Behavior Driven Topic Transition for Search Task Identification

Liangda Li[¶‡], Hongbo Deng[‡], Yunlong He[‡], Anlei Dong[‡], Yi Chang[‡], and Hongyuan Zha[¶†]

[†]College of Computing
Georgia Institute of Technology
Atlanta, GA 30032

[‡]Yahoo Labs
701 First Avenue
Sunnyvale, CA 94089

[¶]Software Engineering Institute
East China Normal University
Shanghai, China 200062

{liangda, hbdeng, yunlong, anlei, yichang}@yahoo-inc.com, zha@cc.gatech.edu

ABSTRACT

Search tasks in users' query sequences are dynamic and interconnected. The formulation of search tasks can be influenced by multiple latent factors such as user characteristics, product features and search interactions, which makes search task identification a challenging problem. In this paper, we propose an unsupervised approach to identify search tasks via topic membership along with topic transition probabilities, thus it becomes possible to interpret how user's search intent emerges and evolves over time. Moreover, a novel hidden semi-Markov model is introduced to model topic transitions by considering not only the semantic information of queries but also the latent *search factors* originated from user search behaviors. A variational inference algorithm is developed to identify remarkable *search behavior* patterns, typical topic transition tracks, and the topic membership of each query from query logs. The learned topic transition tracks and the inferred topic memberships enable us to identify both small search tasks, where a user searches the same topic, and big search tasks, where a user searches a series of related topics. We extensively evaluate the proposed approach and compare with several state-of-the-art search task identification methods on both synthetic and real-world query log data, and experimental results illustrate the effectiveness of our proposed model.

General Terms: Algorithm, Experimentation, Performance

Keywords: Markov model, search task identification, search behavior

1. INTRODUCTION

In recent years, commercial search engines interact with their users through all manner of new features, presentations and experiences to meet users' *information needs*. A series of interactions between user and search engine can be necessary to satisfy a single information need. A *search task* is defined [23, 15, 28] as a set of queries serving for the same *information need*. Different from *query sessions* that are often defined as a sequence of queries issued within a fixed period of time [13, 27], search task is increasingly recognized as a more suitable atomic unit to measure search engine performance than a single query or session. Essentially, search task identification plays an important role in understanding users' search intentions, tracking their satisfaction, and improving downstream search engine applications as well as new features, such as query suggestion [9, 15] and personalized search [30].

Appropriate identification of search tasks in query sequences is a challenging problem because people find it difficult to track how and when users' search intentions come into being, evolve over time, and finally come to an end. The granularities of search tasks vary case by case [18, 29, 4], including both small ones where a user searches the same topic with an atomic information need (i.e., *search goal* as defined in [18]), and large ones where a user searches a series of related topics with an extended information need (i.e., *search mission* as defined in [18]). For example, a trip planning search task may include progressive subtasks such as flight booking, hotel booking, car rental, weather and routes inquiries, where these subtasks are highly correlated with each other sequentially. Another complex search task is that a breaking news search of Nobel Prize winner is likely to evolve to an exploratory search task of studying a certain scientific domain. By simply grouping queries together by topic, it cannot handle various granularities and evolving intentions of search tasks. Taking the above trip planning search task as an example, each subtask may belong to a different topic, thus the whole search task will be split into several individual tasks although they attempt to accomplish the same information need. Generally speaking, such subtasks that together serve an extended information need are highly correlated, and a relatively high transition probability between their corresponding topics can be observed in query sequences. Therefore, it is very intuitive to extend well beyond the topics by modeling how query sequences progress through search tasks from the perspective of search topic transitions.

Existing methods on search task identification [18, 23, 15, 28, 20] generally solve two subproblems sequentially: 1) using queries' textual information and semantic meaning to cluster queries into search topics in observed query sequences, and 2) using obtained clusters together with temporal information to partition query sequences into search tasks. However, these methods suffer several problems. On one hand, the progressive nature of search tasks is seldom considered by previous works. For instance, two search queries of the same semantic meaning may be evolved from queries to two different search tasks and thus bear different user intentions. The query "the martian" after the query "movie showtimes" has the intention of movie ticket booking while the same query is likely to have intention for the book title af-

WWW 2016, April 11–15, 2016, Montréal, Québec, Canada.
ACM 978-1-4503-4143-1/16/04.
DOI: http://dx.doi.org/10.1145/2872427.2883047..

ter the query "audio books" is issued. From another perspective, if two queries from two closely related topics where it is likely one will evolve to the other, they may belong to a search task with an extended information need. Therefore it is very important to model the transition of topics various situation in observed query logs, which is neglected by many existing works. On the other hand, most of the existing methods do not consider the latent factors of search tasks such as personalities and skills of the users, user feedbacks of the previous query and the inherent complexity of the search task. Thus, an effective method for identifying search tasks requires not only an accurate inference of query's topic-membership, but also an appropriate estimation of the probability of topic transitions under different users with various contexts.

Transition of search topics in query sequences is personalized and contextual [11, 12] from the following several perspectives. First, the personalities and skills of users may have an impact on the search process. Patient users are less likely to abandon the search task when irrelevant results are presented for the initial queries. It has also been studied in several works [32, 31] that user's familiarity with the topic domain has significant influence on user search behaviors. For example, experts illustrated significantly higher search success rate for their familiar topics. Second, user's next search choice is affected by the interactions between the user and search engine. In recent years, commercial search engines extend the traditional "10 blue links" experience on search engine results page (SERP) by providing additional services based on contextual information such as location, time of the day and ongoing real-world events. Different search experiences [16] such as cards, direct answer and vertical results (image, videos, etc.) may trigger further user interactions, thus have large impact on search task development and transitions. Third, search tasks have their inherent complexity and different types of search tasks varies in terms of characteristics such as number of queries and lasting time. Intuitively, the inherent complexity, together with user's current progress of the search task, will determine the possibility of topic transitions in the next query. A search task which aims at learning a new domain will probably include more *informational* queries [8] as compared to tasks with particular web pages in mind such as pizza ordering and address inquiry. All of the aforementioned latent factors have their influence on the transition of search topics in search sequences and hence on the formulation of search tasks, which require an advanced model to take them into consideration.

In this work, we propose a new method to identify search tasks and estimate users' information needs based on both the content and contextual information. A novel generative model based on hidden semi-Markov model is proposed which assumes that the transition of topics in query sequences is subject to not only the semantic information of queries but also the corresponding *search factors*. The concept of *search factors* is first introduced in this paper to model the latent factors of formulating a search task, which are assumed to be implicated by the interactions between the user and the search engine, i.e., search behaviors, such as issuing query, clicking URL, turning page, etc. By mining the rich information in query logs, the proposed method represents search behaviors using statistical features such as number of queries, number of clicks, click number per query (CNPQ), and time duration of query sequences[1]. A fast mean-field variational inference algorithm is then developed to infer the *search factor* as well as the topic membership associated with each query. Finally, the proposed method tackles the search task identification problem using the learned topic membership for each query and is able to build search task hierarchy using the probability of transition between topics.

The contribution of this work is three folds: 1) a novel generative model to identify search tasks via topic membership and topic transition probabilities; 2) a novel hidden semi-Markov process to model the topic transitions along with the proposed search factors which are used to model the latent factors originated from user search behaviors; 3) the use of topic membership and topic transition probability to distinguish search goal and search mission [18] in query sequences. Our proposed method is evaluated on both synthetic and real-world data, including both AOL and Yahoo query log data. We compare the performance of our model with some alternative Markov models and search task identification approaches. The experimental results demonstrate that our method can not only more accurately identify search tasks embedded in query logs, but also detect better search topics than alternatives.

In the remainder of the paper, we introduce the general methodology for solving the search task identification problem in Section 2. In Section 3, we will develop a fast mean-field variational inference algorithm to solve the inference problem. The search behavior statistics used in our method is discussed in Section 4 . In Section 5, we demonstrate the advantage of our proposed method in search task identification over several baselines. We introduce the related work in Section 6, and then present our conclusions and future work in Section 7.

