Click Patterns: An Empirical Representation of Complex Query Intents

Huizhong Duan
UIUC Computer Science
201 N Goodwin Ave
Urbana, IL 61801
duan9@illinois.edu

Emre Kıcıman
Microsoft Research
One Microsoft Way
Redmond, WA 98052
emrek@microsoft.com

ChengXiang Zhai
UIUC Computer Science
201 N Goodwin Ave
Urbana, IL 61801
czhai@illinois.edu

ABSTRACT

Understanding users' search intents is critical component of modern search engines. A key limitation made by most query log analyses is the assumption that each clicked web result represents one unique intent. However, there are many search tasks, such as comparison shopping or in-depth research, where a user's intent is to explore many documents. In these cases, the assumption of a one-to-one correspondence between clicked documents and user intent breaks down.

To capture and understand such behaviors, we propose the use of click patterns. Click patterns capture the relationship among clicks on search results by treating the set of clicks made by a user as a single unit. We aggregate click patterns together using a hierarchical clustering algorithm to discover the common click patterns. By using click patterns as an empirical representation of user intent, we are able to create a rich representation of mixtures of multiple navigational and informational intents. We analyze real search logs and demonstrate that such complex mixtures of intents do occur in the wild and can be identified using click patterns.

We further demonstrate the usefulness of click patterns by integrating them into a measure of query ambiguity and into a query recommendation task. We show that calculating query ambiguity as the entropy over the distribution of click patterns provides a measure of ambiguity with improved discriminative power, consistency and temporal stability as compared to previous measures of ambiguity. We explore the use of click pattern similarity and click pattern entropy in generating query recommendations and show promising results.

Categories and Subject Descriptors

H.4.m [**Information Systems Applications**]: Miscellaneous; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

CIKM'12, October 29-November 2, 2012, Maui, HI, USA.
Copyright 2012 ACM 978-1-4503-1156-4/12/10...$15.00..

General Terms

Algorithms, Experimentation, Performance

Keywords

Click pattern, click profile, query ambiguity, entropy

1. INTRODUCTION

Understanding and interpreting a user's query is the first step a web search engine must take to fulfill the user's needs. To help in this endeavor, web search engines routinely analyze the click behaviors of past searchers. These past searchers' clicks provide crucial information about query intents and are used to measure query ambiguity, influence ranking decisions and result presentation, as well as generate query recommendations.

Current query analysis techniques assume that each clicked web result provides evidence of a distinct intent. Such a simplifying assumption is problematic, however, as it ignores the many reasons why users with the same semantic intent may click on more than one web result: they may have a high-recall information need, such as when users are comparison shopping or completing a research task; or they may have an exploratory intent with no specific predefined interest. When query analysis techniques ignore such *multi-click intents*, they lose important evidence of relationships among documents and cloud their representation of users' intents.

For example, state-of-the-art measures of query ambiguity are based on measuring the entropy of clicks on web documents aggregated across users issuing a common query [21, 19, 13]. This approach, however, conflates click entropy due to multi-click intents with click entropy due to lexical and task ambiguities. For instance, the query *wedding dresses* represents a high-recall information need and, correspondingly, most users click on all of the top search results (*brides.com*, *elegantgowns.com*, *onewed.com*). In contrast, the query *auto rent* represents an aggregation of multiple distinct navigational intents, where different users each click on one of the three URLs *rentalcars.com*, *nationalcar.com* and *enterprise.com*. Despite the clear differences in query intents and user behaviors, both have similarly high click entropies.

We argue for explicitly representing multi-click intents by making *click patterns* a first-class abstraction in query analyses. Specifically, we think each individual user demonstrates a particular type of behavior when confronting a search result page. We identify the common patterns of users' behaviors using a clustering algorithm and treat these most common click patterns as a proxy representation of the user intents underlying a query. One of the interesting conceptual implications is that this allows us to represent query intent as a mixture of multiple navigational and multi-click intents.

More formally, we define intent I as the subset of relevant documents in a collection of documents $C = D_1, ..., D_n$. In particular, I corresponds to the subset of documents that a user finds useful and relevant for their corresponding semantic intent or information need, whether that is a navigational intent, a high-recall research task or an exploratory intent. We estimate I by treating our observations of users' clicks (and skips) on search results as a noisy sample of the true intents I for a query. We cluster our observations of user behavior and treat each cluster as a distinct click pattern that represents one intent I_i pulled from the set of intents $I_1, ..., I_n$ corresponding to the given query.

A benefit of using click patterns as an empirical estimation of intents is that, no matter what the true intent is, our clustered click patterns allow us to capture the similarity of intent accurately and robustly. That is, although we do not know for sure the true intent of two users issuing the same query, we can be fairly sure about their similarity: when two users have similar click patterns, we believe their true intents must also be similar.

The rest of this paper presents the following key contributions:

1. We propose click patterns as a useful, empirical representation of user intent. We show how to calculate click patterns from search query and click logs. We present the results of a click pattern based analysis of real search logs, showing examples of queries with mixtures of multiple navigational and informational intents, and demonstrating that such complex mixtures of intents do occur in the wild and can be identified using click patterns. (Section 3)

2. To demonstrate how click patterns can be integrated into existing query analyses, we propose to measure query ambiguity by computing the entropy of click patterns. We show that click pattern entropy provides a more discriminative, consistent and stable measure of user behaviors, as compared to traditional click entropy as well as the user averaged click entropy proposed by Wang and Agichtein [21]. (Section 4 and 5)

3. We further study the effect of click patterns in real world applications by using it as the fundamental unit of user behavior in query recommendation. We develop features based on click pattern entropy and pattern similarity. The new features effectively improve the baseline method which is based on word-level similarity and query popularity. (Section 6)

2. RELATED WORK

User intent/query intent analysis has been the subject of much research in recent years, especially for the purposes of search personalization and vertical search engine selection.

