

Sparse Hidden-Dynamics Conditional Random Fields for User Intent Understanding

Yelong Shen¹Lei Ji²¹Beihang University, Zhichun Road
Beijing, China

shengyelong@gmail.com

Jun Yan²Ning Liu²²Microsoft Research Asia Sigma
Center, Zhichun Road Beijing, China{junyan,leiji, Ningl,
zhengc}@microsoft.comShuicheng Yan³Zheng Chen²³Department of Electrical & Computer
Engineering, National University of
Singapore

shuicheng.yan@gmail.com

ABSTRACT

Understanding user intent from her sequential search behaviors, *i.e.* predicting the intent of each user query in a search session, is crucial for modern Web search engines. However, due to the huge number of user behavior variables and coarse level intent labels defined by human editors, it is very difficult to directly model user behavioral dynamics or user intent dynamics in user search sessions. In this paper, we propose a novel Sparse Hidden-Dynamic Conditional Random Fields (SHDCRF) model for user intent learning from their search sessions. Through incorporating the proposed hidden state variables, SHDCRF aims to learn a substructure, *i.e.* a set of related hidden variables, for each intent label and they are used to model the intermediate dynamics between user intent labels and user behavioral variables. In addition, SHDCRF learns a sparse relation between the hidden variables and intent labels to make the hidden state variables explainable. Extensive experiment results, on real user search sessions from a popular commercial search engine show that the proposed SHDCRF model significantly outperforms in terms of intent prediction results that those classical solutions such as Support Vector Machine (SVM), Conditional Random Field (CRF) and Latnet-Dynamic Conditional Random Field (LDCRF).

Categories and Subject Descriptors

H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval

General Terms: Algorithms, Experimentation

Keywords

user intent, user search session, hidden variable, conditional random field, sparse hidden-dynamic.

1. INTRODUCTION

With the rapid growth of World Wide Web, it has been well recognized that classical relevance based search engines may fail in satisfying users due to the lack of understanding the true intents behind the search queries[25]. For example, when a user submits a

query of “Swimming”, it is unclear whether the user is interested in the sport in water or the famous movie “Swimming” without understanding the user’s search intent.

As shown in recent studies [4-7, 14, 36], user historical search behaviors such as issued queries and clicked URLs [41] could provide rich information for user intent understanding. The continuous behaviors of the same user are often semantically correlated. For example, if the user has issued a query of “Lauren Ambrose” the actor of the movie “Swimming”, right before the query of “Swimming”, it is likely that the user has the intent of “Find Entertainment information” behind the query. Similarly, if the user issues some queries related to sports information before “Swimming”, it is likely that the user has the intent of “Find information on sports”. Therefore, learning user intent based on a sequence of user search behaviors could help search engines to produce better results than treating each user query individually.

Classical machine learning algorithms used for sequential structuralized data analysis focused on two categories of dependencies, namely, user behavioral dependence, such as using sliding window method [45], and class label dependence, such as using Conditional Random Fields (CRFs) and Hidden Markov Model (HMM) [5, 20, 36]. However, both categories are quite limited when applied for user intent understanding. On one hand, there may exist billions of different user search behaviors, if we expect to model the sequential dependencies of these user search behaviors, the huge number of parameters shall require a huge training dataset, which is however of high cost or even impossible in real applications. On the other hand, if we model the dependency over the class labels, the coarse granularity of intent class definitions may lead to serious information loss, which may make the results imprecise. For example, suppose we have two intent labels, “plan a travel” and “find image”, they have different dependency relationships for different query sequences. If the query sequence is “cheap ticket to Seattle” and “Britney Spears”, the class label of the second query does not depend on the class label of the first query. However, if the query sequence is “cheap ticket to Seattle” and then “Seattle image”, the class labels between them may have high dependency. Therefore, the coarse intent labels in previous provides little information for predicting the next user intent label.

The main challenges in modeling sequential user behaviors for intent prediction include:

1. Classical solutions to modeling dependence of user sequential search behaviors are limited, since neither user behaviors nor class labels are suitable for sequential user intent understanding.

This work is accepted when the first author visiting Microsoft Research Asia. Copyright is held by the International World Wide Web Conference Committee (IW3C2). Distribution of these papers is limited to classroom use, and personal use by others.

WWW 2011, March 28 - April 1, 2011, Hyderabad, India.

ACM 978-1-4503-0632-4/11/03.

2. The predefined intent labels could be coarse, which may lead to severe information loss to reflect the true user intents, while each intent generally requires an explainable substructure.

In this work, we propose a novel Sparse Hidden-Dynamics Conditional Random Fields (SHDCRF) model for user intent learning from his/her sequential search behaviors, which are also referred to as search sessions. The proposed SHDCRF model has the following three characters. First, through incorporating hidden-dynamics variables instead of modeling user behavioral dependence and coarse level intent labels dependence, we can capture the true user intent dynamics in the user search session. Second, through using a supervised learning strategy to learn sparse relations between the hidden variables and intent class labels, the hidden state variables become explainable, which could help human editors define new finer scale intent labels. Third, we force the dependency on hidden states variables, which could be efficiently trained and inferred for SHDCRF model.

In terms of computation cost, the model parameters of SHDCRF could be estimated by employing the L-BFGS algorithm [23]. In addition, the SHDCRF model allows natural incorporation of unlabeled data for semi-supervised learning. Experimental results show that the proposed SHDCRF model provides much more accurate intent prediction results in user search sessions than those existing state-of-the-art algorithms such as Support Vector Machines (SVM), Conditional Random Fields (CRFs) and Latent-Dynamic Discriminative Models (LDCRF).

The rest of this paper is organized as follows. In Section 2, we provide a short review of the related work. Then we present the problem formulation for user intent understanding in Section 3. In Section 4, we present the formulation of the SHDCRF model and the training and inference procedures for SHDCRF model. Section 5 demonstrates the detailed experimental results. Finally, we conclude the paper in Section 6.

2. RELATED WORK

There are extensive research efforts dedicated to learning user intents from their online behaviors. Existing methods are generally proposed from two perspectives, namely Non-Context-Aware [34, 35, 37, 22, 19, 11, 3, 9, 15] and Context-Aware [4-7, 14, 36]. Non-Context-Aware methods aim to learn users' intents from their current behaviors such as current search query and clicked URL [1, 2, 13]. Since the single query to be classified is generally short and ambiguous, various techniques have been proposed for feature enrichment, *i.e.* query expansion [34, 35, 37, 22, 19, 11, 3, 9, 15]. While Context-Aware methods assume that the adjacent user behaviors are semantically related and have the same or closely related user intents. For example, several recent studies [4-7, 14, 36] propose to organize user behavioral sequence as temporal time series and learn user intent from them. Generally, it has been well recognized in literature that Context-Aware algorithms generally perform better than Non-Context-Aware approaches.