2. FROM BEHAVIOR DRIVEN TOPIC TRANSITION TO SEARCH TASK IDENTIFICATION

This section introduces a generative topic transition model for describing how search behaviors drive topic transitions in query sequences. It then describes the methodology of search task identification based on the topic membership associated with each query and the transition probability of topics.

The generative model assumes that there exist T topics and K *search factors* in users' query sequences and the n-th query is associated with the topic l_n and search factor Y_n. Each topic is a distribution on the vocabulary of size V such that the semantic information of the n-th query depends on the topic membership l_n associated with it. Each search factor k, on the other hand, is corresponding to a unique distribution of user behaviors. User's interactions with the search engine in the n-th query are assumed to be encoded in the search behavior feature vector $\mathbf{d}_n$ of size M, which follows the distribution associated with its search factor Y_n, i.e., Gaussian distribution with mean ω_{Y_n} and variance σ. The dynamics of search topics are described by a Markov model with the topic transition matrices δ_k's. Instead of having a single topic transition pattern throughout the search log, the model assumes that each *search factor* k determines a distinct probability matrix δ_k of topic transitions. In this sense, the search behavior and the topic of the current query together to determine the topic of the next query.

We now describe the generative model for a sequence of queries as follows:

- For each *topic* t, draw a V dimensional vector $\theta_t \sim \mathrm{Dirichlet}(\alpha)$.
- Each *search factor* k is associated with a feature vector ω_k , and a topic transition matrix $\delta_k \sim$ Symmetric-$\mathrm{Dirichlet}(\alpha'_k)$.
- For each query n:
  - Draw the content of the current query, for the i-th word in the current query: $w_{n,i} \sim \mathrm{Multinomial}(\theta_{l_n})$;

[1]The definitions of these features are introduced in Section 4.

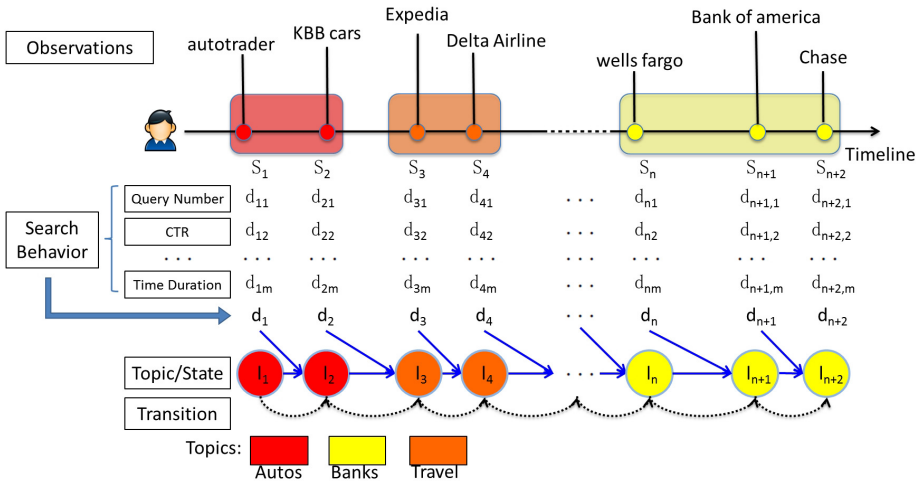


Figure 1: An Illustration of how our proposed model deals with topic transition in query sequences compared with alternative Markov models. The dash curve stands for topic transition, while the blue line denotes the dependency between states l_{n+1}, l_n and search behaviors d_n in our proposed model. We use different colors to denote different topics/states. However, in a normal hidden Markov model l_{n+1} only depends on l_n, while in a hidden semi-Markov model, l_{n+1} depends on l_n and time duration $d_{n,m}$.

- Assign it to a uniformly drawn *search factor* membership Y_n;
- Draw the real search behavior feature vector $\mathbf{d}_n$ of this query n: $\mathbf{d}_n \sim \text{Gaussian}(\omega_{Y_n}, \sigma)$;
- Draw the topic of the next query: $l_{n+1} \sim \text{Multinomial}(\delta_{Y_n,l_n})$.[2]

Comparing with the existing hidden semi-Markov model [6], the state transition in our proposed model is jointly determined by the current state and a multi-dimensional feature vector, instead of the current state and only a single dimension – the time duration on the current state. Figure 1 illustrates how our proposed model differs from alternative hidden Markov models in modeling topic transition in observed query sequences. From the aspect of hidden Markov processes, each hidden state corresponds to a topic. User's search behaviors on the current state refer to the search behaviors on the current query, which is defined to be the sequence of successive queries sharing the same topic and ending with that query. Thus we name this probabilistic model Generalized hidden semi-Markov Model (GHSMM). The graphical model representation of our GHSMM model is shown in Figure 2.

According to the above generative procedure, the topic transition probability matrix δ varies based upon the *search factor* of the current query, instead of being invariant for all queries. In other words, the topic of the query a user is to issue in the next is jointly determined by both the topic of his/her current issued query and the *search factors* of his/her current searching behaviors. For example, suppose a Ph.D. student is searching sports news for fun currently, if the time he/she spends on this query is short, he/she may feel unsatisfied, and turn to search entertainment news in the next. On the other hand, if he/she already spent a long time on this, he/she may turn to search academic papers. For queries that share the same *search factor*, it means that the users' search behavior on those queries are similar, for instance, the number of clicked documents among the returned web documents are similar, or the time spend on those queries are about the same.

We now describe our methodology of search task identification based on the behavior driven topic transition process. Since the selected topic transition rule based on user behavior imply how likely user's next query submission is influenced by his/her submission of the current query, a sequence of queries under the same topic or related topics naturally form a search task. Based on the inferred topic membership of each query and its associated topic transition rule, a thresholding of $\delta_{Y_n,l_n,l_{n+1}}$ with a constant automatically results in search task partition. The inference of search factor membership Y and topic membership L together with the learning of topic transition rules δ consequently partitions observed query sequences into search tasks. Note that our methodology of search task identification could be well aligned with the definitions of goal and mission in [18]. More concretely, the queries with the same topic membership belong to the same goal, i.e. *an atomic information need*, while queries from topics which have high transition probabilities belong to the same mission, i.e., *a related set of information needs*.

[2] δ_{Y_n,l_n} is the vector of probabilities of transitions from topic l_n to other topics under the transition matrix δ_{Y_n}.

3. INFERENCE

According to the description of GHSMM model, the joint probability of N queries $W = \{\mathbf{w}_n\}_{n=1}^N$, their corresponding topics $L = \{l_n\}_{n=1}^N$, the search behaviors of their corresponding queries $D = \{\mathbf{d}_n\}_{n=1}^N$, their search factor memberships $Y = \{Y_n\}_{n=1}^N$, and topic transition matrices δ, can be written as:

$$p(D,W,Y,L,\delta|\omega,\sigma,\alpha,\alpha') = \prod_{n=1}^{N}\prod_{i} P(w_{n,i}|l_n,\theta)\prod_{t=1}^{T} P(\theta_t|\alpha)$$
$$\prod_{n=1}^{N} P(l_{n+1}|l_n,Y_n,\delta)P(\mathbf{d}_n|Y_n,\omega)\prod_{k=1}^{K}\sum_{t,t'} P(\delta_{k,t,t'}|\alpha'_k)$$

Given the observed query sequences and search behavior statistics, there are two central inference problems associated with the GHSMM model, which we will solve in the following two subsections: 1) posterior inference of the per-query *search factor* membership, topic membership, the per-topic word distributions, and topic transition matrices, and 2) parameter estimation of each *search factor* ω and hyper-parameters of word distributions and topic transition matrices.