Understanding user's intent in search queries helps identify queries that require more personalized search results. Song et al. proposed to summarize queries as in three categories: ambiguous query, broad query and clear query [17]. They found that through topical categorization, the three types of queries are to a certain extent distinguishable according to the topical distribution. They classified the queries into these categories and estimated that 16% of queries are ambiguous in sampled logs. Teevan et al. studied how to automatically identify ambiguous queries [19]. They proposed "potential for personalization curve" for measuring the ambiguity of search queries. They measure the ability of one ranking list of search results satisfying multiple users. They show that the implicit measure (using click-through data) correlates well with the explicit measure. They also show that click entropy correlates well with "potential for personalization". Mei and Church studied the difficulty of search and personalization from an information theory perspective [13]. They used conditional entropy of URLs as an indicator of search difficulty, and compared the general search difficulty to the search difficulty when personalized with user's IP address. Their results show that personalization has huge potential in reducing search difficulty. A backoff model for personalization is also proposed where multiple layers of personalization are combined in optimizing the effectiveness.

Query intent analysis has also been studied in refining search result presentation [7] and vertical search engine selection [12, 10]. Daume and Brill proposed to group web search results based on reformulating the original query to alternative queries the user may have intended [7]. Li et al. proposed to identify queries for different vertical search engines by connecting with the close labeled queries in the click graph [12]. Hu et al. leveraged wikipedia to form query intent space, and used it to improve vertical selection[10]. However, these studies are focused on the semantic level of query intent. In our study, we approach user intent from a fundamentally different aspect by analyzing users' click behaviors.

The problem of query ambiguity has potential impact on the performance of retrieval [11, 14, 1]. The study of query ambiguity has a long history [22]. Early studies are focused on word sense disambiguation[20, 15, 6, 18, 9] with the use of dictionary and thesaurus. Allan and Raghavan studied the use of Part-of-speech Pattern to form clarification questions and reduce query ambiguity. Cronen-Townsend and Croft proposed query clarity as a measure of ambiguity [5]. Query clarity is computed as the KL-divergence of the query language model and the collection language model. The query language model is estimated from the top ranked documents of the query. Therefore, a high query clarity indicates the query is more concentrated on specific topics. Query clarity has been used often for predicting query difficulty.

Wang and Agichtein studied how to distinguish informational and ambiguous queries. They propose the use of user averaged click entropy (*average entropy*)[21]. *Average entropy* computes the average of the click entropy on the click distribution of each individual user. The assumption is that while a query may be ambiguous in general, each individual user has a clear navigational intent (which is different from others) and therefore will click on only few web pages; on the other hand, users with information seeking intent tend to click on more web pages, although the overall intent is clear. As a result, informational queries shall have higher average entropy than the ambiguous queries. However, the assumption that individual intents are clear and navigational for ambiguous queries is ungrounded. An ambiguous query can also be (completely/partially) associated with information

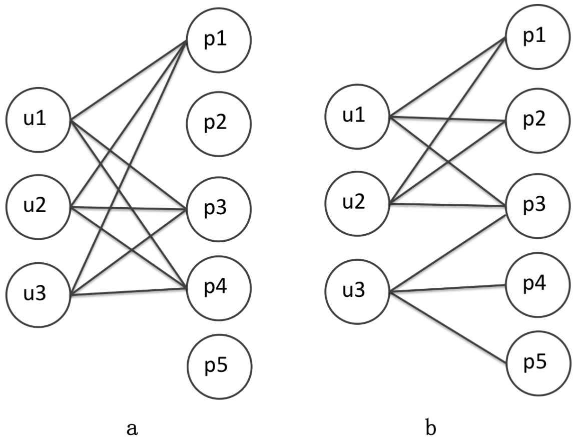

Figure 1: Examples of query clicks.

seeking intents. As an illustrating example, Figure 1 shows two click graphs between users (u) and webpages (p). Figure 1a and Figure 1b correspond to the click graphs of an informational query and an ambiguous query, respectively. It can be calculated that the two queries have the same average entropy. As the most recent work on measuring query ambiguity, we take this work as a baseline and compare our proposed technique to user averaged click entropy in Section 4.2 and Section 5.

In this paper, we propose to study query ambiguity from a different perspective. Rather than analyzing each clicked URL individually, as was done in most previous researches, we tie the concept of query ambiguity to the users' click behaviors. We propose to discover the click patterns and use pattern entropy as a new metric for query ambiguity. To the best of our knowledge, no previous research has studied click pattern for measuring query ambiguity before.

User behavior modeling is an active research area in query log analysis. Craswell et al. studied the problem of position bias in users' click behaviors [4]. Through a large scale experiment of perturbing the search engine rankings, the best explanation for position bias was found to be a model where users view results from top to bottom and leaving as soon as they see a worthwhile document. User modeling has also been studied for search evaluation [8, 23]. Dupret and Piwowarski explored the underlying hypothesis for the mean average precision metric [8]. Yilmaz et al. proposed a new evaluation metric that uses a sophisticated user model tuned by observations over many thousands of real search sessions [23]. Our work extends the study of user modeling with an exploration of fine grained user models where each query may correspond to a mixture of different types of behaviors.

3. CLICK PATTERNS

Each individual user behaves in different ways when they are presented with the same search results. However, we hypothesize that underneath the noisy click behaviors each search query corresponds to an underlying behavior model, where users obey a set of common behavioral patterns. By identifying the common patterns of users' click behaviors, we want to filter out the noises in user behaviors and achieve more accurate user models. Later, we will show its usage in many practical applications such as modeling query ambiguity and identifying similar queries for recommendation.

3.1 Definition

Formally, we define click pattern and click profile as follows.

