Inferring User Interests From Microblogs

Ceren Budak Anitha Kannan Rakesh Agrawal Jan Pedersen
Microsoft
{cbudak, ankannan, rakesha, jpederse}@microsoft.com

Abstract

We address the problem of inferring users' interests from microblogging sites such as Twitter, based on their utterances and interactions in the social network. Inferring user interests is important for systems such as search and recommendation engines to provide information that is more attuned to the likes of its users. In this paper, we propose a probabilistic generative model of user utterances that encapsulates both user and network information. This model captures the complex interactions between varied interests of the users, his level of activeness in the network, and the information propagation from the neighbors. As exact probabilistic inference in this model is intractable, we propose an online variational inference algorithm that also takes into account evolving social graph, user and his neighbors? interests. We prove the optimality of the online inference with respect to an equivalent batch update. We present experimental results performed on the actual Twitter users, validating our approach. We also present extensive results showing inadequacy of using Mechanical Turk platform for large scale validation.

Introduction

Microblogging sites such as Twitter and Weibo serve as important social platforms for users to express and discuss, in real time, their thoughts, views and ideas on a plethora of subject matters. Hence, these social media or microblog streams can serve as an important source for inferring the interests of its users as projected by their utterances in the social media. This is exactly the focus of this paper: inferring user interests from social media streams by taking into account the utterances of the user, his activeness in the social media platform and the utterances of his friends. Such an inference is valuable for many applications. For instance, an online system such as a search or recommendation engine can provide personalized content that is more attuned to the likes of its users. Similarly, information trends presented to a social media user can better reflect their interests.

There is a rich line of work in interest identification in the context of web search. They make use of user query sessions, their past interactions with the search engine in the form of click-history or browsing history (Bing, Lam, and Wong 2011; Liu and Tang 2011; Michelson and Macskassy 2010; Pennacchiotti and Popescu 2011; Rao et al. 2010; Weng et al. 2010). However, very little work has been done in the context of interest identification from microblogs. Microblogs pose a unique set of challenges that make it difficult to directly borrow techniques from these earlier works. Unlike a user query session, microblog postings are noisy, with each posting possibly about multiple topics, requiring the disentanglement of the postings to the appropriate interest. In addition, in microblogs, users co-exist with each other, often influenced, albeit differently, by their neighbors. In addition, social interactions are dynamic in nature and this dynamics needs to be taken into account.

Due to these inherent challenges, much of the work in user understanding from microblogs has focused on inferring high level demographics such as gender, age and location (Rao et al. 2010; Pennacchiotti and Popescu 2011) or narrowly focused on a single interest such as political affiliation (Pennacchiotti and Popescu 2011; Rao et al. 2010). Often, many of the techniques treat each user independent of their network. A straightforward approach is to use the user provided profile on their account to infer the interests - recent works has shown that such a profile often exhibit ideal self-image (Goffman 1959) and does not necessarily capture their exhibited interests. Of particular relevance and importance is the work of (Michelson and Macskassy 2010). The goal of that work is also to infer all the interests of an user, but based *solely* on the corresponding user utterances without taking into account the effect of social network.

In this paper, we address the aforementioned challenges in a principled manner. We propose a probabilistic generative model of user utterances. In this model, the hidden variables are the interests (to be inferred) of the users which compete to explain the utterances of a user, while simultaneously taking into account the user in the context of the network, his level of activeness in the network and is susceptibility to his neighbors. Probabilistic inference in this model yields user-specific distribution over interests. As user interests change over time, our model learns interests using online learning in which the prior distribution over interests at every time step is updated using a linear function of prior and posterior distributions at the previous time step. Exact inference in our model is intractable and therefore we appeal to variational methods to perform approximate inference (Jaakkola and Jordan 1999). We also theoretically show that the apply-

ing variational methods at each time step in a greedy manner provides an optimal choice of variational parameters.

Our evaluation over one year of Twitter data shows the efficacy of our approach in accurately inferring user interests. Our performance evaluation is based on a user study in which we *directly* ask the users if the inferred interests match with their view of their interests. We also report on our experience of attempting to use Amazon Mechanical Turk for performing validation at scale.