3.1 Variational Inference

Under the GHSMM model, given observations of both queries W and search behaviors D in query sequences, the log-likelihood for the complete data is given by $p(D,W|\omega,\sigma,\alpha,\alpha')$. Since this

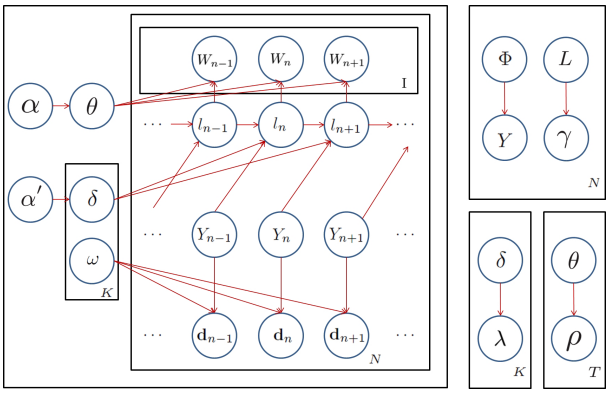


Figure 2: Graphical model representation of GHSMM and the variational distribution that approximates the likelihood. The left figure shows the graphical model representation of GHSMM, while the right figure shows the variational distribution that approximates the likelihood.

true posterior is hard to infer directly, we turn to variational inference methods [7] to approximately solve the optimization problem, whose main idea is to posit a distribution of the latent variables with free parameters, and then fit those parameters such that the distribution is close to the true posterior in Kullback-Leibler (KL) divergence. The variational distribution is supposed to be simpler than the true posterior, thus enables us to approximately solve the original optimization problem. In Figure 2, the right part shows the variational distribution that approximates the data likelihood. Following the variational inference method, we introduce a distribution q that depends on a set of free parameters, and we specify q as the mean-field fully factorized family,

$$\begin{aligned} &q(Y_{1:N}, L, \theta_{1:T}, \delta_{1:K} | \Phi_{1:N}, \gamma_{1:N}, \rho_{1:T}, \lambda_{1:K}) \\ &= \prod_n q_1(Y_n|\phi_n) q_1(l_n|\gamma_n) \prod_t q_2(\theta_t|\rho_t) \prod_k q_3(\delta_k|\lambda_k) \end{aligned}$$

where q_1 is a multinomial, q_2 is a Dirichlet, q_3 is a symmetric Dirichlet, and $\{\Phi_{1:N}, \gamma_{1:N}, \rho_{1:T}, \lambda_{1:K}\}$ are the set of free variational parameters corresponding to latent variables $\{Y, L, \theta, \delta\}$. The original likelihood can be optimized to approximate the lower bound as:

$$\begin{aligned} \mathcal{L} &= \log p(D, W|\omega, \sigma, \alpha, \alpha') \\ &\geq E_q[\log p(D, W, Y, L, \delta|\omega, \sigma, \alpha, \alpha')] \\ &- E_q[\log q(Y_{1:N}, L, \theta_{1:T}, \delta_{1:K})]. \end{aligned} \tag{1}$$

The right-hand side of above equation, which we denote as $\mathcal{L}'$, is the lower bound that we use as the surrogate to the true log-likelihood $\mathcal{L}$ in the following latent variable inference and parameter estimation. To tighten the bound with respect to the variational parameters, we are actually to maximize the alternative lower bound $\mathcal{L}'$. We employ a coordinate ascent framework for this optimization, and optimize the lower bound $\mathcal{L}'$ against each variational latent variables and the model hyper-parameter. For variational latent variables, we have

- update rules for ϕ's as:

$$\begin{aligned} \phi_{n,k} \propto \exp(&\sum_{t,t'} \gamma_{n,t}\gamma_{n+1,t'}[\Phi(\lambda_{k,t,t'}) - \Phi(\sum_{t''} \lambda_{k,t,t''})] \\ &- \frac{1}{2\sigma^2} \sum_m (d_{n,m} - \omega_{k,m})^2) \end{aligned}$$

- update rules for γ's as:

$$\begin{aligned} \gamma_{n,t} \propto \exp(&\sum_i \sum_v w_{n,i,v}[\Phi(\rho_{t,v}) - \Phi(\sum_v \rho_{t,v})] \\ &+ \sum_k \sum_{t'} \gamma_{n-1,t'}[\Phi(\lambda_{k,t',t}) - \Phi(\sum_{t''} \lambda_{k,t',t''})] \\ &+ \sum_k \sum_{t'} \gamma_{n+1,t'}[\Phi(\lambda_{k,t,t'}) - \Phi(\sum_{t''} \lambda_{k,t,t''})]) \end{aligned}$$

- update rules for ρ's as:

$$\rho_{t,v} \propto \alpha_v + \sum_n \sum_i \gamma_{n,t} w_{n,i,v}$$

- update rules for λ's as:

$$\lambda_{k,t,t'} \propto \alpha'_k + \sum_n \gamma_{n,t}\gamma_{n+1,t'}\phi_{n,k},$$

In summary, the probability of the n-th query belonging to the *search factor* k is jointly determined by the probability that topic l_n transfers to topic l_{n+1} in the transition matrix associated with the *search factor* k, and the difference between feature values of the *search factor* k and user's search behavior on that query. The probability of a query n belonging to topic t is jointly determined by: (a) Semantic clustering of queries; (b) Past influence: the transition probability from the previous state; and (c) Future influence: the transition probability to the next state.

3.2 Estimation

We use a variational expectation-maximization (EM) algorithm [10] to compute the empirical Bayes estimate of topic hyper-parameters α, topic transitions hyper-parameters α', and parameters of *search factors* ω in our GHSMM model. This variational EM algorithm aims to optimize the lower bound as shown in Eqn (1) instead of the real likelihood, which iteratively fits the variational distribution q to approximate the posterior and maximizes the corresponding bound with respect to the parameters. The latter M-step is equivalent to finding the MLE using expected sufficient statistics under the variational distribution.

Notice that a closed form solution for the approximate maximum likelihood estimate of α does not exist, we use a linear-time Newton-Raphson method, where the gradient and Hessian are

$$\frac{\partial \mathcal{L}'}{\partial \alpha_v} = N(\Psi(\sum_v \alpha_v) - \Psi(\alpha_v)) + \sum_t (\Psi(\rho_{t,v}) - \Psi(\sum_v \rho_{t,v})),$$

$$\frac{\partial \mathcal{L}'}{\partial \alpha_{v_1} \alpha_{v_2}} = N(\mathbb{I}_{(v_1=v_2)} \Psi'(\alpha_{v_1}) - \Psi'(\sum_v \alpha_v)),$$

where Ψ is the digamma function. Similar update rules can be derived for α'.

The maximum likelihood estimation of *search factors* ω's can be derived through calculating the first derivative of lower-bound $\mathcal{L}'$ against corresponding parameters. We obtain the update formulas given as follows:

$$\omega_{k,m} = \frac{\sum_n \phi_{n,k} d_{n,m}}{\sum_n \phi_{n,k}}.$$

Given the *search factor* membership ϕ_n of each query, the parameters of *search factors* ω and their corresponding topic transition matrices δ can be estimated through simple statistical counting.

In our mean-field variation inference algorithm, the computational cost of inferring variational latent variables is $O(N * (K * T^2 + K * M + T * \bar{C}) + T * V)$, where $\bar{C}$ is the average number of

Table 1: Search Behavior Features

Due to space limitation, we avoid showing accumulated features that are based on similar search actions with instant features.

Feature q	Description
Click Number	The total number of clicks on the returned results of a submitted query.
Dwell Time	The average time between a user's final action on the last query and the submission of the current query.
Click Position	The average position of clicks on the results of a query.
Time Duration	The time duration of a query.
Click Speed	The number of clicks divided by the time duration of a query.
Scanned Pages	The number of result pages user scanned for an issued query.
Time Interval	The time interval between the submission of current query and that of the next query.
Query Number	The total number of queries within the search task that current query belongs to.
Click Number Per Query(CNPQ)	The average number of clicks per query within the current search task.

words in a query, N is total number of queries, K is the number of *search factor*s, M is the dimension of search behavior features, T is the number of topics, and V is the vocabulary size. Notice that in computing γ we take the advantage of the sparsity of W which contains only $N * \bar{C}$ nonzero elements. The computational cost of the estimation of hyper-parameters is $O(K * T^2 + T * V)$. The computational cost of the estimation of the parameters of *search factors* is $O(M * K * N)$. Thus the total computational cost of our algorithm is $O(N * (K * T^2 + K * M + T * \bar{C} * V))$, where $M < T^2$ can by ensured by controlling the number of search factor features we use. Also notice that $\bar{C}$ is a small constant, and we have $V \ll N$ when the number of queries N is large enough, thus the total computational cost can be simplified to $O(N * K * T^2)$, which is linear in the number of queries with a fixed number of *search factor* patterns K, a fixed number of states/topics T.