Definition 1: Click Pattern Given a query q and its click-through document set C_q, a click pattern τ_q is a probability distribution over the C_q representing how likely each document will be clicked on. Specifically, we use multinomial distribution to model click patterns, i.e. $\tau_q = \{p(c)|c \in C_q, \sum_{c\in C_q} p(c) = 1\}$. In practice, we simplify the representation of click pattern by only considering the top 3 most probable clicks. Therefore, each click pattern is written as a ordered list of 3 elements:

$$\tau := \{(c_1, p(c_1)), (c_2, p(c_2)), (c_3, p(c_3))\}$$

where $p(c_1) > p(c_2) > p(c_3)$.

Definition 2: Click Profile We further define Γ_q as the click profile of query q. $\Gamma_q = \{p(\tau_q)\}$ is a multinomial probability distribution over all the possible click patterns of q. The probability $p(\tau_q)$ indicates how likely a user will obey a certain click pattern τ_q when C_q presented. In later discussions, we also use Γ_q to denote the set of all possible click patterns (i.e. $p(\tau_q) > 0$) of query q where it doesn't cause confusion.

3.2 Modeling and Identifying Click Patterns

Ideally, we want to discover a small set of underlying click patterns that capture the common click behavior of most users. Users obeying the same click pattern are expected to show very similar click behaviors, while users obeying different click patterns shall behave in quite different manners.

Because the clicked documents are usually quite different for each query, obtaining direct supervision is difficult. Therefore, we resort to unsupervised methods. Specifically, we employ the divisive clustering algorithm to find the patterns among the noisy sample of observed click behaviors. The detail of the algorithm is shown in Algorithm 1.

```
Algorithm 1: Divisive Clustering for Discovering Click Patterns
input : A set of click-vectors $V_q$ for query $q$, a
        similarity threshold $\sigma$, a distance function $\mathcal{F}$
output: The click profile $\Gamma_q$ for $q$
Init Cluster $c_0$ with all click vectors in $V_q$
Init Result list $l$
Enqueue $c_0$ to queue $s$
while $s$ is not empty do
    Dequeue first item $c$ from $s$
    Compute average intra-cluster distance $dist_c$ of $c$ with $\mathcal{F}$
    if $dist_c < \sigma$ then
        Add ($c.center$,$c.prob$) to $l$
    else
        Use KMeans to divide $c$ into two sub clusters $c'$ and $c''$
        Enqueue $c'$ and $c''$ to queue $s$
return $l$
```

The algorithm starts with a click profile of only one click pattern. Then it essentially keeps splitting the click pattern until the average intra-distance of each cluster is below a certain threshold. One merit of the algorithm is that it does not require us to preset the number of click patterns, which varies across queries. Instead, the number of click patterns is determined dynamically by the threshold σ. This aligns with our principle of modeling click patterns as we control the similarity of click behaviors of users who obey the same click pattern.

We do not specify the distance function $\mathcal{F}$ in this algorithm. Each alternative distance function $\mathcal{F}$ might lead to an interesting exploration of click patterns. In our study, we use cosine distance (one minus cosine similarity) as the distance function. We choose this distance function mainly because it makes the setting of the distance threshold σ more intuitive.

However, it is worth mentioning that here are potentially better ways for devising the distance function, e.g. by taking into account the ranking positions. This could allow us to explore more specialized behavior models. For instance, in analyzing mobile search logs, the ranking position is of particular importance. We plan to continue exploring this idea in our future work.

3.3 Exploring Click Patterns in Search Logs

In this subsection, we explore click patterns and click profiles in real search logs [1], to study how they can help us understand user behaviors and represent complex query intents.

To facilitate our exploration, we empirically categorize click patterns into three categories as follows. These categories are intended to provide the reader with an intuitive understanding of common click patterns.

- **Navigational patterns**. A navigational pattern has a single dominating URL in the click vector. It represents the user's intent of directly navigating to a particular URL. A click pattern $\tau = \{(c_1, w_1), (c_2, w_2), (c_3, w_3)\}$ is a navigational pattern if $w_1 \geq \mu w_2$ and $w_1 \geq \mu w_3$, where μ is a model parameter.

- **Informational patterns**. An informational pattern corresponds to an information seeking intent where the purpose is to acquire knowledge on a given topic. In this case, each URL has similar chance of getting clicked. In practice, we categorize a pattern $\tau = \{(c_1, w_1), (c_2, w_2), (c_3, w_3)\}$ as informational pattern if $w_1 < \mu w_2$ and $w_2 < \mu w_3$. Possible scenarios for informational patterns are "comparison shopping" or "research" of a topic.

- **Semi-navigational patterns**. Beside navigational patterns and informational patterns, there is a third category of click pattern, where more than one URL dominate the click distribution. In this case, the query intent is still mostly navigational, but the destination of the navigation is not a single URL. We refer to this type of click patterns the semi-navigational pattern. In our study, a semi-navigational pattern τ satisfies the condition $w_1 < \mu w_2$ and $w_2 \geq \mu w_3$. A common scenario for semi-navigational pattern is when there is a complementary page to the major destination of the navigation. The complementary page might not be necessary for completing the user's task, but it contains useful information and improves the user's understanding of the topic.

Table 1 shows the click profiles of several queries discovered by the proposed algorithm. The queries were sampled from the test data set released by Wang and Agichtein [21]. For each click profile, the major click patterns and their probabilities are shown. *Query 1*, *Query 2* and *Query 3* have simple click profiles as each of them consists of only one click pattern. For *Query 1*, the intent is to navigate to the website of "*radio shack*". Consequently a single navigational pattern is observed. For *Query 2*, the intent is to survey the information of "*wedding dresses*" available on the Web. Correspondingly an informational pattern is observed. For *Query 3*, while users mostly navigate to the the URL *prom-hair.org*, some find the site *prom.hairresources.net* also helpful and explore both resources. The click pattern falls into the category of semi-navigational pattern. All three queries have clear query intents.

