiftsupply.com/ (Good)	http://www.gotobaby.com/ (Good)

Table 9: Improvements in the ranking of the query "baby gifts". More popular and well known cites are promoted in the new ranking.

ACM SIGIR, pages 18–24, New York, NY, USA, 2004. ACM.

[11] G. Dupret and C. Liao. A model to estimate intrinsic document relevance from the clickthrough logs of a web search engine. In *Proceedings of WSDM*, pages 181–190, 2010.

[12] R. Fagin, R. Kumar, and D. Sivakumar. Comparing top k lists. *SIAM Journal on Discrete Mathematics*, 17(1):134–160, 2003.

[13] D. Fetterly and Z. Gyöngyi, editors. *AIRWeb 2009, Fifth International Workshop on Adversarial Information Retrieval on the Web, Madrid, Spain, April 21, 2009*, ACM International Conference Proceeding Series, 2009.

[14] D. Fetterly, M. Manasse, and M. Najork. Spam, damn spam, and statistic: using statistical analysis to locate spam web pages. In *Proceedings of the 7th International Workshop on the Web and Databases (WebDB'04)*, pages 1–6, 2004.

[15] J. H. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29:1189–1232, 2001.

[16] N. S. Glance, M. Hurst, and T. Tomokiyo. Blogpulse: Automated trend discovery for weblogs. In *WWW 2004 Workshop on the Weblogging Ecosystem*, New York, NY USA, May 2004. ACM.

[17] J. Guiver and E. Snelson. Learning to rank with SoftRank and Gaussian processes. In *Proceedings of SIGIR*, pages 259–266, 2008.

[18] F. Guo, C. Liu, A. Kannan, T. Minka, M. Taylor, Y.-M. Wang, and C. Faloutsos. Click chain model in web search. In *Proceedings of WWW*, pages 11–20, 2009.

[19] Z. Gyöngyi and H. Garcia-Molina. Web spam taxonomy. In *Proceedings of the 1st International Workshop on Adversarial Information Retrieval on the Web (AIRWeb'05)*, 2005.

[20] M. R. Henzinger, R. Motwani, and C. Silverstein. Challenges in web search engines. *SIGIR Forum*, 36(2):11–22, 2002.

[21] K. Järvelin and J. Kekäläinen. Ir evaluation methods for retrieving highly relevant documents. In *SIGIR 2000*, pages 41–48, New York, NY, USA, 2000. ACM.

[22] Y. joo Chung, M. Toyoda, and M. Kitsuregawa. A study of link farm distribution and evolution using a time series of web snapshots. In Fetterly and Gyöngyi [13], pages 9–16.

[23] Y.-R. Lin, H. Sundaram, Y. Chi, J. Tatemura, and B. L. Tseng. Splog detection using self-similarity analysis on blog temporal dynamics. In *AIRWeb '07: Proceedings of the 3rd international workshop on Adversarial information retrieval on the web*, pages 1–8, New York, NY, USA, 2007. ACM.

[24] G. Mishne, D. Carmel, and R. Lempel. Blocking blog spam with language model disagreement. In *Proceedings of the 1st International Workshop on Adversarial Information Retrieval on the Web (AIRWeb'05)*, 2005.

[25] G. A. Mishne. *Applied Text Analytics for Blogs*. PhD thesis, University of Amsterdam, Amsterdam, 2007.

[26] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly. Detecting spam web pages through content analysis. In *Proceedings of WWW*, pages 83–92, 2006.

[27] F. Radlinski and T. Joachims. Active exploration for learning rankings from clickthrough data. In *Proceedings of SIGKDD*, pages 570–579, 2007.

[28] G. Shen, B. Gao, T.-Y. Liu, G. Feng, S. Song, and H. Li. Detecting link spam using temporal information. In *Proceedings of ICDM*, pages 1049–1053, 2006.

[29] B. Wu and B. D. Davison. Identifying link farm spam pages. In *Proceedings of WWW*, pages 820–829, 2005.

[30] J. Xu and H. Li. Adarank: a boosting algorithm for information retrieval. In *Proceedings of ACM SIGIR*, pages 391–398, 2007.

[31] R. zhang, Y. Chang, Z. Zheng, D. Metzler, and J. yun Nie. Search engine adaptation by feedback control adjustment for time-sensitive query. In *Proceedings of North American Chapter of the Association for Computational Linguistics - Human Language Technologies conference*, 2009.

[32] Z. Zheng, K. Chen, G. Sun, and H. Zha. A regression framework for learning ranking functions using relative relevance judgments. In *Proceedings of ACM SIGIR*, pages 287–294, 2007.