4. SEARCH BEHAVIOR FEATURES

As the search behavior features are very valuable and useful, we experimented with many search behavior features besides those popular query content features utilized in existing approaches. The objective is to capture some key search behaviors which may influence a user's choice of the topic of his/her next query given the topic of the current issued query. We summarize these search behavior features in Table 1. Our features generally originate from statistical counting of search engine users' basic actions, such as issuing query, clicking URL, turning page. Those actions are measured by counting the number or time for each issued queries, resulting in the following features. For each query, we simply measure how many URLs are clicked (denoted as "Click Number"), and how many seconds it lasts (denoted as "Time Duration"). Moreover, for each query, we further calculate some complex statistical features, including the average time interval between each click and its following click (denoted as "Dwell Time"), the average position of clicks (denoted as "Click Position"), the number of clicks divided by the time duration (denoted as "Click Speed"), and the average number of result pages scanned by user (denoted as "Scanned Pages").

Besides features that describe users' behaviors on each single query, we also make use of users' accumulated behaviors on the search task that the current query belongs to. Those behaviors also originate from search engine users' basic actions, but are measured under the scale of search tasks instead of queries. For instance, "Click Number" measures the total number of clicks completed in the current search task, "Click Position" measures the average position of clicks in the current search task. In addition, some accumulated-behavior features collect novel information compared with instant-behavior features. For example, "Query Number" measures how many queries are already issued in the current search task, and "CNPQ" calculates the number of clicks divided by the number of the issued queries in the current search task. Notice the update of topic membership of each query can change the value of accumulated features from iteration to iteration. Based on those collected features, we are able to form a feature vector $\mathbf{d}_n$ for each query n, which is used for latent variable inference and parameter estimation in GHSMM.

5. EXPERIMENTS

In this section, we first describe alternative Markov models and search task identification approaches as our baseline models. Then we evaluate our proposed GHSMM model and compare it with the baseline models on real world data. Experimental results demonstrate the effectiveness of our proposed model.

5.1 Baseline Models

To evaluate the effectiveness of the proposed GHSMM model, we compare it with the following alternative Markov models:

`HMM[25]:` This approach is based on hidden Markov model, which assumes the topic of the next query depends only on the topic of the present query;

`HSMM[6]:` This approach is based on hidden semi-Markov model, which assumes the topic of the next query depends on both the topic of the present query and the time a user spends on the present query.

and four state-of-the-art search task identification approaches:

Bestlink-SVM [28]: This method identified search tasks using a semi-supervised clustering model based on the latent structural SVM framework. A set of effective automatic annotation rules were proposed as weak supervision to release the burden of manual annotation.

QC-HTC/QC-WCC [23]: This series of methods viewed search task identification as the problem of best approximating the manually annotated tasks, and proposed both clustering and heuristic algorithms to solve the problem. QC-WCC conducted clustering by dropping query-pairs with low weights, while QC-HTC considered the similarity between the first and last queries of two clusters in agglomerative clustering.

Reg-Classifier[18]: This method designed a diverse set of syntactic, temporal, query log and web search features, and used them in a logistic regression model to detect search tasks.

LDA-Hawkes[20]: This method casted search task identification into the problem of identify semantic influence in observed query sequences, and proposed a probabilistic model by combining LDA model with Hawkes processes to address the problem using both temporal and textual information.

In addition, since the performance of search task identification depends heavily on the quality of learned topics, we also compare the proposed model with two state-of-the-art query clustering approaches:

Session-Similarity[35]: This method evaluated query similarity based on both query sessions and query content, and used those similarity scores for query clustering.

Table 2: Inference and Estimation of GHSMM on Synthetic Data

Metric	Small Synthetic	Large Synthetic
$\frac{1}{V}\sum_v \lvert\frac{\alpha_v-\hat{\alpha}_v}{\alpha_v}\rvert$	0.129	0.285
$\frac{1}{K}\sum_k \lvert\frac{\alpha'_k-\hat{\alpha}'_k}{\alpha'_k}\rvert$	0.077	0.110
$\frac{1}{K}\sum_k \lvert\frac{\omega_k-\hat{\omega}_k}{\omega_k}\rvert$	0.139	0.301
$\frac{1}{K}\sum_k \lvert\frac{\delta_k-\hat{\delta}_k}{\delta_k}\rvert$	0.162	0.319
$\frac{1}{N}\sum_n I(Y_n \neq \hat{Y}_n)$	0.096	0.138

GATE[2]: This is a Greedy Agglomerative Topic Extraction algorithm. It extracted topics based on a pre-defined topic similarity function, which considered both semantic similarity and mission similarity. Here mission similarity refers to the likelihood that two queries appear in the same mission, while missions are sequences of queries extracted from users' query logs through a mission detector.

5.2 Data Sets

5.2.1 Synthetic Data

The goal of the experiments on synthetic data is to show that our proposed algorithm is able to reconstruct the underlying *search factors* and corresponding topic transition matrices from the observed topic transition examples and users' search behaviors. For given model dimensions (M, N, K, T), we start by randomly drawing the word distribution in T topics based on hyper-parameters α, K *search factor*s $\{\hat{\omega}_k\}_{k=1:K}$ where each $\hat{\omega}$ is a vector of length M, and their corresponding topic transition matrices $\{\hat{\sigma}_k\}_{k=1:K}$ based on hyper-parameters α'. Then based on the generative process of our GHSMM model, we randomly draw the topic and *search factor* of the first query for each user, and draw the topic of the next query based on the topic and *search factor* of the current query. The *search factor* $\hat{Y}_n$ of each query S_n is randomly assigned, and accordingly we generate that query's search behavior d_n, i.e. clicked information associated with the query, which fits the assigned *search factor*. Also we draw the content of each query based on its topic and the topic's corresponding word distribution θ. Note that vectors α and α' are of size V and K respectively, where each element α_v and α'_k is generated in $[0.5\hat{\alpha}, 1.5\hat{\alpha}]$ and $[0.5\hat{\alpha'}, 1.5\hat{\alpha'}]$ respectively before simulation. Our experiments are conducted on synthetic data simulated with the following two settings:

- `Small`: $M = 100$, $N = 1000$, $K = 10$, $T = 20$, $\hat{\alpha} = 0.1$, $\hat{\alpha'} = 0.1$. Simulations were run 100 times using the pre-generated model parameters of *search factors* $\hat{\omega}$ and topic transition matrices $\hat{\sigma}$. We report the average performance over the 100 data sets;
- `Large`: M = 1,000, N = 50,000, $K = 100$, $T = 100$, $\hat{\alpha} = 0.1$, $\hat{\alpha'} = 0.1$. Simulations were run 5 times.

To test the robustness of our method, we add noise to the original synthetic data:

`Behavior Noisy`: Instead of using $\mathbf{d}_n$ to simulate users' search behaviors at the n-th query, we use a noisy value $\mathbf{d}'_n$, which is obtained by adding Gaussian noise on $\mathbf{d}_n$.

5.2.2 Real World Data

We evaluate our method on two real world data sets. The first data set is adapted from the query log of `AOL` search engine [3]. The entire collection consists of 19.4 million search queries from about 650,000 users over a 3-month period. We cleaned the data by removing the duplicated queries which were submitted consecutively within 1 minute. We randomly selected a subset of users who submitted over 1,000 queries during this period, and collected their corresponding search activities, including the anonymized user ID, query string, timestamp, the clicked URL. As a result, we collected 1,786 users with 2.2 million queries, and their activities span from 18 days to 3 months. The second data set is collected from Yahoo search engine, from Jan 2013 to September 2013. Similarly, we cleaned the data and randomly selected a subset of users who submitted over 3,000 queries during this period. As a result, we collected 1,475 users with 1.9 million queries, and their activities span from 54 days to 9 months.