Table 1: Examples of click patterns

Query 1: radio shack	
τ_1 (100%): Nav.	http://radioshack.com/:0.97 http://radioshack.com/search/:0.03
Query 2: wedding dresses	
τ_1 (100%): Inf.	http://brides.com/:0.42 http://elegantgowns.com/:0.38 http://onewed.com/dresses/:0.2
Query 3: prom hair	
τ_1 (64%): Sem.	http://prom-hair.org/:0.56 http://prom.hairresources.net/:0.26 http://prom-hairstyles.us/:0.04
Query 4: rental cars	
τ_1 (50%): Nav.	http://rentalcars.com/:0.83 http://priceline.com/:0.08
τ_2 (21%): Nav.	http://nationalcar.com/:0.77 http://alamo.com/:0.2 http://enterprise.com/:0.01
τ_3 (10%): Nav.	http://enterprise.com/:0.83 http://priceline.com/:0.08 http://thrifty.com/:0.04
Query 5: honda parts	
τ_1 (70%): Inf.	http://hondapartspro.com/:0.34 http://partstrain.com/:0.2 http://hondapartstore.com/:0.15
τ_2 (24%): Nav.	http://estore.honda.com/:0.76 http://partstrain.com/:0.11 http://hondapartspro.com/:0.09

Query 4 is a typical ambiguous query where different users have very different targets. Its corresponding click profile shows three major navigational patterns. *Query 5* is an interesting query whose click profile is a mixture of different types of click patterns. In fact, this type of click profile is not rare in search queries. For this particular query *honda parts*, some of the users directly navigate to the online com-

[1] For our analysis, we use the MSN search log dataset released in 2006.

pany store, while others take time to survey all the other web stores beside the official company store. Compared with the former three queries, the query intents of *Query 4* and *Query 5* are much more complex. Such a difference is well captured by the increase of complexity in their click profiles.

Having presented several examples of real queries with various mixtures of click patterns and how they match to user behavior, we turn our attention to the question of how frequently the more complex mixtures of click patterns occur in query logs.

Table 2 presents the distribution of click patterns in MSN query logs across three different samples of 5k queries. We see that in a random sample of 5k queries, not weighted by query popularity, the majority of queries (63%) have only a single click pattern, over a third (37%) of queries have multiple click patterns, and 11% have a mixture of different kinds of click patterns, as determined by a categorization of navigational, informational and semi-navigational patterns.

Table 2: Distribution of Click Patterns in MSN query log

Query sample	Single-intent	Multiple-intent	Mixed-intent-type
Random 5k	63%	37%	11%
Most popular 5k	29%	71%	30%
Highest click entropy 5k	5%	95%	68%

The distribution of click patterns shifts when we examine the most popular 5k queries. In this sample, we find significantly fewer single-intent queries (29%) as well as many more mixed-intent queries (30%). This shift in the distribution becomes more pronounced when we focus on the 5k queries with the greatest click-entropy. In this final sample of queries, we find that only 5% of queries have a single click pattern, and a clear majority of queries (68%) are associated with a mixture of different kinds of click patterns.

Based on this analysis, it is clear that click pattern's richer representation of user behavior—as compared to representations that assume each unique URL represents a distinct semantic intent—is useful in capturing the observed behavior of a significant fraction of all queries, and is even more important when focusing on the most popular queries or the high-entropy queries that are the hardest to answer.

4. CLICK PATTERNS AND QUERY AMBIGUITY

In this section, we study one important usage of click pattern, i.e. measuring query ambiguity. The measurement of query ambiguity plays a key role in search result presentation, personalized search, vertical selection, and many other applications.

Typically, query ambiguity is measured by "click entropy", which is computed as the information entropy of the distribution of user clicks. Formally, given query q and the set of clicked documents C_q, the click entropy $H_c(q)$ is computed by Equation 1:

$$H_c(q) = \sum_{d \in C_q} -p(d) \log p(d) \tag{1}$$

where $p(d)$ is the empirical probability of document d being clicked. A higher entropy value indicates the query is more ambiguous. The concept of information entropy was originally proposed by [16] to measure the value of information in a message. Click entropy, however, does not work well in discriminating ambiguous queries, especially from information seeking queries, as both scenarios may result in very similar distribution of clicks, although the search intent of the latter is much clearer.

The fundamental assumption of using click entropy as a measure of query ambiguity is that each document represent a unique "semantic meaning" of the search query. We think this is the reason that click entropy is not able to discriminate ambiguous queries effectively. Even semantically distinct web pages may not always increase query ambiguity if they play a complementary role in fulfilling the user's intent. Recently proposed measures such as domain entropy [21] begin to consider the relationship among multiple clicked URLs. However, they do not fully separate the notion of a user's intent from the lexical and task ambiguity of a query. As a result, they still cannot represent and measure the ambiguity of complex mixtures of intents shown in Section 3 that exist in real search logs.

4.1 Pattern Entropy

We propose to model query ambiguity as an empirical notion pertaining to user behaviors, in contrast to the previous measures that emphasize on the semantic ambiguity of the query. Particularly, we compute the information entropy of the click profile of a query, i.e. the empirical distribution over the click patterns as a measurement of query ambiguity. We refer to this measurement as *pattern entropy.*

Formally, given the click profile Γ_q for query q, the pattern entropy $H_p(q)$ is computed as:

$$H_p(q) = \sum_{\tau_q} -p(\tau_q) \log p(\tau_q) \tag{2}$$

where $p(\tau_q)$ is the empirical probability of τ_q, given by Γ_q.

Pattern entropy is superior to click entropy as an ambiguity measure. It yields low entropy values to both navigational and informational queries where the search intents are clear, while maintaining high entropy values for queries of ambiguous intents. Take two previously discussed queries , *wedding dresses* and *auto rent*, as examples. Although both queries have similar distribution of clicks, they have distinct click profiles. For the query *wedding dresses*, most users tend to obey the same informational pattern to explore all the URLs. For the query *auto rent*, different users form three distinct navigational patterns, leading to a more complex click profile. With the entropy of click profiles, we can recognize that *auto rent* is an ambiguous query and *wedding dresses* is a clear query with an informational intent.