Related Work

Our work lies in the intersection of research in online social networks and behavioral targeting. Here we give an overview of the related literature on these topics.

Online Social Networks: With increasing popularity of online social networks, understanding user characteristics and interests of users has attracted large attention (Bing, Lam, and Wong 2011; Liu and Tang 2011; Michelson and Macskassy 2010; Pennacchiotti and Popescu 2011; Rao et al. 2010; Weng et al. 2010). In (Liu and Tang 2011), the authors study three methods that use social signals in behavioral targeting: classification (supervised), ensemble (supervised), and network propagation (unsupervised). They conclude that the social signals tend to be noisy and do not provide improvements in the case where there is already some information about the user. (Bing, Lam, and Wong 2011) investigates query refinement through social data and shows that use of friends' actions can help with this task. (Weng et al. 2010) focuses on identifying influential users in Twitter on a per-topic basis. Topics are learned using Latent Dirichlet Allocation (LDA). Focus of (Michelson and Macskassy 2010) is on identifying user interests by classifying entities in the tweets. They leverage Wikipedia to disambiguate and categorize the entities and identify static interests while we focus on evolving interests. (Rao et al. 2010) studies the problem of extracting user characteristics such as gender, age, regional origin, and political orientation from a user's tweets. In a similar vein, (Pennacchiotti and Popescu 2011) studies the problem of identifying *user profiles* in Twitter. This work also focuses on understanding small number of user characteristics such as the political affiliation, ethnicity and affinity. Both (Pennacchiotti and Popescu 2011) and (Rao et al. 2010) rely on a large amount of labeled data for learning a classifier for each interest type. Unlike the studies listed above, our approach is completely unsupervised and parallelized through a DryadLinq (Yu et al. 2008) implementation, enabling inference at scale. We also model changing interests and evaluate our solution through a user study.

Behavioral Targeting: Much of the literature in behavioral targeting is focused on understanding web user behavior during a short time window (Chen, Pavlov, and Canny 2009; Hassan, Jones, and Klinkner 2010; Kim and Chan 2003; Sugiyama, Hatano, and Yoshikawa 2004). The work of (Sugiyama, Hatano, and Yoshikawa 2004) focuses on personalizing search by constructing user profiles from a day-long browse history session. (Hassan, Jones, and Klinkner 2010) focuses on identifying user search goal success in search engines to show that models that use user behavior are more predictive of goal success than those using document relevance. (Chen, Pavlov, and Canny 2009) focuses on efficiency of behavioral targeting and proposes a scalable solution using Hadoop MapReduce framework. The approach proposed in (Kim and Chan 2003) is to learn a user interest hierarchy from a set of web pages visited by a user. (Ahmed et al. 2011) infers both long term and short term interests by employing a time-varying LDA to define interests over histories of multiple users. Unlike these works, we study the problem of understanding interests of users from their interactions in a social network.

Methodology

Here we formalize a general model of user interactions in online social networks and introduce our problem statement. We assume that time is divided into fixed time steps and a time interval is denoted as $[t,t+1)$. Users emit zero or more messages in each interval. Each message is simply a string of characters, consisting of phrases. The set of phrases contained in all the messages emitted by a user during a certain time interval constitutes his utterances during this interval.

Each phrase in a user's utterance comes from a universe of phrases, which we assume for notational simplicity to comprise of M phrases known a priori. U_t^v denotes the utterance of user v during time interval $[t,t+1)$ and is a binary vector of size M, with $U_t^v[u] := 1$ if the phrase u is present in at least one of the messages emitted by v during $[t,t+1)$, and 0 otherwise. The entire history of utterances for a particular user is denoted as $\{\mathcal{U}^v\}$.

For our purposes, an interest is simply a literal. The universe of interests I is known a priori, consisting of K literals. We use the notation I_t^v to represent the set of interests of user v at time t and view it as a binary vector of size K, with I_t^v[i] := 1 if v is interested in i at time t and 0 otherwise. We assume that interests are independent and that if a user is interested (uninterested) in i at time t, he remains interested (uninterested) in i during $[t,t+1)$.