Table 3: Likelihood Comparison in Training and Testing

In the column of Metric, "Training" stands for training likelihood, while "Predictive" stands for predictive likelihood. A higher likelihood means a better performance.

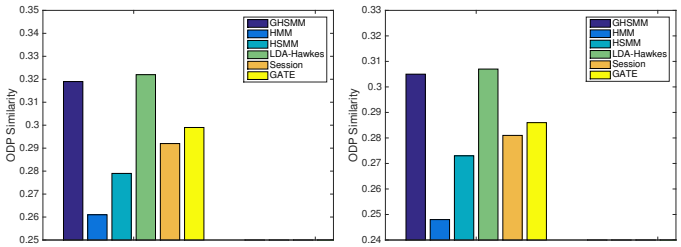
Data set	Metric	GHSMM	HMM	HSMM
Small Synthetic	Training	-107.38	-186.61	-138.49
	Predictive	-143.42	-238.14	-180.31
Small Synthetic with Behavior Noisy	Training	-116.97	-227.31	-154.09
	Predictive	-158.94	-276.28	-201.25
Large Synthetic	Training	-168.75	-261.53	-207.34
	Predictive	-198.49	-313.77	-248.69
Large Synthetic with Behavior Noisy	Training	-181.57	-303.72	-231.28
	Predictive	-221.98	-372.25	-271.04
AOL	Training	-271.35	-463.28	-332.06
	Predictive	-368.92	-580.74	-419.19
Yahoo	Training	-309.14	-531.87	-353.90
	Predictive	-404.08	-650.83	-459.05

5.3 Evaluations

Inference and Estimation. Table 2 evaluates the accuracy of our proposed variation inference algorithm in parameter estimation and *search factor* membership inference under the GHSMM model on synthetic data. Notice that we infer $\hat{\delta}$ and $\hat{Y}$ based on the corresponding variational parameters ρ and Φ, respectively. From the table, we find that our GHSMM model can not only recover the parameters of *search factor*s ω and topic transition matrices δ very well, but also recover the hyper-parameters α and α' very well. Meanwhile, we find that GHSMM can accurately predict the membership of *search factor* for each query.

Model Fitness. Table 3 compares the fitness of the proposed GHSMM model with Markov-based models on real world data sets. For each model, we show its log probability on the training data, and log predictive likelihood on queries falling in the final 10% of the total number of queries in the entire sequences. To avoid overfitting issues, we adopt the cross validation strategy, and select the optimal number of search factors K and topic number T. Basically higher likelihood means better. From Table 3, we find that GHSMM fits real world data sets better than alternative Markov models. And the differences between the proposed model and those alternative models are statistically significant. HSMM performs better than HMM, which implies that real world query logs do embody multiple topic transition rules rather than a single one. The experiments on synthetic data sets with additional behavior noise show that even though the predictability of GHSMM degrade when noise presents, it is robust enough to outperform HMM and HSMM.

Query Clustering. As the topic memberships of queries play an important role in not only identifying the search tasks in query logs, but also learning typical topic transition tracks, we find it necessary



(a) AOL (b) Yahoo

Figure 3: Query Labeling on Real World Data Sets. The Y-axes of figures are measured by ODP Similarity.

to assess the quality of query clustering of the proposed model. In this series of experiments, we evaluate the quality of obtained query clusters/topics, which depends on their purity, or semantic coherence. Since no ground truth about the correct composition of a topic is available, we assess purity by the average similarity of each pair of queries within the same topic as:

$$\text{Purity} = \frac{1}{K}\sum_{k}\frac{\sum_{q_i,q_j\in t_k}\text{Sim}(q_i,q_j)}{N_k(N_k-1)/2} * 100\%,$$

where N_k is the number of queries in topic k.

To evaluate the similarity between queries, we employ the Open Directory Project (ODP)[3] directory, which has been widely used for evaluating the similarity between two queries automatically [5]. The ODP, also known as DMOZ, is a human-edited directory of more than 4 million URLs. These URLs belong to over 590,000 categories organized in a tree-structured taxonomy where more general topics are located at higher levels. For instance, the URL {tech.groups.yahoo.com/group/amrc-l/} belongs to Top/Arts/Animation/Anime/Clubs_and_Organizations, while the URL {http://valleyofazure.tripod.com/} belongs to another directory Top/Arts/Animation/Anime/Characters.

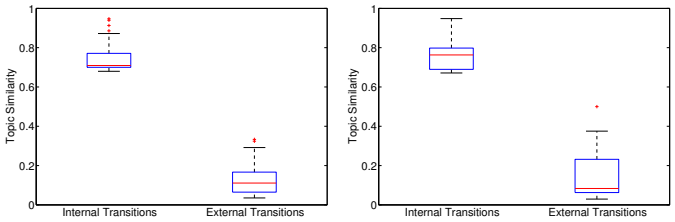
To measure the similarity between categories, we use a notion of similarity between the corresponding categories provided by ODP. In particular, we measure the similarity between category C_i of query q_i and category C_j of query q_j as the length of their longest common prefix $P(C_i, C_j)$ divided by the length of the longest path among those of C_i and C_j. More precisely, we define this similarity as: **ODP Similarity**

$$\text{Sim}(q_i,q_j) = \frac{|P(C_i,C_j)|}{\max(|C_i|,|C_j|)},$$

where $|C|$ denotes the length of a path. For instance, the similarity between the two queries above is 3/5 since they share the path "Top/Arts/Animation" and the longest one is made of five directories. We evaluate the similarity between two queries by measuring the similarity between the most similar categories of the two queries, among the top 5 answers provided by ODP.

Figure 3 compares the performance of our GHSMM model with alternative Markov models, and several state-of-the-art query clustering approaches. GHSMM and LDA-Hawkes can better categorize unlabeled queries than all the other methods. HSMM perform better than HMM. Thus, we conclude that multiple topic transition rules lie under real world query logs instead of a single fixed one.

Quantitative Analysis of the Inferred Topics. In addition to **query clustering**, we design the following experiment to analyze how well our inferred topics from real world query log match the



(a) AOL (b) Yahoo

Figure 4: Quantitative Analysis of the Inferred Topics.

real-world categories from another aspect. For the transitions between each topic-pair (including self-transition), we calculate their average similarity as

$$\text{Sim}(l,l') = \frac{\sum_{i,j:l_i=l\&l_j=l'} I(C_i == C_j)}{\sum_{i,j:l_i=l\&l_j=l'} 1}$$

where C_i and C_j is the corresponding category of query q_i and q_j.

Figure 4 presents the statistical visualization (Box plot) of $\text{Sim}(l,l')$ of both *internal transitions*: self-transitions, and *external transitions*: transitions between different topics. As we can see from Figure 4, the queries assigned to the same topic have much higher similarity than those assigned to different topic. This implies that the queries with the same topic are much more likely to belong to the same real world topic category, while queries assigned with different topics are more likely to belong to different real world topic categories.

Search Task Identification. To justify the effectiveness of the proposed model in identifying search tasks in query logs, we employ a public AOL data subset[4] provided by [23]. Manual annotation was done on 1424 queries and results in 554 annotated search tasks with 2.57 queries per task. This subset contains 13 users with around 110 queries per user. We also recruited eight search editors to annotate the associated task for each query in a subset of query sequences from the Yahoo data. The subset was extracted by randomly sampling 100 users. The average number of queries per user is around 50 and the number of annotated tasks is 1150 in this subset.