Cao *et al.* [5, 6] propose the variable length Hidden Markov Model (vHMM) and CRFs model to learn user intent from the user search session. Both the vHMM and CRFs model strongly rely on the Markov property assuming that the next user intent depends only on the current user intent and the next user behavior. As studied in [7], user behavior and user intent at a certain time could have complicated relations and high-order dependency. However,

higher-orders CRFs model could not be computed efficiently since the computational cost increases exponentially with its order. Sutton *et al.* [38] propose a skip-chain CRFs which tries to relax strong Markov assumption by adding long-distance edges. But as studied in [16], it needs a lot of human knowledge to determine which long-distance edge should be added. Sarawagi *et al.* [32] propose a Semi-Markov CRFs model. But it can only deal with segment-based higher order feature.

One common technique to simplify the complex dependency relations is to incorporate a new intermediate layer of hidden state variables. Trinh *et al.* [24] and Peng *et al.* [28] propose Neural conditional random fields (NCRFs) and Conditional Neural Fields (CNFs) respectively, both of which could capture high level features by adding hidden layers. However, their models do not consider the hidden variables dependence. Ariadna *et al.* [30] propose a Hidden-state Conditional Random Fields (HCRF) model for object recognition. The HCRF model has also been successfully employed for phone classification [12], ECG classification [42]. However, these HCRF models can only be applied to label segmented sequence. Sutton *et al.* [39] propose a Dynamic Conditional Random Fields (DCRF) to model sequence data. DCRF model could learn complex interactions and higher-order Markov dependence between user behaviors and user intents. However, learning a DCRF model with hidden variables shall result in a very difficult optimize problem [29].

Morency *et al.* [29] present a Latent-Dynamic Conditional Random Field (LDCRF) model, which also incorporates hidden-dynamic variables for Continuous Gesture Recognition. However, the LDCRF model does not automatically learn the relations between class label and hidden state variables. In fact, in order to make the training and inference phases of the LDCRF model tractable, Morency *et al.* [29] impose a constraint that each class label has a disjoint set of associated hidden state variables. The Sparse Hidden-Dynamic Conditional Random Field (SHDCRF) model overcomes the shortage of LDCRF and does not require the user to manually assign hidden states to each class before the training phase. Instead, it automatically learns to allocate the hidden states to each class optimally. Yu *et al.* [43] present a Deep-Structured Conditional Random Fields model that uses a layer-wise learning strategy to learn discriminative intermediate representations in deep hidden layers. Compared to Deep-Structured CRFs, SHDCRF only incorporates one hidden layer such that the training of the model is more efficient.

Our proposed SHDCRF model is different from various previous studies in many aspects. In contrast to CRFs, which models the intent label dependence, SHDCRF learns the intermediate hidden-dynamics between intent class labels and user behavior variables. In contrast to Latent-Dynamic Conditional Random Fields (LDCRF) [29], SHDCRF proposes to learn the sparse relations between the hidden variables and intent labels instead of specifying by human in LDCRF. In addition, we propose a supervised learning strategy to learn the sparse relations between the hidden variables and intent labels to make the hidden state variables explainable.

3. PROBLEM FORMULATION

In this section, we introduce the notations to be used in the remaining part of this paper.

Mathematically, given a user u , a user behavioral session x is defined as a sequence of observed user behaviors $x_1, x_1 \dots x_T$ where each observed user behavior $x_t (1 \leq t \leq T)$ consists of a query q_t and a set of URLs u_t clicked by the user after issuing query q_t . Let Y be the set of all possible user intent class labels, each user behavior $x_t (1 \leq t \leq T)$ has an intent label $y_t \in Y$. The user intent understanding problem is defined as “*learn a mapping function between a sequence of observations $x = \{x_1, x_2, \dots, x_T\}$ and a sequence of intent labels $y = \{y_1, y_2, \dots, y_T\}$ from the training set.*”

4. SPARSE HIDDEN-DYNAMICS CONDITIONAL RANDOM FIELDS

For clarity and self-containedness, we begin with a brief recap of the standard Conditional Random Fields (CRFs).

CRFs is one of the most commonly used solutions for sequential data classification. In order to estimate the parameters in CRF model, we essentially try to maximize the following objective function:

$$L(\Lambda) = \sum_{(x,y)} \tilde{p}(x,y) \log p_{\Lambda}(y|x) - \frac{\|\Lambda\|^2}{2\sigma^2} \quad (1)$$

where the first term in Eqn.(1) is to maximize the log-likelihood of the training data. Λ is the vector of model parameters, the second term is the regularization term to avoid over fitting, which imposes a zero prior on all the parameter values. σ is used for penalizing large parameter values. (x,y) indicates a sequence of observations with the corresponding sequence of class labels. $\tilde{p}(x,y)$ and $p_{\Lambda}(y|x)$ indicate the empirical distribution and conditional distribution through CRF respectively. Figure 1 shows the graphical structure of CRF model. However, as mentioned above, modeling user intent label dependence is likely to be imprecise due to the weak dependency between the coarse level intent labels of two consecutive behaviors.

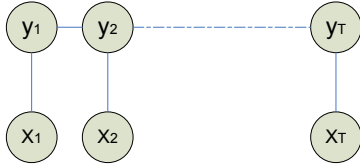


Figure 1. The graphical structure of classical CRF model.

Motivated by the limitation of CRF model, we propose a new probabilistic graphical model named Sparse Hidden-Dynamic Conditional Random Fields (SHDCRF), for sequential data labeling. Figure 2 shows the graphical structure of the SHDCRF model. The SHDCRF model incorporates a vector of hidden variables $h = \{h_1, h_2, \dots, h_T\}$, each h_t takes its value in H where H is the finite set of all possible hidden states. These hidden variables are not observable in the training examples.

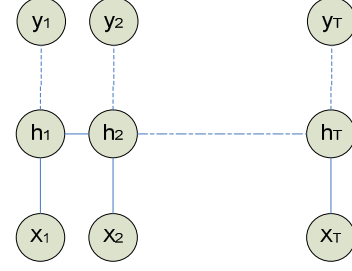


Figure 2. The graphical structure of SHDCRF model.

As shown in Figure 2, we can define the hidden-dynamics conditional probabilistic model as follows:

$$p_{\Lambda}(y|x) = \sum_h p_{\Lambda}(y|h) p_{\Lambda}(h|x) \quad (2)$$

where $h = \{h_1, h_2, \dots, h_T\}$, each h_t is a member of H . Here $p_{\Lambda}(y|h)$ and $p_{\Lambda}(h|x)$ are defined as:

$$p_{\Lambda}(y|h) = \frac{1}{Z_1(h)} \exp \sum_{k=1}^p \beta_k G_k(y, h) \quad (3)$$

$$p_{\Lambda}(h|x) = \frac{1}{Z_2(x)} \exp \sum_{k=1}^q \lambda_k F_k(h, x) \quad (4)$$

where $Z_1(h)$ and $Z_2(h)$ are the partition functions, G_k and F_k are the feature functions for the sequence of intent class labels, hidden variables and user behaviors. In the above equations, we assume that the total number of feature functions is p for the sequence of intent class labels and hidden variables, and q for the sequence of hidden variables and user behaviors. λ_k and β_k are the model parameters corresponding to feature function F_k and G_k respectively. Therefore, model parameters in SHDCRF denote by $\Lambda = \{\lambda_1, \lambda_2, \dots, \lambda_q, \beta_1, \beta_2, \dots, \beta_p\}$ with $p + q$ parameters.