4.2 Properties of Ambiguity Metrics

In this section, we identify three important properties of metrics of query ambiguity, i.e. *discriminative power*, *consistency* and *temporal stability*. We then put the previous proposed measurements to test with a synthetic search log and a real search log, according to the three properties.

- **Discriminative power**: The most important property of an ambiguity metric is the ability to discriminate ambiguous queries from queries of clear search

Table 3: Comparison of ambiguity metrics on synthetic queries

Query	Description	Click Entropy	Avg Entropy	Pattern Entropy
a	all users click on one same URL.	0.00	0.00	0.00
b	all users perform 10 random clicks on 10 URLs.	3.31	2.57	0.00
c	half of the users perform 10 random clicks on 10 URLs.	3.31	2.57	0.00
d	all users perform 5 random clicks on 5 URLs.	2.31	1.52	0.00
e	two groups of users, each of first group click on the same URL, the second group all click on a different URL.	0.99	0.00	1.00
f	two groups of users, each of first group performs 5 random clicks on 5 URLs; second group perform 5 random clicks on the 5 different URLs.	3.31	1.59	1.00
g	two groups of users, each of first group performs 5 random clicks on URL 1-5; second group perform 5 random clicks on the URL 3-8.	2.85	1.56	0.97
h	three groups of users, each perform 3 random clicks on 3 different set of URLs (1-3,4-6,7-9).	3.31	1.17	1.58
i	three groups of users, each perform 5 random clicks on 3 overlapping set of URLs (1-5,3-8,6-10).	3.24	1.57	0.99

intents. That being said, the metric should distinguish ambiguous queries not only from navigational queries, but also from informational queries.

- **Consistency**: It is also important that an ambiguity metric generate consistent output for the same type of queries. The number of log entries for different queries usually vary a lot, even when they are of the same type (navigational, informational or ambiguous). A good query ambiguity metric should not be affected too much by this factor.

- **Temporal Stability**: Temporal stability is a property of particular practical importance. Query log is a continuous data stream and we can only process the log within a certain temporal period at a time. The analysis on the query log needs to be updated from time to time. Therefore, being able to generate temporal stable results is an important criteria for any query log analysis method.

In the following discussions, we first test the discriminative power and consistency of different ambiguity metrics with a synthetic query log. Then we continue to study the temporal stability with a real query log.

4.2.1 Discriminative Power and Consistency

To test the discriminative power and consistency of different ambiguity metrics and to better understand their behaviors under controlled conditions, we created a synthetic query log where different types of user behaviors were observed for different queries. The query ambiguity metrics we study are click entropy, average entropy [21] and pattern entropy. Essentially, average entropy is computed as the average of each user's click entropy.

Table 3 shows the comparison for different ambiguity metrics on the synthetic queries. The first column of the table is the query ID. The second column describes how the synthetic queries are generated. The rest of the columns are values from different ambiguity metrics.

Query a is a clear navigational query where every user click on the same document. In this case every metrics yields the lowest value 0. Query b, c and d are clear, informational queries intended to test the consistency of the metrics. In each of the three queries, the users have a clear intention of exploring a set of URLs. Query b has in total 20 users. Query c reduces the number of participants by half to simulate the lack of data (data sparsity). Query d is different from b and c as it targets at a document set of half the size. Yet they all have a clear informational intent. We see neither click entropy nor average entropy is consistent on clear intent queries $a - d$ as they give low value to navigational queries and high value to informational queries. Pattern entropy, however, consistently generates the lowest value for all these the four queries.

Query $e - i$ are all ambiguous queries. Query e, f and g each has two groups of users with different behaviors. Query e has two different navigational patterns, while query f and g both have two informational patterns. We see that either the click entropy metric and the average entropy metric give consistent results to the three queries. They assign relative low entropies to queries with navigational patterns and high entropies to the ones with more informational patterns. In contrast, pattern entropy is consistent for the three queries. Query h and i both have three types of user behaviors. They are even more ambiguous than Query e, f and g. There is, however, no reflection of the increase in ambiguity in click entropy or average entropy. The average entropy of query h even decreases as the number of individual clicks is reduced. Query h shows an increase in pattern entropy which is in accordance with the increase in the level of ambiguity. However, our algorithm does not recognize all the three patterns in query i. This is because the second click pattern on documents 3-8 overlaps much with both the other two patterns on documents 1-5 and 6-10. As a result, instances from the second random click patterns are mistaken as from the other two patterns.

Overall we see pattern entropy is superior in discriminating the ambiguous queries and demonstrates more consistency in dealing with queries of the same level of ambiguity, compared to click entropy and average entropy. We will further discuss the discriminative power of the proposed metric in the later discussions when we perform automatic classification of ambiguous queries.

4.2.2 *Temporal Stability*

Unlike discriminative power and consistency that can be demonstrated with synthetic queries, temporal stability must be tested with large scale real search logs. We use the MSN search query log in the following discussions to perform further analysis on the three different ambiguity metrics, i.e. click entropy, average entropy and pattern entropy.

Our goal is to test whether a metric generates stable results in different time periods. We first randomly sample a set of 5000 queries, denoted as *rand5k*, from the MSN query log. The MSN query log spans over an entire month from May 1st, 2006 to May 31st, 2006. Therefore, we extract the log entries of the queries in *rand5k* and split them into two buckets (May 1st to May 15th and May 16th to May 31st). We then compute the ambiguity metrics on the two buckets of logs and draw the scatter plots of click entropy, average entropy and pattern entropy in Figure 2, Figure 3 and 4, respectively. In addition, we compute the correlation between the values of each metric calculated with different time period's data:

$$cor(X,Y) = \frac{\sum_{i=1}^{N}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i-\bar{x})^2\sum_{i=1}^{N}(y_i-\bar{y})^2}} \qquad (3)$$

We can see that both click entropy and pattern entropy show strong correlation in the two halves of months' data, with *cor* of 0.81 and 0.68, respectively. Therefore, they tend to generate stable results across different time periods. Average entropy, on the other hand, does not show strong visual correlation and has a low *cor* value of 0.54. This indicates that unlike the other two metrics, the computation of average entropy is more volatile in terms of time. This makes it less dependable for query log analysis and application purposes.