We measure the performance by a widely used evaluation metric, F_1 *score*

$$F_1 = \frac{2 * p_{pair} * r_{pair}}{p_{pair} + r_{pair}},$$

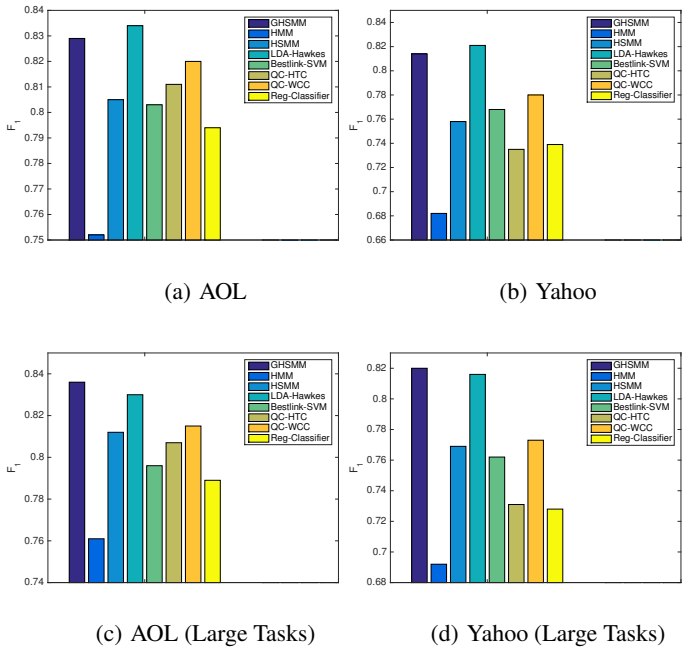
where p_{pair} denotes the percentage of query-pairs in our predicted search tasks that also appear in the same ground-truth task, while r_{pair} denotes the percentage of query-pairs in the ground-truth tasks that also appear in the same predicted task.

The annotations in the above two data sets do not distinguish search goals and search missions, while one of the advantage of the proposed method is that it can merge the queries of closely related topics to large search tasks (i.e. search missions). To justify the effectiveness of the proposed model on identifying large search tasks, we filtered the task annotations such that each remaining task contains more than 10 queries. This results in a much smaller number of tasks in both data sets. Precision and recall are calculated based on the filtered annotations which we denote as "AOL (Large Tasks)" and "Yahoo (Large Tasks)" in Figure 5.

In Figure 5, we compare the proposed model with alternative probabilistic models and state-of-the-art search task identification approaches by F_1 score. From Figure 5, we find that GHSMM

[3] http://www.dmoz.org/

[4] http://miles.isti.cnr.it/~tolomei/?page_id=36



(a) AOL (b) Yahoo

(c) AOL (Large Tasks) (d) Yahoo (Large Tasks)

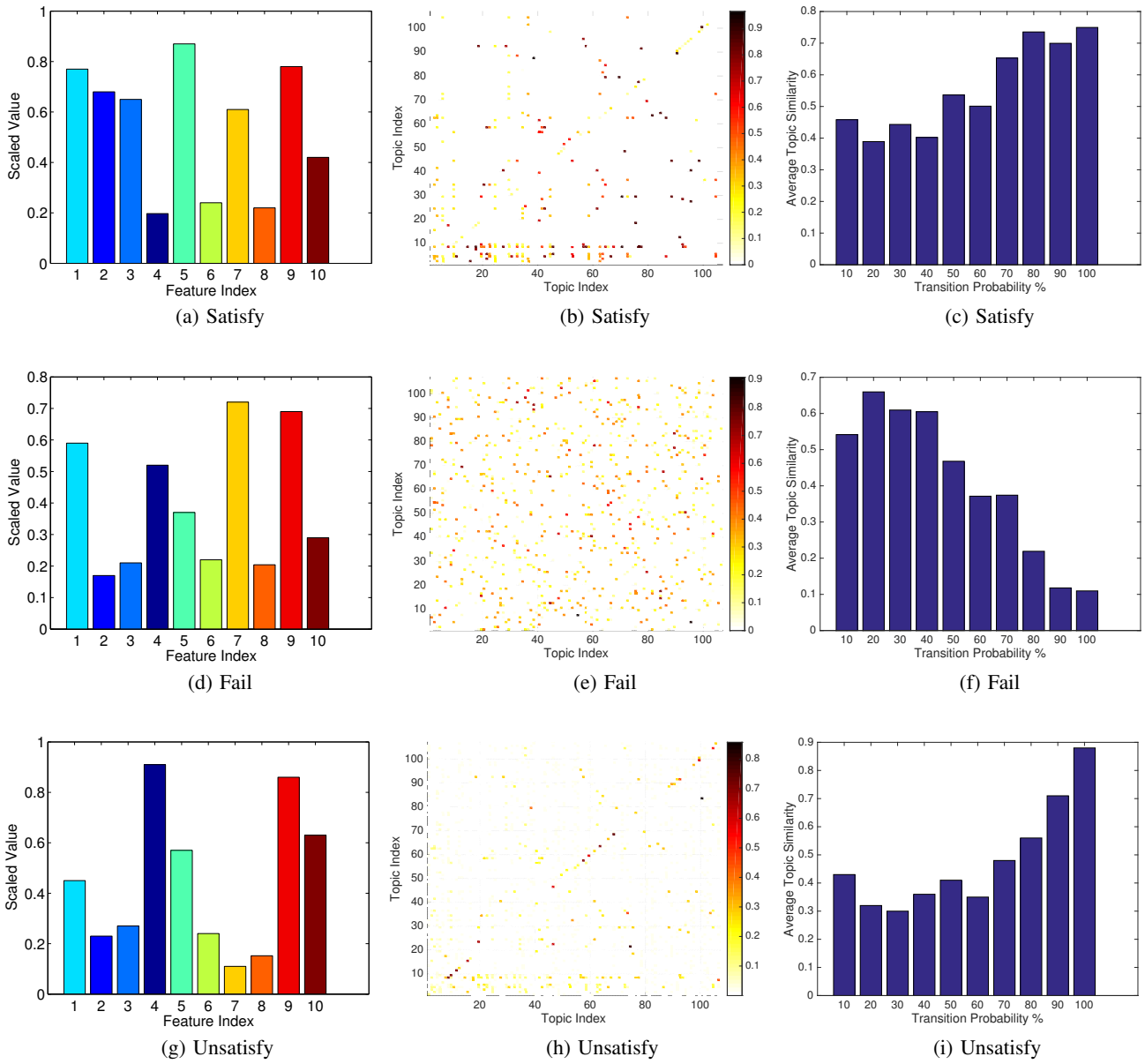
Figure 5: Performance Comparison of Search Task Identification Measured by F_1 Score.

achieves comparable performance with LDA-Hawkes, while performs better than the rest approaches. Moreover, on the selected annotated data subset, GHSMM performs better than all compared methods, which illustrate the advantage of the proposed model in identifying large-scale search tasks. HMM performs the worst, which illustrate that users' choices of query submission depend not only on the previous query, but also a group of *search factor*s like their satisfactory of the returned results and their search habits. Moreover, GHSMM's advantage over HSMM illustrates the importance of various *search factor*s in the influence of user's query submission choice other than temporal information only. The advantage of GHSMM over QC-HTC and QC-WCC demonstrates that appropriate usage of users' search behaviors observed in query logs can even better reflect the semantic relationship between queries, rather than exploiting it in some collaborative knowledge. Moreover, the performance improvement of GHSMM compared with Reg-Classifier illustrates the effectiveness of our designed behavioral features. The advantage of the performance of GHSMM over most baselines on Yahoo query log is greater than that on AOL. One possible reason is that the average length of search tasks in Yahoo is larger than that in AOL, which favorites the strength of the proposed model.

Relationship among Topic Transition, *Search Factor*, and Search Behavior. In the following, we explain how our utilized *search behaviors* are clustered by the proposed model into appropriate *search factors*, which are capable of distinguishing different topic transition rules. Based on the *search factors* and the corresponding topic transition matrices learned by the proposed GHSMM model from a real world data set, we analyze the relationship among topic transition, *search factor*, and search behavior. The goal of the this analysis is to show that our learned *search factors* do have intuitive explanations, based upon their own compositions of search behavior features. We also show that the topic transition rules associated with distinct *search factors* differ in various aspects.