The partition functions $Z_1(h)$ and $Z_2(h)$ are defined as in Eqn. (5) and (6).

$$Z_1(h) = \sum_{y'} \exp \sum_{k=1}^p \beta_k G_k(y', h) \quad (5)$$

$$Z_2(x) = \sum_{h'} \exp \sum_{k=1}^q \lambda_k F_k(h', x) \quad (6)$$

In the above equations, y' is a sequence of intent class labels $y' = \{y'_1, y'_2, \dots, y'_T\}$, h' is a sequence of hidden variables $h' = \{h'_1, h'_2, \dots, h'_T\}$, each y'_t and h'_t is a member of Y and H respectively.

Note that G_k and F_k are the feature functions for the sequences, which could be rewritten as:

$$G_k(y, h) = \sum_{t=1}^T g_k(y_t, h_t) \quad (7)$$

$$F_k(h, x) = \sum_{t=1}^T f_k(h_{t-1}, h_t, x_t) \quad (8)$$

where $g_k(y_t, h_t)$ is the state feature function, and $f_k(h_{t-1}, h_t, x_t)$ contains both the state feature function $s_k(h_t, x_t)$ and transition function $t_k(h_{t-1}, h_t, x_t)$. Here the transition function $t_k(h_{t-1}, h_t, x_t)$ is used to capture the hidden-dynamic in SHDCRF.

However, although incorporating a intermediate layer of hidden state variables could help simplify the complex dependency relations between x and y , it also makes the likelihood function non-convex. In order to avoid bad locally optimal solution and yield sparse relations between the hidden states and the desired outputs, we add an entropy based regularization term in SHDCRF model. Although L_1 regularization term is very popular in most

existing sparse learning literature, such as sparse PCA [46], dictionary learning [47]. However, L_1 regularization term typically makes the objective function non-differentiable. Moreover, L_1 regularization term is not suitable for probability model.

Following the graphical structure as shown in figure 2, we define the objective function in SHDCRF model to learn the model parameters Λ as:

$$L(\Lambda) = \sum_{(x,y)} \tilde{p}(x,y) \log p_{\Lambda}(y|x) - \frac{\|\Lambda\|^2}{2\sigma^2} - \alpha H_{\Lambda}(Y|H) \quad (9)$$

where the first two terms of the formulation are similar with those in Eqn. (1), and the third term aims to minimize the conditional entropy between hidden states variables and class labels. The conditional probability distribution of class labels given hidden states. With the reduction of the conditional entropy, the uncertainty of the class labels given hidden states is also decreasing. As a special case, when the conditional entropy $H_{\Lambda}(Y|H)$ is minimized to be zero, the entropy of conditional probability for intent class labels given each hidden state is also equal to zero. In other words, each hidden state corresponds to only one intent class label, constructing the sub-structure of the intent. Moreover, we ensure the sparseness of the relations between the hidden variables and the intent class labels so as to make the hidden state variables explainable. The conditional entropy $H_{\Lambda}(Y|H)$ is defined as:

$$H_{\Lambda}(Y|H) = -\sum_{h \in H} \sum_{y \in Y} p_{\Lambda}(\dot{y}|\dot{h}) \log p_{\Lambda}(\dot{y}|\dot{h}) \quad (10)$$

Therefore, to estimate the model parameters, our goal is to optimize the objective function and learn the optimal model parameters Λ^* with (11).

$$\Lambda^* = \operatorname{argmax}_{\Lambda} L(\Lambda) \quad (11)$$

4.1 Learning Model Parameters

In this subsection, we aim to learn the model parameters by maximizing the objective function in Eqn. (9), given the training set that consists of n labeled sequences (x^i, y^i) , $i = 1..n$.

Since the SHDCRF model contains the hidden layer, the objective function is not a convex, and thus there generally does not exist closed form solution. In this work, we employed a gradient-based method to search for the locally optimal parameters.

We first give the partial derivatives of the objective function:

$$\begin{aligned} \frac{\partial L(\Lambda)}{\partial \lambda_k} &= \sum_{i=1..n} \tilde{p}(x^i, y^i) \sum_{h'} p_{\Lambda}(h'|x^i, y^i) F_k(x^i, h') \\ &- \sum_{i=1..n} \tilde{p}(x^i) \sum_{h', y'} p_{\Lambda}(h', y'|x^i) F_k(x^i, h') - \frac{\lambda_k}{\sigma^2} \end{aligned} \quad (12)$$

$$\begin{aligned} \frac{\partial L(\Lambda)}{\partial \beta_k} &= \sum_{i=1..n} \tilde{p}(x^i, y^i) \sum_{h'} p_{\Lambda}(h'|x^i, y^i) G_k(y^i, h') \\ &- \sum_{i=1..n} \tilde{p}(x^i) \sum_{h', y'} p_{\Lambda}(h', y'|x^i) G_k(y^i, h') - \frac{\beta_k}{\sigma^2} \\ &+ p(\dot{y}|\dot{h})(\beta_k - \sum_{\tilde{y} \in Y} p(\tilde{y}|\dot{h}) \beta_{K(\tilde{y}, \dot{h})}) \end{aligned}$$

(13)

In these two equations h' and y' are sequence variables, and each element in h' and y' is a member of H and Y respectively. In equation (13), \dot{y} and \dot{h} are also the members of Y and H respectively, the pair (\dot{y}, \dot{h}) is given by parameter β_k under the constraint of feature function $g_k(\dot{y}, \dot{h}) = 1$. (Note that $g_k(\dot{y}, \dot{h})$ is the feature function corresponding with parameter β_k). $K(\dot{y}, \dot{h})$ in Eqn. (13) equals to the unique k^* given \dot{y} and \dot{h} , where k^* satisfied the constraint: $g_{k^*}(\dot{y}, \dot{h}) = 1$.