It is interesting to see that in all three figures, many data points falls onto the axis. This phenomenon is mostly caused by the volatility of users and queries on search engines. Such queries are usually related to temporal events, such as *what is May day* (on the event "May day" which takes place in May every year) or *Typhoon Chanchu* (on the event "Typhoon Chanchu" in 2006).

5. CLASSIFYING AMBIGUOUS QUERIES

To further verify the effectiveness of the proposed ambiguity metric, we use a human labeled query set to experiment with automatic classification of ambiguous queries.

5.1 Problem Setup

We use the same dataset as used in Wang and Agichtein's work [21]. There are in total 150 queries in the dataset, labeled as either "navigational", "informational" or "ambiguous" by human annotators. The Kappa value was 0.77. The query log is extracted from the MSN search query log released in 2006. We target at identifying ambiguous queries from the query log. Therefore, instead of performing three class classification, we merge together the "clear" and "informational" queries in the query log as all clear queries. Then

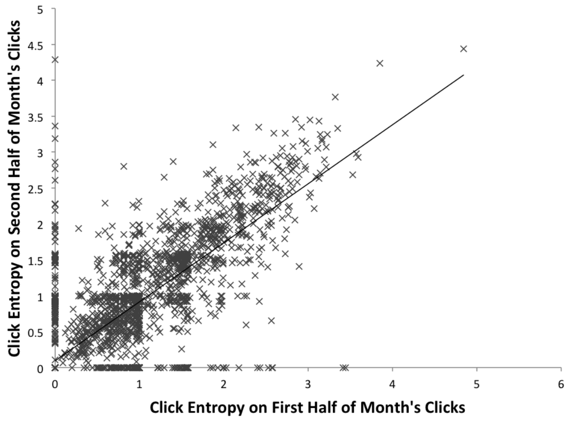

Figure 2: Temporal stability of click entropy on 5000 random queries, $cor = 0.81$.

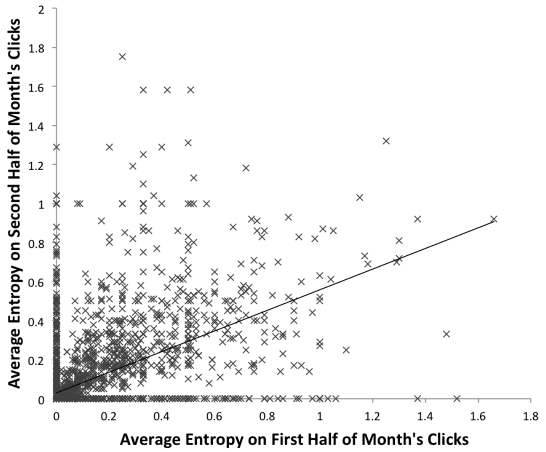

Figure 3: Temporal stability of average entropy on 5000 random queries, $cor = 0.54$.

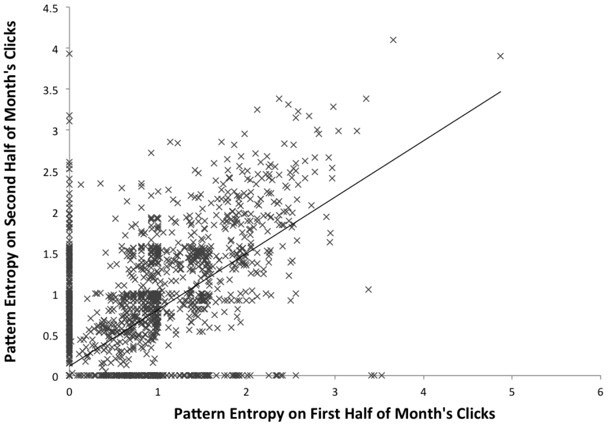

Figure 4: Temporal stability of pattern entropy on 5000 random queries, $cor = 0.68$.

we perform binary classification to separate ambiguous and clear intent ("clear" and "informational") queries. We use logistic regression as our classification method. All results are generated by 10-fold cross validation. The feature sets we use are listed in Table 4. The domain entropy features are computed by replacing the URLs with corresponding domains, where domains are obtained by truncating the URLs of the clicked web pages to their top level domain (truncating before the first "/" in the URL).

Table 4: Features used for classification

Feature set	Description of features
clk-ent	length of query frequency of query click entropy domain click entropy
avg-ent	average entropy domain average entropy
pat-ent	pattern entropy domain pattern entropy

5.2 Classification Results

Table 5 shows classification results. We can see that both average entropy and pattern entropy features improve the overall accuracy of classification. Both sets of features improve the ambiguous queries more than the clear queries. We also observe that pattern entropy features perform better than average entropy features. This confirms that pattern entropy is a superior ambiguity metric. However, we do not see further improvement in combining the average entropy features and pattern entropy features. This is probably because the additional information delivered by the two set of features overlapped with each other.

Table 5: Classification results

	Overall	Clear		Ambiguous	
	Acc	Prec	Rec	Prec	Rec
clk-ent	0.77	0.80	0.95	0.14	0.07
w/ avg-ent	0.78	0.82	0.92	0.37	0.20
w/ pat-ent	**0.81**	**0.83**	**0.96**	**0.42**	**0.21**
w/ both	0.79	**0.83**	0.94	0.37	0.19

5.3 Case Study

In Table 6 we show several examples of human annotated queries with different entropy values to help understand how pattern entropy helps improving the classification of ambiguous queries.

Most navigational queries tend to have low entropy values for any ambiguity metrics. Therefore they are easy to handle for any ambiguity metric. We see that for navigational queries, their patterns are solely navigational patterns.