Figure 6 shows several detected *search factors*, and the corresponding topic transition matrices learned by GHSMM from the Yahoo data set. We intuitively name each *search factor* based on the values of features of search behaviors. Notice that to distinguish instant features and accumulated features based on similar search actions, we use "Instant: *" to denote instant features, and "Accumulated: *" to denote accumulated features. Moreover, to facilitate the presentation of learned *search factors*, we scale the value of each feature to the range of $[0, 1]$. We name the *search factor* presented in Figure 6(a) as "Satisfy", since the values in the dimensions of "Instant: Click Number" and "Instant: Time Duration" are significantly higher than those in the dimensions of "Accumulated: Click number" and "Accumulated: Time Duration", while the average click position of the current query is much higher than those of the previous queries in the same search task. The *search factor* shown as in Figure 6(d) may represent "Fail". The value in the dimension of "Instant: Click Number" is very low, while a great amount of time has already been spent on the current search task. The *search factor* presented in Figure 6(g) is likely to imply the user status that he/she is not satisfied with the existing results, and is willing to continue search the same task to satisfy his/her information need. We can notice that the values in the dimensions of "Instant: Time Duration" and "Instant: Click Number" are relatively low, while the amount of time the user has spent on the current search task is still small. Meanwhile, we notice that the topic transition matrices associated with different *search factors* significantly differ from each other. Users with the *search factor* "Satisfy" are likely to continue searching similar topics, as a large proportion of associated topic transition rules with high probabilities are between similar topics. Compared with "Satisfy", users with the *search factor* "Fail" like to search relatively dissimilar topics in the next, since they tend to loss interest and feel tired in the current topic and may want to search a completely new topic for refresh. Users with the *search factor* "Unsatisfy" have a large chance to continue search the same topic and similar topics in the next. Such phenomenon illustrates that multiple distinct topic transition rules do exist in real world query logs, and those transition rules can be distinguished by diverse *search factors*. Also this series of experiments demonstrate that the proposed GHSMM model is capable of detecting those various *search factors* with distinct topic transition matrices, and the learned *search factors* can be powerful signals for the inference of a user's current status in conducting a search task.

Case Study of Identified Search Tasks. In this part, we show a few search task examples identified by GHSMM in Yahoo query log, in order to illustrate effectiveness of the proposed model in recognizing and distinguishing search goals and missions among identified search tasks. From Figure 7, we can find that the proposed model successfully detected related topics that are likely to serve the same information need. Although the word distribution in the topics "travel" and "job market" and very different, the transition probability between those two topics under certain rules are large, which can be learned since those two topics occurs a lot in the query sequences of some users. Similar transitions can be detected between the topic "job market" and "insurance", under the circumstance that the user is planning for travel. In the presented case, separate search goals "travel", "job market", and "insurance" together form the search mission "travel plan", where "travel" is mainly about travel destination information, "job market" looks for the information of travel associated work opportunities, and "insurance" is for safety issues during travel. Those above three goals serve the same information need when the user is working on travel plans. On the other hands, the transition probability between the topic "insurance" and "electronics" under the selected transition



(a) Satisfy (b) Satisfy (c) Satisfy

(d) Fail (e) Fail (f) Fail

(g) Unsatisfy (h) Unsatisfy (i) Unsatisfy

Figure 6: Relationship among Topic Transition, *Search Factor*, and Search Behavior. The left column presents *search factors*. Indices of selected features in *search factors* including not only instant features: 1-'Time Duration', 2-'Dwell Time', 3-'Click Number', 4-'Click Position', 5-'Scanned Pages', but also accumulated features: 6-'CNPQ', 7-'Query Number', 8-'Click Number', 9-'Click Position', 10-'Time Duration'. The middle column presents the corresponding topic transition matrices. Both X and Y axes are topic indices. The right column presents the average topic similarity of topic transition rules with different probabilities. The bar of $c\%$ shows the average similarity of topic transition rules with probabilities in the range of [(c-10)%, c%]

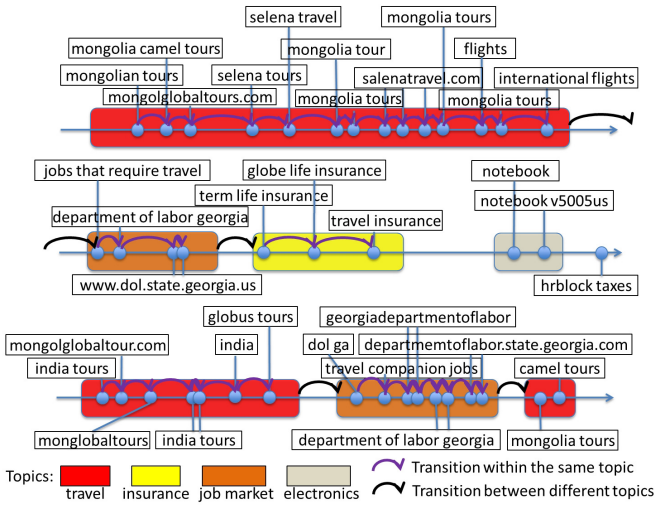


Figure 7: Case Study: Purple arrow line denotes that between two consecutive queries, the transition happens within the same topic, black arrow line denotes the transition between different topics. Search tasks are formed by sequences of queries linked by arrow lines. Rounded rectangle denotes the identified search tasks, rectangle denotes detected topics.

rule is very small, the reason may be that users rarely search them in sequence. Thus we may conclude that search behavior based topic transition learning can be very effective in clarifying search task hierarchy, which can benefit search task identification.

6. RELATED WORK

There are three research areas of related work: 1) search task and session identification, 2) modeling topic transition in query sequences, and 3) analyzing search behavior in the query logs.

There has been a large body of work focused on the problem of identifying search tasks or sessions from sequences of queries. Many of early works use the idea of a "timeout" cutoff between queries, where two consecutive queries are considered as two different sessions or tasks if the time interval between them exceeds a certain threshold [13, 14, 22]. Often a 30-minute timeout is used to segment sessions [9, 22, 28]. Beyond that, there have been attempts to extract search tasks [26, 18, 23, 19, 1, 28] from query sequences based on classification and clustering methods. Jones and Klinkner [18] proposed to learn a binary classifier to detect whether two queries belong to the same task or not, which organized and segmented query sequences into hierarchical units. Kotov et al. [19] and Agichtein et al. [1] studied the problem of cross-session task extraction via binary same-task classification, and found different types of tasks demonstrate different life spans. Cao et al. [9] proposed a clustering algorithm for summarizing queries into concepts throughout a click-through bipartite graph built from a search log. In addition, Wang et al. [28] proposed a semi-supervised clustering method for identifying cross-session tasks. Li et al. [20] casted search task identification into the problem of identify semantic influence in observed query sequences, and proposed a probabilistic model by combining LDA model with Hawkes processes to address the problem using both temporal and textual information. Different from these existing methods, this paper studies query sequences progress through search tasks from the perspective of search topic transitions. The proposed unsupervised model identifies search tasks via both topic membership and topic transition probabilities. Moreover, the proposed method is able to distinguish whether the queries belong to the same search goal or a broader search mission.

Topic transition in query logs has been extensively studied to understand web users' search intents. A number of works [25, 24] have been proposed to learn topic transition rules underlying observed query sequences. For instance, the method proposed in [34] modeled the task of finding the topic transition probabilities as a multiple output linear regression problem. Other probabilistic graphical models, as in [17], attempted to model time-varying dependency in topic transition. One category of probabilistic graphical models that naturally learn topic transition rules is Markov processes, which is a class of stochastic processes that model state transitions. In modeling topic transitions in observed query sequences, Markov models [21] usually view a state as a sequence of queries that belong to the same topic. However, a normal hidden Markov model determines the next state based on the current state only. Hidden semi-Markov models [6, 33], went a further step by assuming the state transition probability is jointly determined by the present state and its duration. However, those models only take into account one search behavior: time duration of the present state, thus the transition probabilities are difficult to directly estimate due to the huge space of time duration. Our model attempts to fully utilize observed search behaviors on each query by assuming that the state change depends not only on the current state, but also on users' search behaviors on that state.