The partial derivatives for model parameters Λ in Eqn. (12) and (13) cannot be calculated directly, since h', y', x^i and y^i are all sequence variables, Therefore, in order to efficiently calculate Eqn. (12) and (13) using Viterbi path [40], we rewrite Eqn. (12) and (13) by disengaging sequence variables according to the graphical structure shown in Figure 2, with details given in Eqn. (14) and (15).

$$\begin{aligned} \frac{\partial L(\Lambda)}{\partial \lambda_k} &= \sum_{i=1..n} \tilde{p}(x^i, y^i) \sum_{t=1..T^i} p_{\Lambda}(h_{t-1}, h_t | x^i, y^i) f_k(h_{t-1}, h_t, x_t^i) \\ &- \sum_{i=1..n} \tilde{p}(x^i) \sum_{t=1..T^i} p_{\Lambda}(h_{t-1}, h_t | x^i) f_k(h_{t-1}, h_t, x_t^i) - \frac{\lambda_k}{\sigma^2} \end{aligned} \quad (14)$$

$$\begin{aligned} \frac{\partial L(\Lambda)}{\partial \beta_k} &= \sum_{i=1..n} \tilde{p}(x^i, y^i) \sum_{t=1..T^i} p_{\Lambda}(h_t | x^i, y^i) g_k(h_t, y_t^i) \\ &- \sum_{i=1..n} \tilde{p}(x^i) \sum_{t=1..T^i} p_{\Lambda}(h_t | x^i) p_{\Lambda}(h_t, y_t) g_k(h_t, y_t^i) \\ &- \frac{\beta_k}{\sigma^2} + p(\dot{y}|\dot{h})(\beta_k - \sum_{\tilde{y} \in Y} p(\tilde{y}|\dot{h}) \beta_{K(\tilde{y}, \dot{h})}) \end{aligned} \quad (15)$$

In the above equations, $p_{\Lambda}(h_t | x^i, y^i)$, $p_{\Lambda}(h_{t-1}, h_t | x^i, y^i)$, $p_{\Lambda}(h_{t-1}, h_t | x^i)$ and $p_{\Lambda}(h_t | x^i)$ are the four conditional probabilities. All of them can be estimated efficiently by Viterbi path [40].

Take the computation of $p_{\Lambda}(h_t | x^i, y^i)$ and $p_{\Lambda}(h_{t-1}, h_t | x^i, y^i)$ for examples. Each position $t > 1$ at the labeled sequence (x^i, y^i) , we define $|H| \times |H|$ matrix random variable denoted by $M_t(x^i, y^i)$. $M_t(x^i, y^i) = [M_t(h_{t-1}, h_t | x^i, y^i)]$, details as in Eqn. (16).

$$M_t(h_{t-1}, h_t | x^i, y^i) = \exp\{\varphi_t(h_{t-1}, h_t | x^i, y^i)\} \quad (16)$$

$$\begin{aligned} \varphi_t(h_{t-1}, h_t | x^i, y^i) &= \sum_{k=1..q} \lambda_k f_k(h_{t-1}, h_t | x_t^i) + \sum_{k=1..p} \beta_k g_k(h_t | y_t^i) \end{aligned}$$

At position $t = 1$, we define the $|H|$ vector random variable $M_1(x^i, y^i) = [M_1(h_t | x^i, y^i)]$ by Eqn. (17).

$$M_1(h_t | x^i, y^i) = \exp\{\phi(h_t | x^i, y^i)\} \quad (17)$$

$$\phi_t(h_t|x^i, y^i) = \sum_{k=1..q} \lambda_k s_k(h_t|x_t^i) + \sum_{k=1..p} \beta_k g_k(h_t|y_t^i)$$

Then the conditional probability $p_\Lambda(h_t|x^i, y^i)$ and $p_\Lambda(h_{t-1}, h_t|x^i, y^i)$ can be calculated below, *i.e.*

$$p_\Lambda(h_t = a|x^i, y^i) = \frac{[M_1^T \times \prod_{t'=2..t}(M_{t'})]_a \times [\prod_{t'=t..T^i}(M_{t'}) \times I]_a}{M_1^T \times \prod_{t'=2..T^i}(M_{t'}) \times I} \quad (18)$$

$$= \frac{p_\Lambda(h_{t-1} = a, h_t = b|x^i, y^i) [M_1^T \times \prod_{t'=2..t-1}(M_{t'})]_a \times M_t[a, b] \times [\prod_{t'=t..T^i}(M_{t'}) \times I]_b}{M_1^T \times \prod_{t'=2..T^i}(M_{t'}) \times I} \quad (19)$$

where I is a $|H|$ dimensional vector with all the elements equal to 1. The other two conditional probabilities $p_\Lambda(h_{t-1}, h_t|x^i)$ and $p_\Lambda(h_t|x^i)$ have the similar form, we omit the details for saving space.

The gradient of the objective function could be estimated using equations (14) and (15). For each training sample, the space and time complexity of the gradient estimating are both $O(L * N^2)$, where L is the length of the sequence, N is the number of hidden states. In our experiments, we use an L-BFGS [23] method to optimize the objective function. The pseudo-code of the training algorithm is as follows:

Training Algorithm

Input: A training set consisting of n labeled sequences $(x^i, y^i), i = 1..n$.

Output: The optimal model parameters Λ , $\Lambda = \{\lambda_1, \lambda_2, \dots, \lambda_q, \beta_1, \beta_2, \dots, \beta_p\}$.

Algorithms:

Initialize the model parameters Λ randomly.

For each parameter $\lambda_i \in \Lambda$ or $\beta_i \in \Lambda$, estimate the partial derivatives using Eqn. (14) and (15).

Use the L-BFGS method to update Λ .

Repeat steps 2 and 3 until convergence.

Another key challenge facing by many classification models' training process is the scalability issue in dealing with large-scale user behavioral data. Our proposed SHDCRF model is easy to be implemented in a parallel manner under a map reduce framework [17]. In detail, we partition the training data into multiple subsets and distribute each subset to a processor. In the map stage, each processor calculated the gradient of objective function for each training sequence by equation (14) and (15) respectively. In the reduce stage, each processor merges all gradient values and update the model parameters. The two stages are repeated until converge.

4.2 Inference

After the model parameters being estimated, for a new test sequence x , the most probable label sequence y^* could be inferred via maximizing the conditional model,

$$y^* = \operatorname{argmax}_y p_\Lambda(y|x) \quad (20)$$

where the model parameters Λ are learned from the training dataset. Since the conditional probability $p_\Lambda(y|x)$ can be rewritten as in Eqn. (21). By combining the two equations of (20) and (21), we obtained the Eqn. (22).

$$p_\Lambda(y|x) = \sum_h p_\Lambda(y|h) p_\Lambda(h|x) \quad (21)$$

$$y^* = \operatorname{argmax}_y \sum_h p_\Lambda(y|h) p_\Lambda(h|x) \quad (22)$$

Thus, for each position t in the test sequence x , the most probable label y_t^* can be computed as

$$y_t^* = \operatorname{argmax}_{y_t} \sum_{h_t \in H} p_\Lambda(y_t|h_t) p_\Lambda(h_t|x) \quad (23)$$

And, the two terms in equation (23) can be estimated using Eqn. (3) and (4) respectively.