The social network at time t, $G_t = (N_t, E_t)$, consists of the set of users N_t and friend relationships E_t between them. If a user $w \in N_t$ is a friend of $v \in N_t$ at time t, then there is as an edge $e_{v,w}$ from v to w in E_t. It is possible that $\exists\, e_{v,w} \in E_t$, but $\nexists\, e_{w,v} \in E_t$. We assume that G_t remains fixed during the time interval $[t,t+1)$, that is, N_t and E_t do not change during $[t,t+1)$. Yet, both sets can evolve over time, across different time steps.

Problem Statement: We are given a social network that evolves over time $\{G_t\}$ and the utterances $\{\mathcal{U}^v\}$ of its users $\{v\}$. For each user v, the prior distribution over the interests at time 0 is known and given by $p(I_0^v) = \prod_{l=1}^{K} p(I_0^v[l])$. The goal is to infer, for each user, at any time period, the probability of the user being interested in each of the interests. In particular, we would like to compute $p(I_t^v[l] = 1 | G_{t-1}, \{\mathcal{U}\})$ for $1 \le l \le K$ and for all users $\{v\}$ at every time step $1 \le t$.

Model

In our model of social network, people communicate through utterances consisting of phrases. The set of phrases a user includes in an utterance depends on his latent interests. The same interest can cause different phrases to be uttered by different users or by the same user in different utterances. Similarly, different interests might result in utterance of some of the same phrases.

Every user does not participate equally in the social network. Various users make different number of utterances, depending upon their level of activeness. People have varying number of friends. What a user utters gets influenced by what his friends utter and how susceptible the user is to the influence of his friends.

Our procedure for inferring a user's latent interests is designed to capture the intuition that if the probability of a phrase appearing in an utterance of a particular user is low but the utterance does include the phrase, then it is highly probable that the user's interests include the likely interest corresponding to this phrase.

Modeling a user's utterance independent of his activity level and friends: A user's utterance can be attributed to multiple interests. The first model depicted in Figure 1(a) models an utterance observation as an outcome of one or more interests. This model corresponds to the 'noisy OR' model (Pearl 1986). Conditioned on the interest vector, the probability of the phrase j appearing in utterance $U_t^v[j]$ at time t is given by:

$$p(U_t^v[j]=1|I_t^v) = 1-(\prod_{l=1}^{K}(1-\alpha_{j,l})^{I_t^v[l]}). \quad (1)$$

Here $\alpha_{j,l} = p(U[j]=1|I[l]=1)$ is the probability of phrase j for interest l and is independent of time t and user v.

Including user-specific activity levels: Some users tend to utter more than others. To incorporate activity level of user v, we extend Eq. 1 as follows (Figure 1(b)):

$$p(U_t^v[j]=1|I_t^v) = 1-(\prod_{l=1}^{K}(1-p_{active}^v * \alpha_{j,l})^{I_t^v[l]}). \quad (2)$$

Here p_{active}^v denotes the activity level of user v. Because of scaling of $\alpha_{j,l}$ with p_{active}^v, even if $\alpha_{j,l}$ is large, $p(U_t^v[j]=1|I_t^v)$ becomes small for a less active user. Therefore, when we see an utterance from user v that includes the phrase j, that will be a strong indicator to the inference procedure of v's interest in l.

Adding influence of friends: Some users are more susceptible than others to information received from their friends. Suppose a user v has received an utterance U from his friend w in time step $t-1$. The influence of w on v will manifest in the form of some of the phrases from U appearing in the utterance of v at time step t. Thus, denoting by $sus_{v,w}$, the susceptibility of user v to his friend w amongst his friends E_v, we extend Eq. 2 as follows (Figure 1(c)):

$$p(U_t^v[j]=1|I_t^v,\{U_{t-1}^w\}_{w\in E_v}) = 1-(\prod_{l=1}^{K}(1-p_{active}^v * \alpha_{j,l})^{I_t^v[l]}) * (1-\underset{w\in E_v}{\mathrm{f}}(sus_{v,w} * U_{t-1}^w[j])). \quad (3)$$

At time step $t-1$, different utterances from various friends of v might have contained the same phrase j. The aggregation function $\underset{w\in E_v}{\mathrm{f}}$ defines the combined susceptibility of v to all of his friends from whom he received the phrase j. The appropriate choice for function f depends on the social network for which inference is being performed. In certain networks, social influence can be additive while in others this influence might quickly saturate.