There have been several studies of web search behavior [11, 12, 32, 31] and its influence on search applications such as predicting next topics [29], query suggestion [9, 15] and personalized search [30]. Some of the search behavior features are also investigated in the previous works [29, 9, 30]. Our paper is the first one to model how these web search behaviors have an influence on topic transitions for search task identification and search intent understanding.

7. CONCLUSION AND FUTURE WORK

In this paper, we have presented a novel method of search task identification based on a generative model for behavior driven search topic transition. The underlying assumption is that several latent *search factors* exist in query logs, each associated with a distinct topic transition rule, and these *search factors* can be implicated by users' search behaviors. Given observed query sequences and search behaviors, we proposed a variational inference algorithm to simultaneously estimate the topic membership of each query as well as those remarkable *search factors* and the corresponding topic transition matrices. Experiments on both synthetic and real world data demonstrate that the proposed method better models query log data compared with alternative Markov models and is comparable with the state-of-the-art method on real-world data sets. The case study on a real-world data set shows that the new method is able to identify search task of different scales and also detects interesting latent search factors in search logs.

In future work, we plan to develop more advanced models which explore the dependency of search behaviors within a user's next query on both the topic and search behaviors associated with the current query. Moreover, it would be interesting to consider additional search behaviors, e.g., contents of clicked URLs, into this framework, and investigate the performance of GSHMM model in other domains.

8. ACKNOWLEDGMENT

This work is supported in part by NSF grant IIS-1116886.

9. REFERENCES

[1] E. Agichtein, R. W. White, S. T. Dumais, and P. N. Bennet. Search, interrupted: understanding and predicting search task continuation. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 315–324. ACM, 2012.

[2] L. M. Aiello, D. Donato, U. Ozertem, and F. Menczer. Behavior-driven clustering of queries into topics. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, CIKM '11, pages 1373–1382, New York, NY, USA, 2011. ACM.

[3] AOL. http://gregsadetsky.com/aol-data/.

[4] A. H. Awadallah, R. W. White, P. Pantel, S. T. Dumais, and Y.-M. Wang. Supporting complex search tasks. In *Proceedings of the ACM International Conference on Information and Knowledge Management, ACM, New York, NY*, pages 1–10, 2014.

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

[6] V. Barbu and N. Limnios. *Semi-Markov Chains and Hidden Semi-Markov Models Toward Applications: Their Use in Reliability and DNA Analysis*. Springer Publishing Company, Incorporated, 1 edition, 2008.

[7] D. Blei and M. Jordan. Variational inference for dirichlet process mixtures. In *Bayesian Analysis*, volume 1, pages 121–144, 2005.

[8] A. Broder. A taxonomy of web search. In *ACM Sigir forum*, volume 36, pages 3–10. ACM, 2002.

[9] H. Cao, D. Jiang, J. Pei, Q. He, Z. Liao, E. Chen, and H. Li. Context-aware query suggestion by mining click-through and session data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 875–883. ACM, 2008.

[10] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.

[11] N. Ford, D. Miller, and N. Moss. The role of individual differences in internet searching: An empirical study. *Journal of the American Society for Information Science and technology*, 52(12):1049–1066, 2001.

[12] N. Ford, D. Miller, and N. Moss. Web search strategies and human individual differences: Cognitive and demographic factors, internet attitudes, and approaches. *Journal of the american society for information science and technology*, 56(7):741–756, 2005.

[13] D. He and A. Göker. Detecting session boundaries from web user logs. In *Proceedings of the BCS-IRSG 22nd annual colloquium on information retrieval research*, pages 57–66, 2000.

[14] D. He, A. Göker, and D. J. Harper. Combining evidence for automatic web session identification. *Information Processing & Management*, 38(5):727–742, 2002.

[15] W. Hua, Y. Song, H. Wang, and X. Zhou. Identifying users' topical tasks in web search. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 93–102. ACM, 2013.

[16] L. Jie, S. Lamkhede, R. Sapra, E. Hsu, H. Song, and Y. Chang. A unified search federation system based on online user feedback. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1195–1203. ACM, 2013.

[17] F. Johansson, T. Färdig, V. Jethava, and S. Marinov. Intent-aware temporal query modeling for keyword suggestion. In *Proceedings of the 5th Ph.D. Workshop on Information and Knowledge*, PIKM '12, pages 83–86, New York, NY, USA, 2012. ACM.

[18] R. Jones and K. L. Klinkner. Beyond the session timeout: automatic hierarchical segmentation of search topics in query logs. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 699–708. ACM, 2008.

[19] A. Kotov, P. N. Bennett, R. W. White, S. T. Dumais, and J. Teevan. Modeling and analysis of cross-session search tasks. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 5–14. ACM, 2011.

[20] L. Li, H. Deng, A. Dong, Y. Chang, and H. Zha. Identifying and labeling search tasks via query-based hawkes processes. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 731–740. ACM, 2014.

[21] L. Li, H. Deng, A. Dong, Y. Chang, H. Zha, and R. Baeza-Yates. Analyzing user's sequential behavior in query auto-completion via markov processes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, pages 123–132, New York, NY, USA, 2015. ACM.

[22] Z. Liao, Y. Song, L.-w. He, and Y. Huang. Evaluating the effectiveness of search task trails. In *Proceedings of the 21st international conference on World Wide Web*, pages 489–498. ACM, 2012.

[23] C. Lucchese, S. Orlando, R. Perego, F. Silvestri, and G. Tolomei. Identifying task-based sessions in search engine query logs. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 277–286. ACM, 2011.

[24] S. Ozmutlua and C. G. Cosara. Defining topic boundaries in search engine transaction logs using genetic algorithms. In *Applied Artificial Intelligence*, volume 23, pages 910–931, 2009.

[25] X. Shen, S. Dumais, and E. Horvitz. Analysis of topic dynamics in web search. In *Special Interest Tracks and Posters of the 14th International Conference on World Wide Web*, WWW '05, pages 1102–1103, New York, NY, USA, 2005. ACM.

[26] A. Spink, S. Koshman, M. Park, C. Field, and B. J. Jansen. Multitasking web search on vivisimo. com. In *Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on*, volume 2, pages 486–490. IEEE, 2005.

[27] A. Spink, M. Park, B. J. Jansen, and J. Pedersen. Multitasking during web search sessions. *Information Processing & Management*, 42(1):264–275, 2006.

[28] H. Wang, Y. Song, M.-W. Chang, X. He, R. W. White, and W. Chu. Learning to extract cross-session search tasks. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1353–1364. International World Wide Web Conferences Steering Committee, 2013.

[29] R. W. White, P. N. Bennett, and S. T. Dumais. Predicting short-term interests using activity-based search context. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, pages 1009–1018, New York, NY, USA, 2010. ACM.

[30] R. W. White, W. Chu, A. Hassan, X. He, Y. Song, and H. Wang. Enhancing personalized search by mining and modeling task behavior. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1411–1420. International World Wide Web Conferences Steering Committee, 2013.

[31] R. W. White, S. T. Dumais, and J. Teevan. Characterizing the influence of domain expertise on web search behavior. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 132–141. ACM, 2009.

[32] B. M. Wildemuth. The effects of domain knowledge on search tactic formulation. *Journal of the american society for information science and technology*, 55(3):246–258, 2004.

[33] S.-Z. Yu. Hidden semi-markov models. *Artificial Intelligence*, 174(2):215–243, 2009.

[34] X. Zhang and P. Mitra. Learning topical transition probabilities in click through data with regression models. In *Procceedings of the 13th International Workshop on the Web and Databases*, WebDB '10, pages 11:1–11:6, New York, NY, USA, 2010. ACM.

[35] Z. Zhang and O. Nasraoui. Mining search engine query logs for query recommendation. In *Proceedings of the 15th international conference on World Wide Web*, WWW '06, pages 1039–1040, New York, NY, USA, 2006. ACM.