4.3 Semi-Supervised Extension of the SHDCRF Model

One advantage of the proposed SHDCRF model is that it allows the natural incorporation of unlabeled data for training. Recall the first term of the objective function (Eqn. (9)) in SHDCRF model is the log-likelihood as in Eqn. (24).

$$\sum_{x,y} \tilde{p}(x, y) p_\Lambda(y|x) = \sum_{x,y} \tilde{p}(x, y) \log A - \sum_x \tilde{p}(x) \log Z \quad (24)$$

The variables A and Z in the above equation are defined as in Eqn. (25)

$$p_\Lambda(y|x) = \frac{\sum_{h'} \exp(\sum_{k=1}^q \lambda_k F_k(h', x) + \sum_{k=1}^p \beta_k G_k(h', y))}{\sum_{h', y'} \exp(\sum_{k=1}^q \lambda_k F_k(h', x) + \sum_{k=1}^p \beta_k G_k(h', y'))} = \frac{A}{Z} \quad (25)$$

In the classical supervised learning configuration, $\tilde{p}(x)$ and Z are only estimated by using the labeled data. However, the computation of $\tilde{p}(x)$ and Z could be naturally extended to involving unlabeled data. It is only assuming that the expectation of feature F_k and G_k computed based on labeled data are good estimation of the expectation of F_k and G_k computed based on the whole dataset including both unlabeled and labeled data [35]. Through incorporating the unlabeled data to calculate the Eqn. (25), we can learn the model parameters in a semi-supervised manner.

5. EXPERIMENTS

In this section, we use the real user search sessions logged by a popular commercial search engine to empirically validate the effectiveness of the proposed SHDCRF model for user intent classification. We first elaborate on the experiment configurations on dataset, metrics and baselines. Then, we introduce the extensive experimental results along with the algorithmic sensitivity analysis. Finally, some case studies shall be illustrated to show how the SHDCRF model helps real world applications.

5.1 Experiments Configuration

Dataset. We collect a set of 5,629 real user search sessions from a commercial search engine. The average session length is 21.3 clicks. Table 1 shows the statistics of the user behavioral sequence length in the dataset, which contains 56,733 unique queries and 224,893 unique URLs. Five human editors are asked to label each query in all the 5,629 user search sessions using 8 different user intent labels, which are “plan travel”, “communicate with others”, “Shopping”, “Find a fact”, “Entertainment”, “Find Online Services”, “Download file” and “Others”. Table 2 shows the distribution of intent labels in the dataset. Each record in the dataset is a user behavioral sequence, which include both queries and clicked URLs issued through the query. Table 3 uses an example to show the format of the data for experiments. In this work, we use the classical n -gram feature in the Bag of Words model (BOW) [21] to represent each query and extract title for clicked URLs [31]. Then each user behavior is represented as a 16,463-dimensional feature vector. The whole dataset is randomly divided into five folds, each time four of them are used for training and the remaining one for testing. The results reported in the remaining part of this paper are the average of the five runs.

Table 1. The statistics of session length in the dataset.

Session Length	1-8	8-16	16-24	>24
Num	714	1475	1554	1886

Table 2. The distribution of intent labels in the dataset.

Label	Percentage
plan travel	1.31%
communicate	22.76%
Shopping	14.36%
Find fact	13.35%
Entertainment	19.98%
Online Services	9.29%
Download file	3.83%
Others	15.10%

Table 3. An exemplar data record in the dataset.

User Session ID	Query	Clicked URLs	Label
000000000000	Resorts Atlantic City Casino Hotel	http://www.resortsac.com/...	plan travel
000000000000		http://www.resortsac.com/hotel	plan travel
000000000000	hotels in atlantic city	http://www.achotelexperts.com/	plan travel
000000000000		http://tickets.amtrak.com/itd/amtrak	plan travel
...
000000000001	trey songz freestyle	http://www.youtube.com/watch?v=afZPuYujLA0	entertainment
000000000001		http://www.youtube.com/watch?v=fULaXDW6v8U&feature=related	entertainment
...

Evaluation metrics. In this work, we use the classical Precision, Recall and F-Measure to evaluate the effectiveness of

different classification models, where the Precision, Recall and F-Measure are defined as follows:

$$\begin{aligned}
 \text{Precision} &= \sum_{\text{Category } (i)} \frac{\# \text{correctly classified queries}}{\# \text{classified queries}} * \frac{\# \text{category queries}}{\# \text{total queries}} \\
 \text{Recall} &= \frac{\# \text{correctly classified queries}}{\# \text{total queries}} \\
 \text{F-Measure} &= 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}
 \end{aligned}$$

Baselines. Our proposed SHDCRF model is compared with three baseline models, which are Support Vector Machine (SVM) [8], which assumes the queries in a user search session are independent, the classical Conditional Random Field (CRF) [20], which considers the sequential information and the Latent-Dynamic Conditional Random Fields (LDCRF) [29], which assigns a disjoint set of hidden state variables to each class label in advance. In addition, we also compared the proposed SHDCRF model in a semi-supervised problem configuration, which is named as the semi-supervised SHDCRF model (denoted as SHDCRF*) through incorporating 20,000 unlabeled user search sessions. The detailed configuration for each baseline model is:

- **SVM.** In our experiments, we use SVM-light [18] as the toolbox for model training and testing. The SVM model is trained using a linear kernel. The parameter C is determined by cross-validation and the results reported in all experimental results are the parameter configurations for the best results
- **CRF.** The Conditional Random Field model we used for experiments is a single chain structured model [20], and the regularization term in CRFs is determined by cross-validation. In our experiments, we use the CRF-Suit [26] as the tool to obtain the experiment results.
- **LDCRF.** The Latent-Dynamic Conditional Random Fields (LDCRF) [29] was trained by varying the number of hidden states per label, say, from 2 to 6 states per label, and the regularization term in LDCRF was determined by cross-validation to achieve the best performance for comparative study.

5.2 Experiment Results

The experiments are conducted using a 5-fold cross-validation and the experimental results reported in this subsection are the average of five runs. The performances of different algorithms for user intent classification are shown in Table 4, where all the models with hidden variables (LDCRF, SHDCRF, SHDCRF*) are set to have the same number of hidden variables, which is set to be 32 in this group of experiments. The parameter α in the SHDCRF and SHDCRF* models is set to be {0.001, 0.005, 0.01, 0.05, 0.1} respectively and the best performance of different parameter settings are reported to compare with the baselines. As shown in Table 4, our proposed SHDCRF and its semi-supervised configuration, *i.e.* SHDCRF* models, outperform the baselines. The performance of SVM model is the worst among all the baselines since the SVM model does not utilize any context information for classification. In terms of F-measure, our proposed SHDCRF model can relatively improve the performance as high as 12.4% in contrast to the classical CRF model, and 3.5% in contrast to LDCRF model. Through incorporating the unlabeled

data, we could obtain the better empirical data distribution and the proposed SHDCRF* model improves the performance as high as 1.3% in contrast the SHDCRF model

Table 4: Performances of different algorithms for user intent understanding.