Clearly, the observation likelihood in Eq. 3 will be reduced by a greater amount for a user who is less susceptible to his friends. Thus, when such a user utters a phrase used by his friends in the previous time step, that will more strongly indicate the existence of the corresponding interest compared to a more susceptible user. In fact, our model allows for detecting such intricacies on an per-edge basis since the susceptibility of a particular user can differ from one friend to another.

Modeling Interest Evolution over Time

It is easy for a user to get distracted for a short time from his intrinsic interests. However, this transience does not change his stable interests. Hence, we propose a temporal extension in which we model interest evolution over time. In particular, the current interests of a user is a function of his interests in the previous time step and the estimate of this interests in the current time step. We extend online learning so that the new prior for time $t+1$ is calculated based on the prior and the posterior at time t. Algorithm 1 gives the online learning algorithm. Here, the hysteresis parameter β controls the sensitivity with which new information affects the update of current interest distribution.

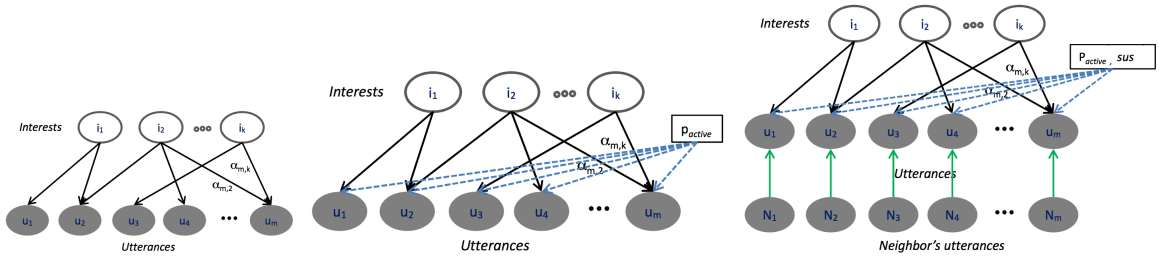
Algorithm 1 The calculation of $p(I_{t+1})$ given $p(I_t)$ and utterances at interval $[t,t+1)$

1: Given prior distribution $p(I_t^v)$ and utterances known
2: Infer $p(I_t^v[l]=1|G_{t-1},\{\mathcal{U}^v\})$ for $1\le l\le K$
3: Update the prior for the next time step using online learning as:
$p(I_{t+1}^v) = \beta * p(I_t^v) + (1-\beta)p(I_t^v|G_{t-1},\{\mathcal{U}^v\})$

Probabilistic Inference

For each user v, the exact computation of $p(I_t^v[l]=1|G_{t-1},\{\mathcal{U}\})$ requires marginalizing over all other interests and computing $P(\mathcal{U})$, which is exponential in the number of interests:

$$p(I_t^v[l]=1|G_{t-1},\{\mathcal{U}\}) \propto \sum_{I_t^v \setminus I_t^v[l]} p(U_t^v|I_t^v,\{U_{t-1}^w\}_{w\in E_v})p(I_t^v). \quad (4)$$



(a) 'Noisy OR' for explaining utterances. (b) Incorporating user-specific activity level. (c) Incorporating user susceptibility by including friends' utterances from the earlier step

Figure 1: Graphical models of increasing complexity for inferring interests of a user from his utterances and the social network.