Both the two queries *online auction* and *white pages* are labeled as informational queries. The average entropy for the former is relatively high. But the latter does not get the same high value. By identifying click patterns, we find that *online auction* has a single semi-navigational pattern. It therefore has a zero pattern entropy. The query *white pages*, however, has a mixture of both navigational and informational patterns, and the majority of the users are with different navigational patterns. This leads to the relatively low average entropy and pattern entropy.

The queries *song lyrics* and *ares* are labeled as ambiguous queries. The query *song lyrics* has a mixture of navigational, informational and semi-informational patterns. A possible explanation is depending on the "lyric" the user is looking for, he/she may have to try different number of websites to get it. As a result, the query has a relative high average entropy and pattern entropy. For the query *ares*, multiple navigational and semi-navigational click patterns are discovered. Since the individual intents are mostly navigational, it has a low average entropy. However, the pattern entropy is high because different patterns have comparable amount of users.

Overall, we see click pattern is quite consistent in separating the ambiguous queries from the clear queries (both navigational and informational). However, the average entropy does not work well in some cases when mixtures of click patterns exist for one query.

Table 6: Examples of entropies and patterns

Query	Avg-Ent	Pat-Ent	Patterns
Clear Navigational Queries			
chase	0.03	0.00	Nav.
ca lottery	0.01	0.13	Nav.
Clear Informational Queries			
online auction	0.81	0.00	Sem.
white pages	0.13	0.78	Nav+Inf.
Ambiguous Queries			
song lyrics	0.64	3.09	Nav.+Inf.+Sem.
ares	0.28	1.56	Nav.+Sem.

6. CLICK PATTERNS AND QUERY RECOMMENDATIONS

In this section, we explore the use of click pattern and pattern entropy in the application of query recommendation. The purpose of query recommendation is to help users explore the query space so that they can reach their intended query faster. To serve this purpose, we want to start from the close neighborhood of the query and find the queries that are likely to be used by the user, so as to shorten the distance between the user and the intended query. Baeza-Yates et al. proposed to use query log for query recommendation based on the notions of query similarity and support [2]. It was suggested later that the reduction of query ambiguity of the query should also be accounted for in query recommendation [24, 3].

The support of a query can be easily measured by the popularity of the query in the query log. In this paper, we want to test 1) whether click profiles can be used to measure the similarity of queries and 2) whether pattern entropy can effectively quantify the reduction of query ambiguity.

We compute the pattern similarity between two queries q and q' as below:

$$S_p(q, q') = \sum_{\tau \in \Gamma_q} \sum_{\gamma \in \Gamma_{q'}} p(\tau) \cdot p(\gamma) \cdot \cos(\tau, \gamma) \qquad (4)$$

where Γ_q is the set of click patterns and w_τ is the weight of pattern τ.

An alternative is to compute the maximum of the cosine similarity between any two patterns:

$$S_p(q, q') = \max_{\tau \in \Gamma_q, \gamma \in \Gamma_{q'}} \cos(\tau, \gamma) \tag{5}$$

This is more intuitive but in practice we found the sum formula in Equation 4 more effective.

It is worth noting that the strategy of recommendation for different types of queries should in fact be different. For ambiguous queries, our primary goal is to reduce the ambiguity. For navigational and information queries, where the intents are already clear, the goal is to explore more similar and interesting queries. Based on this consideration, click entropy and average entropy clearly do not fit in the task, as they both tend to assign high entropy values to informational queries. A recommendation system using click entropy or average entropy as ambiguity metric will be biased to suggesting only navigational queries.

6.1 Problem Setup

We devise query recommendation as a classification task by restricting the search space to queries that add only one term to the original query. We randomly select 200 queries that have frequency above 10 from MSN search query log release in 2006 and generate the candidate recommendations in this way. The queries with no candidate suggestions are removed from the dataset, and we finally have 127 queries. Instead of performing human annotation, we adopt an automatic annotating process. Specifically, we count how many times a candidate suggestion appear after the original query in a session with a different query log. We use the AOL search query log released in 2006 for this purpose. The reason to use a secondary query log is to avoid overfitting a single query log. As the AOL query log is released the same year as the MSN query log, the changes in queries would not be dramatic. We then label the candidate as recommendation or not by checking if the count is above 10 times. In total we have 848 queries labeled as recommendations out of 3416 candidates. We then perform binary classification for each candidate query. Note that this experiment is intended to demonstrate the usage of click pattern and pattern entropy in a real world application, rather than compete with current state-of-the-art techniques for query recommendation.

6.2 Query Recommendation Results

Table 7 shows the classification result for query recommendation. We see by adding pattern entropy and pattern similarity features, we can incrementally improve the performance of classification. The best performance by using both pattern entropy and pattern similarity with query popularity.

6.3 Case Study

In Table 8 we show the classification result for candidate query suggestions for "baby names". We can see that the recommended queries generally have high frequency, low pattern entropy and high pattern similarity with the original query.

Table 7: Results of Query Recommendation as a Classification Task

	Overall	Recommended	
	Acc	Prec	Rec
popularity	0.75	0.67	0.54
w/ pattern entropy	0.76	0.68	0.56
w/ pattern similarity and pattern entropy	**0.78**	**0.70**	**0.59**

Table 8: Query recommendation for "baby names" ($H_p(q)$=2.76)

Query	Rec	Freq.	Pat-Ent	Pat-Sim
unique baby names	Y	31	1.44	0.01
popular baby names	Y	30	1.35	0.24
girl baby names	Y	22	1.33	0.01
unusual baby names	Y	18	1.38	0.49
top baby names	Y	12	0.99	0.08
irish baby names	N	19	2.38	0.00
celebrity baby names	N	15	2.61	0.00
spanish baby names	N	14	0.99	0.00
biblical baby names	N	12	1.94	0.00
	......			