Method	Precision	Recall	F-Measure
SVM	0.698 ± 0.011	0.642 ± 0.012	0.669 ± 0.006
CRF	0.721 ± 0.018	0.662 ± 0.019	0.691 ± 0.015
LDCRF	0.798 ± 0.010	0.762 ± 0.008	0.780 ± 0.009
SHDCRF	0.831 ± 0.007	0.782 ± 0.012	0.815 ± 0.010
SHDCRF*	0.849 ± 0.008	0.805 ± 0.006	0.828 ± 0.005

To verify the statistical significance of our experiments, we perform the paired t-test (2-tail) over the F-measure of the experimental result. As shown in Table 5, all the t-test results are less than 0.01, which means the improvements of SHDCRF and SHDCRF* models are statistically significant in contrast to the baselines.

Table 5: Paired t-Test (2-tail) results.

t-Test	SVM	CRF	LDCRF
SHDCRF	6.631E-11	9.851E-07	2.19E-05
SHDCRF*	3.92E-11	6.05E-07	3.11E-06

5.3 Sensitivity Analysis

There are two importance parameters to be analyzed in our proposed model. First, the number of hidden states is an important parameter in both LDCRF and SHDCRF models. In Figure 3, we show the performance of LDCRF, SHDCRF and SHDCRF* models with different number of hidden states, which are 16, 24, 32, 40, 48, respectively. In this figure, the SHDCRF and SHDCRF* models achieve better performance than LDCRF in most cases. The only exception is that when the hidden states number is small, their performances are comparable. This means that with a reasonable large number of hidden variables, our proposed model can consistently outperform the baseline model. From this figure, we can also observe that the effect of this parameter almost converges when we increase the number of hidden variables, *i.e.* the performance of our proposed model gets stable with the increasing of the hidden variable number.

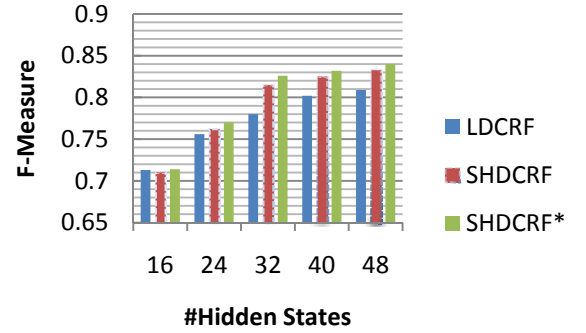


Figure 3. Performance of LDCRF, SHDCRF and SHDCRF* with different number of hidden states.

Another parameter we need to exploit is the parameter α , which is first introduced in Eqn. (9) and used to determinate the strength of the sparse relations between the hidden variables and intent class labels. The experiment results using different α values, $\{0, 0.001, 0.005, 0.01, 0.05, 0.1\}$, for SHDCRF and SHDCRF* models are given in Figure 4 and 5 respectively. From these results, we can observe that the parameter α , without being set to be 0, has very limited impact on the proposed models if the number of hidden states is given. Among the results with slight differences, we assign the parameter value 0.1 or 0.05, which empirically gives the best performance. However, when setting α to be 0, the performance becomes pretty bad. It can be interpreted that the model parameters trapped into the bad local maxima during the learning phrase.

From Figure 4 and 5, it can be concluded that the sparsity condition in SHDCRF is essential for avoid trapping into the bad local maxima and ensuring good experiment results.

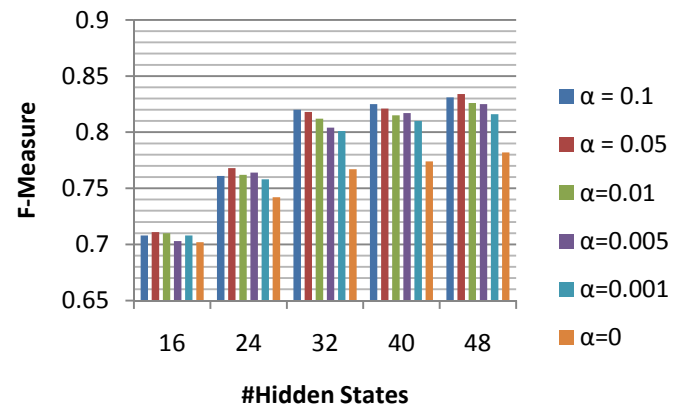


Figure 4. Parameter α sensitivity analysis in SHDCRF model.

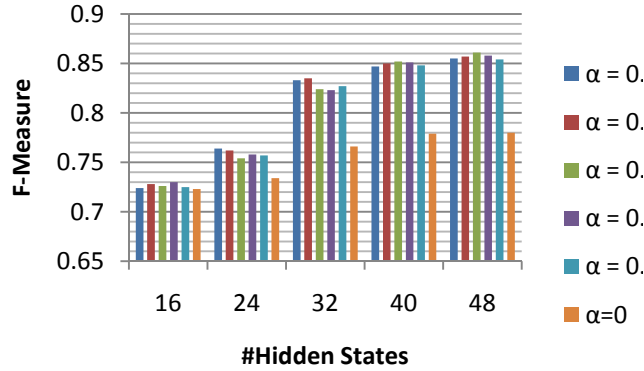


Figure 5. Parameter α sensitivity analysis in SHDCRF* model.

In addition, we analyze the relations between intent class labels and hidden variables with parameter α setting to be 0 and 0.05 respectively, when our SHDCRF model incorporates 24 hidden variables. In Figure 6 and 7, there are two charts both containing 8*24 small cells, which indicates 8 intent class labels and 24 hidden variables. The gray of the cell at i^{st} row j^{st} column reflects the conditional probability of hidden variable h_j given the intent class label y_i , which can be denoted as $p(h_j|y_i)$.

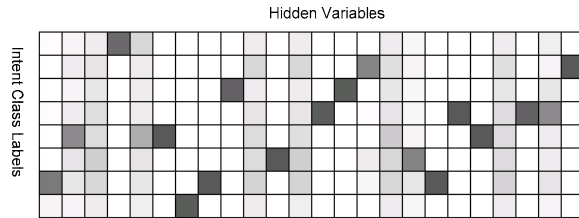


Figure 6. Relations between intent class labels and hidden variables with parameter $\alpha = 0$.

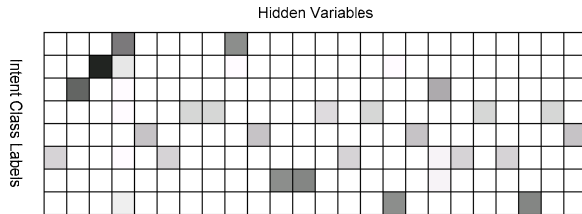


Figure 7. Relations between intent class labels and hidden variables with parameter $\alpha = 0.05$.