Hence, the exact computation of marginal posterior distribution is intractable. Therefore, we resort to variational approximation in which we upper and lower bound $p(U_t^v[j]=1|G_{t-1},I_t^v)$ such that the resulting bounds can be computed efficiently. Refer to the variational parameters that are used to provide an upper and lower bound on the likelihood of utterance u_j of user v at time step t to as $\varepsilon_{v,t,j}$ and $q_{v,t,j}$, respectively. Similarly, refer to the set of variational parameters for *all* utterances for user v at time t as $\varepsilon_{v,t}$ and $q_{v,t}$. Finally, refer to the set of variational parameters for user v from time 0 to t as $E_{v,t}$ and $Q_{v,t}$. Next, we show the computation of bounds on the conditional probabilities, and how they in turn enable computing $p(I_t^v[l]=1|G_{t-1},\{\mathcal{U}\})$ efficiently.

Variational upper bound on conditional probability: We start by noting that Eq. 3 can be rewritten as:

$$p(U_t^v[j]=1|I_t^v,\{U_{t-1}^w\}_{w\in E_v}) = e^{\log(1-e^{-x})}, \tag{5}$$

where $x = \underset{w\in E_v}{\mathrm{f}}(sus_{v,w,j} * U_{t-1}^w[j]) + \sum_{l=1}^K (p_{active}^{v,l} * \alpha_{j,l})^{I_t^v[l]}$. The exponent $g(x)=\log(1-e^{-x})$ is a concave function of x, therefore there must exist a variational upper bound for this function that is linear in x (Jaakkola and Jordan 1999). More particularly, Jensen's inequality dictates that $g(x) \leq \varepsilon x - g^*(\varepsilon)$. Substituting this into Eq. 3, we have:

$$\begin{aligned} p(U_t^v[j]=1|I_t^v,\{U_{t-1}^w\}_{w\in E_v}) &= \\ & e^{g(\underset{w\in E_v}{\mathrm{f}}(sus_{v,w,j}*U_{t-1}^w[j])+\sum_{l=1}^K (p_{active}^{v,l}*\alpha_{j,l})^{I_t^v[l]})} \\ & \leq e^{\varepsilon_{v,t,j}(\underset{w\in E_v}{\mathrm{f}}(sus_{v,w,j}*U_{t-1}^w[j])+\sum_{l=1}^K (p_{active}^{v,l}*\alpha_{j,l})^{I_t^v[l]})-g^*(\varepsilon_{v,t,j})} \\ & \equiv p(U_t^v[j]=1|I_t^v,\{U_{t-1}^w\}_{w\in E_v},\varepsilon_{v,t,j}). \end{aligned} \tag{6}$$

Note that $\underset{w\in E_v}{\mathrm{f}}(sus_{v,w,j} * U_{t-1}^w[j])$ is irrespective of user interests and user utterances except for u_j and therefore can be computed efficiently. Both $p_{active}^{v,l}$ and $\alpha_{j,l}$ are known apriori and therefore their product can be treated as a single parameter $\alpha_{v,j,l}$. Therefore, $p(U_t^v[j]=1|I_t^v,\{U_{t-1}^w\}_{w\in E_v},\varepsilon_{v,t,j})$ is the exponential of a term that is linear in number of interests and can be computed efficiently.

Variational lower bound on conditional probability: A different view on Jensen's inequality can help us draw lower bounds on Eq. 3. In particular, lower bounds for a concave function g can be computed as $g(a+\sum l z_l) \geq \sum_l (q_l g(a + \frac{z_l}{q_l})))$, and we can rewrite Eq. 3 as:

$$\begin{aligned} p(U_t^v[j]=1|I_t^v,\{U_{t-1}^w\}_{w\in E_v}) &= e^{g(C+\sum_{l=1}^K (p_{active}^{v,l}*\alpha_{j,l})^{I_t^v[l]})} \\ & \geq e^{\sum_l q_{t,l|j} g(C+\frac{\alpha_{v,j,l} i_l}{q_{v,t,l|j}})} \\ & = e^{\sum_l q_{v,t,l|j} i_l \left[g(C+\frac{\alpha_{v,j,l}}{q_{v,t,l|j}})-g(C)\right]+g(C)} \\ & \equiv p(U_t^v[j]=1|I_t^v,\{U_{t-1}^w\}_{w\in E_v},q_{v,t,j}), \end{aligned} \tag{7}$$

where $C = \underset{w\in E_v}{\mathrm{f}}(sus_{v,w,j} * U_{t-1}^w[j])$, $i_l = I_t^v[l]$ and $\alpha_{v,j,l} = p_{active}^{v,l} * \alpha_{j,l}$. As in the case of the upper bound, this implies that the variational evidence can be incorporated into the posterior distribution in time linear in the size of the interest vector.