Once again, this application confirms the effectiveness of pattern entropy as an ambiguity metric. Beside query recommendation, click pattern is also potentially useful in many other applications, e.g. personalized search and diversification of search results. In the future, we plan to further study the applications of click pattern and pattern entropy.

7. CONCLUSIONS

In this paper, we propose and study the use of *click patterns* as an empirical representation of user intent in search engines, and a first-class abstraction for query analyses. We show how click patterns can be extracted from logs of user behavior and demonstrate that click patterns provide a richer representation of query intents, from multi-click intents such as high-recall research tasks to navigational intents, and mixtures of query intents of various kinds. We examine real query logs and find that the richer representation of query intents afforded by click patterns is critical for capturing the user behavior for a significant fraction of queries, and especially for popular and high entropy queries. We further demonstrate the integration of click patterns into existing query analyses by adapting traditional query ambiguity and query recommendation tasks to use click patterns as the fundamental unit of user behavior.

As search engines continue their advancement from simple document retrieval to supporting higher-level question-answering and aiding task completion, developing richer and more complete representations of query intent is of paramount importance. We believe click patterns represent a significant advance as an empirical representation of user intent, and have the potential to impact a broad set of information-retrieval technologies, from the ranking to the presentation of results, where modeling user intent is critical.

8. REFERENCES

[1] D. N. Aurelio and R. R. Mourant. The effects of web search engine query ambiguity and results sorting method on user performance and preference. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 46(14):1271–1275, 2002.

[2] R. Baeza-yates, C. Hurtado, and M. Mendoza. Query recommendation using query logs in search engines. In *In International Workshop on Clustering Information over the Web (ClustWeb, in conjunction with EDBT), Creete*, pages 588–596. Springer, 2004.

[3] S. Bhatia, D. Majumdar, and P. Mitra. Query suggestions in the absence of query logs. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, SIGIR '11, pages 795–804, New York, NY, USA, 2011. ACM.

[4] N. Craswell, O. Zoeter, M. Taylor, and B. Ramsey. An experimental comparison of click position-bias models. In *Proceedings of the international conference on Web search and web data mining*, WSDM '08, pages 87–94, New York, NY, USA, 2008. ACM.

[5] S. Cronen-Townsend and W. B. Croft. Quantifying query ambiguity. In *Proceedings of the second international conference on Human Language Technology Research*, HLT '02, pages 104–109, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.

[6] K. Darwish and D. W. Oard. Probabilistic structured query methods. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, SIGIR '03, pages 338–344, New York, NY, USA, 2003. ACM.

[7] H. Daumé, III and E. Brill. Web search intent induction via automatic query reformulation. In *Proceedings of HLT-NAACL 2004: Short Papers*, HLT-NAACL-Short '04, pages 49–52, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.

[8] G. Dupret and B. Piwowarski. A user behavior model for average precision and its generalization to graded judgments. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10, pages 531–538, New York, NY, USA, 2010. ACM.

[9] J. Gonzalo, F. Verdejo, I. Chugur, and J. Cigarrin. Indexing with wordnet synsets can improve text retrieval. In *Proceedings of CORR*, pages 38–44, 1998.

[10] J. Hu, G. Wang, F. Lochovsky, J.-t. Sun, and Z. Chen. Understanding user's query intent with wikipedia. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 471–480, New York, NY, USA, 2009. ACM.

[11] R. Krovetz and W. B. Croft. Lexical ambiguity and information retrieval. *ACM Trans. Inf. Syst.*, 10:115–141, April 1992.

[12] X. Li, Y. yi Wang, and A. Acero. Learning query intent from regularized click graphs. In *SIGIR 2008*, pages 339–346. ACM, 2008.

[13] Q. Mei and K. Church. Entropy of search logs: how hard is search? with personalization? with backoff? In *Proceedings of the international conference on Web search and web data mining*, WSDM '08, pages 45–54, New York, NY, USA, 2008. ACM.

[14] M. Sanderson. Ambiguous queries: test collections need more sense. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 499–506, New York, NY, USA, 2008. ACM.

[15] M. Sanderson and C. J. Van Rijsbergen. The impact on retrieval effectiveness of skewed frequency distributions. *ACM Trans. Inf. Syst.*, 17:440–465, October 1999.

[16] C. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423,623–656, 1948.

[17] R. Song, Z. Luo, J.-R. Wen, Y. Yu, and H.-W. Hon. Identifying ambiguous queries in web search. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 1169–1170, New York, NY, USA, 2007. ACM.

[18] C. Stokoe, M. P. Oakes, and J. Tait. Word sense disambiguation in information retrieval revisited. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, SIGIR '03, pages 159–166, New York, NY, USA, 2003. ACM.

[19] J. Teevan, S. T. Dumais, and D. J. Liebling. To personalize or not to personalize: modeling queries with variation in user intent. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 163–170, New York, NY, USA, 2008. ACM.

[20] O. Uzuner, B. Katz, and D. Yuret. Word sense disambiguation for information retrieval. In *Proceedings of the sixteenth national conference on Artificial intelligence and the eleventh Innovative applications of artificial intelligence conference innovative applications of artificial intelligence*, AAAI '99/IAAI '99, pages 985–, Menlo Park, CA, USA, 1999. American Association for Artificial Intelligence.

[21] Y. Wang and E. Agichtein. Query ambiguity revisited: clickthrough measures for distinguishing informational and ambiguous queries. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 361–364, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[22] S. F. Weiss. Learning to disambiguate. *Information Storage and Retrieval*, 9(1):33–41, 1973.

[23] E. Yilmaz, M. Shokouhi, N. Craswell, and S. Robertson. Expected browsing utility for web search evaluation. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 1561–1564, New York, NY, USA, 2010. ACM.

[24] Z.-J. Zha, L. Yang, T. Mei, M. Wang, and Z. Wang. Visual query suggestion. In *Proceedings of the 17th ACM international conference on Multimedia*, MM '09, pages 15–24, New York, NY, USA, 2009. ACM.