In figure 8 and 9, we show the black cells in the two charts if the corresponding conditional probability $p(h_j|y_i) > 0.1$ with parameter α setting to be 0 and 0.05 respectively. It clearly shows that sparsity condition in SHDCRF could generate sparser relations

between intent class labels and hidden variables than without the condition.

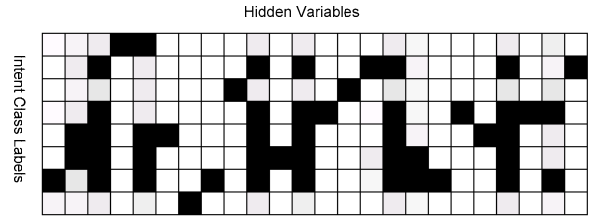


Figure 8. Sparse Relations between intent class labels and hidden variables with parameter $\alpha = 0$.

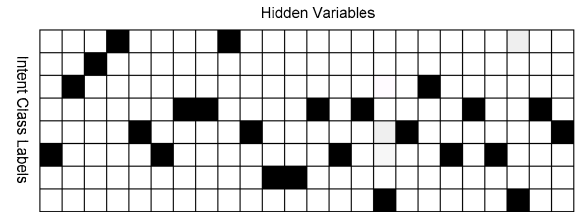


Figure 9. Sparse Relations between intent class labels and hidden variables with parameter $\alpha = 0.05$.

5.4 Case Studies for Understanding the Hidden Variables

One of the major advantages of learning the sparse structure between hidden variables and intent labels is to make the substructure, *i.e.* the hidden variables, explainable. In this subsection, we use the case studies to analyze the hidden state variables in SHDCRF model. Through these case studies, we show that the hidden substructure discovered by SHDCRF model could help us define new finer scale user intent labels.

To simplify the case studies and without loss of generality, we arbitrarily trained the SHDCRF model using 24 hidden variables. The analysis for the hidden states in SHDCRF model is reported in Table 6. In this Table, the column “Label” indicates the user intent labels, which could be “plan travel”, “communicate”, “Shopping”, “Find a fact”, “Entertainment”, “Find Online Services”, “Download file” or “Others”. The column “H” indicates the index of hidden state variables in the model. For each hidden state h_i , we sample some queries that satisfy $h_i = \arg\max_h p_A(h|x)$. Then we give some example queries in the “Example Queries” column of this table. In addition, three human editors are asked to tag these sampled queries with a new label, which is in the column “Tag”. The column “Acc%” shows the accuracy of these sampled queries that can be tagged with the new label. Take the intent label “Plan travel” as an example, in SHDCRF model, it is divided into two sub-structures “Map” and “Rental & book hotel” responded to the two hidden states respectively. As shown in the column “Acc %”, the accuracies of sampled queries which can be tagged with “Map” and “Rental & book hotel” is 85.5% and 86% respectively. As shown in Table 6, we can also find some other interesting sub-structures discovered by SHDCRF model. For instances the intent label “Shopping”, in SHDCRF model, it is divided into “Home shopping”, which indicates buying something for family and home, and “Entertainment shopping”, which indicates that user wants to

buy something for entertainment such as buying CD and ticket for concert etc.

6. CONCLUSION

In this paper, we present a novel Sparse Hidden-Dynamic Conditional Random Fields model for user intent understanding from user search logs. The SHDCRF model aims to learn a substructure for each intent label which is used to model the intermediate dynamics between user intent labels and user

behavioral variables. In addition, we propose to learn sparse relations between the hidden variables and intent labels in SHDCRF to scale up the computation and make the hidden state variables explainable. Extensive experimental results show that the proposed SHDCRF model gives much more accurate intent prediction results in user search sessions than some existing state-of-the-art methods including Support Vector Machines, CRFs, Latent-Dynamic Discriminative Models etc.

Table 6: Sub-structure discovered using SHDCRF.

Label	H	Example Queries	Tag	Acc%	Label	H	Example Queries	Tag	Acc%
Plan Travel	H0	Yahoo maps MapQuest Miami Topographic maps Michigan	Map	85.5	Entertainment	H12	Free Porn Free sex videos riley evans kristina rose karla lane clips	Porn	61.2
	H1	budget car rental hotels in atlantic city Tropicana Casino & Hotel	Rental & book Hotel	86.0		H13	Ragdoll Avalanche 2 ATV Tag Race One Shot One Kill play bejeweled 2 online	Game	67.0
Communicate	H2	Myspace Bebo Facebook	Communi cate	71.2		H14	boxing logo Computer Photo Art hp wallpapers landscaping ideas abstract wallpapers	Image	64.6
Shopping	H3	portable air conditioner Pet Supplies grant hill shoes	Home shopping	66.5	Online Services	H15	Project Engineer Job heico Jobs in carmel Quality Inspector Job	Job	57.4
	H4	Sevendust CD Sevendust Discography courtney love flash concert	Entertain ment- shopping	59.6		H16	tenn vols football WVLT Volunteers Football university tenn vols football	education tasks	68.2
Find a Fact	H5	Hair Salon in Fairfax find people donna edmondson	Find people & place	61.5		H17	State salary results Government Salary Data Sarasota Weather New York, NY detroit news	personal accounts	70.9
	H6	fdr quotes american top 40 the 70s Seth Rogan praxair 2008 annual report	Find Informati on	64.0	Download	H18	Retail jobs in Bolingbrook Yahoo! HotJobs Retail jobs in Bolingbrook Management Opportunities	Job	65.1
	H7	Abt Electronics gonzales county texas pd list dealer	Find company	56.8		H19	Brickell Atrium Condo Miami Florida Condos mortgage calculator	Real estate	65.4
	H8	fernando colunga biography albert pujols nathan myers OBIT	famous people	63.5		H20	Popular Screensavers topgear 1680x1050 lamborghini 1680x1050	Downloa d	71.2
	H9	build a earthquake resist model build a bird's nest mystery dungeon of the sky	specific fact	61.1	Others	H21	Itunes free printable chore list free excel database templates bejeweled 2	Downloa d	73.9
	H10	john mcyntire simulator james a stewart linkedin william love attorney	specific fact	70.9		H22	VOA News stupcat 2009 air jamaica	Others	54.5
Entertainment	H11	jeark by solder boy Swimming diana ross inundaciones 2012 end of the world	Video & actors	62.3		H23	veronica lara arias apaseo CD arranque xp meselation 2009	Others	65.6

7. REFERENCES

- [1] E. Agichtein, Eric Brill and S. Dumais. Improving Web search ranking by incorporating user behavior information. In SIGIR'06, pp. 19-26.
- [2] E. Agichtein, Eric Brill, S. Dumais and R. Ragno. Learning User Interaction Models for Predicting Web Search Result Preferences, In SIGIR'06, pp. 3-10.
- [3] Z.B. Andrei, M. Fontoura, E. Gabrilovich, A. Joshi, V. Josifovski and T. Zhang. Robust Classification of Rare Queries Using Web Knowledge. In SIGIR'07, pp. 231-238.
- [4] H. Cao, D. Jiang, J. Pei, Q. He and Z. Liao, E. Chen and H. Li. Context-Aware Query Suggestion By Mining Click-Through and session Data. In SIGKDD'08.

- [5] H. Cao, D. Jiang, J. Pei, E. Chen and H. Li. Towards context-aware search by learning a very large variable length hidden markov model from search logs. In WWW'09.
- [6] H. Cao, D.H. Hu, D. Shen, D. Jiang, J.T. Sun, E. Chen and Q. Yang. Context-aware query classification. In SIGIR'09.
- [7] Z. Cheng, B. Gao and T.Y. Liu. Actively predicting diverse search intent from user browsing behaviors. In WWW'10.
- [8] C. Cortes and V. Vapnik. Support-Vector Networks. *Machine Learning*, 1995, Vol. 20, pp.273-297.
- [9] E.R. Daniel and Danny Levinson. Understanding User Goals in Web Search. In WWW'04, pp. 13-19.
- [10] G. David, D. Nichols, M. Brain and O.D. Terry. Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 1992, vol.12, pp. 61–70.
- [11] V. Ganti, A.K. Christian and X. Li. Precomputing search features for fast and accurate query classification. In WSDM'10.
- [12] Gunawardana, M. Milind, A. Alex, and J.C. Platt. Hidden Conditional Random Fields for Phone Classification. *International Conference on Speech Communication and Technology(ICSCCT)*, 2005.
- [13] Hassan, R. Jones, and K.L. Klinkner. Beyond DCG, User Behavior as a Predictor of a Successful Search. In WSDM'10.
- [14] Q. He, D. Jiang, Z. Liao, C.H. Steven, K. Chang, E.P. Lim and H. Li. Web Query Recommendation via Sequential Query Prediction. In ICDE'09.
- [15] J. Hu, G. Wang, F. Lochovsky, J.T. Sun and Z. Chen. Understanding user's query intent with wikipedia. In WWW'09.
- [16] D.H. Hu, D. Shen, J.T. Sun, Q. Yang and Z. Chen. Context-Aware Online Commercial Intention Detection. *LNCS(5829)*, 2009, pp. 135-149.
- [17] D. Jeffrey and S. Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. In *Operating Systems Design and Implementation(OSDI)*, 2004, pp. 137-150.
- [18] T. Joachims. SVM-light Support Vector Machine. <http://svmlight.joachims.org/>
- [19] I.H. Kang, G.C. Kim. Query Type Classification for web document Retrieval. In SIGIR'03.
- [20] J. Lafferty, A. McCallum and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML'01*.
- [21] Lewis and David. Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval. In *ECML'98*, pp.4-15.
- [22] X. Li, Y.Y. W, Alex Acero. Learning Query Intent from Regularized Click Graphs. In SIGIR'08.
- [23] D. C. Liu and J. Nocedal. On the Limited Memory Method for Large Scale Optimization. *Mathematical Programming*, 1989. pp. 503-528.
- [24] S.Martigny and T. Artieres. Neural conditional random fields. In *AISTATS'10*, pp.177-184.
- [25] H. Nguyen. Capturing User Intent For Information Retrieval. In *AAAI'04*.
- [26] N. Okazaki. CRFsuit A fast implementation of Conditional Random Fields. <http://www.chokkan.org/software/crfsuite/>.
- [27] J. Pearl. Probabilistic Reasoning in Intellignet Systems: Networks of Plausible Inference. Morgan Kaufmann, 1998.
- [28] J. Peng, L.F. Bo and J.B. Xu. Conditional Neural Fields. In *NIPS'09*.
- [29] L.M. Philippe, A. Quattoni and T. Darrell. Latent-Dynamic Discriminative Models for Continuous Gesture Recognition. In *CVPR'07*.
- [30] Quattoni, M. Collins, and T. Darrell. Conditional random fields for object recognition. In *NIPS'04*.
- [31] B.Y. Ricardo, C. Hurtado and M. Mendoza. Query Clustering for Boosting Web Page Ranking. *LNCS(3034)*, 2004, pp.164-175.
- [32] S. Sarawagi and William W. Cohen. Semi-Markov conditional random fields for information extraction. In *NIPS'04*.
- [33] K. Seymore, A. McCallum and R. Rosenfeld. Learning Hidden Markov Model Structure for Information Extraction. In *AAAI'99 Workshop on Machine Learning for Information Extraction*, 2009.
- [34] D. Shen, J.T. Sun, Q. Yang, and Z. Chen. Building Bridges for Web Query Classification. In SIGIR'06.
- [35] D. Shen and R. Pan. Query Enrichment for Web-Query Classification. *ACM Transactions on Information System* 2006.
- [36] M.B. Steven, C.J. Eric, F. Ophir , D.L. David, C. Abdur , K. Aleksander. Improving Automatic Query Classification via Semi-Supervised Learning. In *ICDM'05*.
- [37] M.B. Steven and C.J. Eric. Automatic Classification of Web Queries Using Very Large Unlabeled Query Logs. In *TOIS'06*, vol.24, pp.320-352.
- [38] Sutton and A. McCallum. Collective Segmentation and Labeling of Distant Entities in Information Extraction. In *ICML workshop on Statistical Relational Learning*, 2004.
- [39] Sutton, K. Rohanimanesh and A. McCallum. Dynamic Conditional Random Fields: Factorized Probabilistic Models for Labeling and Segmenting Sequence Data. In *ICML'04*.
- [40] Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 2003, Vol.13, pp. 260-269.
- [41] J. Wang, A.de Vries and M. Reinders. A User-Item Relevance Model for Log-Based Collaborative Filtering. *LNCS3936* (January 2006), pp. 37-48.
- [42] S.B. Wang, A. Quattini, L.P. Morency and D. Demirdjian. Hidden Conditional Random Fields for Gesture Recognition. In *CVPR'06*.
- [43] Yu, L. Deng, and S. Wang. Learning in the Deep-Structured Conditional Random Fields. In *NIPS'09*.
- [44] X.H. Zhang. Building Maximum Entropy Text Classifier Using Semi-Supervised Learning. PhD thesis, NUS, 2004.
- [45] Thomas G. Dietterich. Machine Learning for Sequential Data: A Review. *LNCS(2396)*, 2002, pp. 15-30.
- [46] H. Zou, T. Hastie and R. Tibshirani. Sparse Principal Component Analysis. *Journal of Computational and Graphical Statistics*, 2006.
- [47] J. Mairal, F. Bach, J. Ponce and G. Sapiro. Online dictionary learning for sparse coding. In *ICML'09*.