Bounds on the posterior marginals: Given the upper and lower bounds in Eq. 6 and Eq. 7, the corresponding bounds for the prior and marginal posterior probabilities at any time step t can be determined as follows. Assume that upper and lower bounds for the likelihood of a particular interest i_l at time step t are given by $P_t^{upper}(i_l|E_{v,t-1},Q_{v,t-1})$ and $P_t^{lower}(i_l|E_{v,t-1},Q_{v,t-1})$, respectively. The set of variational parameters from time 0 to t are $E_{v,t}$ and $Q_{v,t}$. The upper bound for the joint probabilities at time t is:

$$\begin{aligned} P_t(U_t^{v,+},i_l) &= \sum_{I\setminus i_l} p(U_t^{v,+}|i_l,\{U_{t-1}^w\}_{w\in E_v}) * P_t(i_l) \\ & \leq \sum_{I\setminus i_l} p(U_t^{v,+}|i_l,\{U_{t-1}^w\}_{w\in E_v},\varepsilon_{v,t,j}) * P_t^{upper}(i_l|E_{v,t-1},Q_{v,t-1}) \\ & \equiv P_t(U_t^{v,+},i_l|E_{v,t},Q_{v,t-1}). \end{aligned} \tag{8}$$

Similarly, a lower bound can be given as:

$$
\begin{aligned}
P_t(U_t^{v,+}, i_l) &= \sum_{I \setminus i_l} p(U_t^{v,+} | i_l, \{U_{t-1}^w\}_{w \in E_v}) * P_t(i_l) \\
&\geq \sum_{I \setminus i_l} p(U_t^{v,+} | i_l, \{U_{t-1}^w\}_{w \in E_v}, q_{v,t,j}) * P_t^{lower}(i_I | E_{v,t-1}, Q_{v,t-1}) \\
&\equiv P_t(U_t^{v,+}, i_l | E_{v,t-1}, Q_{v,t}).
\end{aligned} \tag{9}
$$

Clearly, variational parameters do not affect the prior distribution estimation at time $t = 0$. Therefore, the base case can be given as $P_0^{upper}(i_l) = P_0^{lower}(i_l) = P_0(i_l)$. Finally, the bounds for the marginal posterior probabilities can be computed as:

$$
\begin{aligned}
&P_t^{lower}(i_l | U_t^{v,+}, E_{v,t}, Q_{v,t}) = \\
&\qquad \frac{P_t(U_t^{v,+}, i_l | E_{v,t-1}, Q_{v,t})}{P_t(U_t^{v,+}, i_l | E_{v,t-1}, Q_{v,t}) + P_t(U_t^{v,+}, \bar{i_l} | E_{v,t}, Q_{v,t-1})},
\end{aligned} \tag{10}
$$

$$
\begin{aligned}
&P_t^{upper}(i_l | U_t^{v,+}, E_{v,t}, Q_{v,t}) = \\
&\qquad \frac{P_t(U_t^{v,+}, i_l | E_{v,t}, Q_{v,t-1})}{P_t(U_t^{v,+}, i_l | E_{v,t}, Q_{v,t-1}) + P_t(U_t^{v,+}, \bar{i_l} | E_{v,t-1}, Q_{v,t})}.
\end{aligned} \tag{11}
$$

The bounds on the priors at $t+1$ can be computed as: $P_{t+1}^{lower}(i_l | E_{v,t}, Q_{v,t}) = \beta P_t^{lower}(i_l | E_{v,t-1}, Q_{v,t-1}) + (1 - \beta) P_t^{lower}(i_l | U_t^v, E_{v,t}, Q_{v,t})$ (Line 4 of Algorithm 1). A similar conclusion can be reached for the upper bound on the prior distributions at time $t+1$.

Bounds for Temporal Updates of Interest Distribution

In our setting, the marginal posterior probabilities $p(\{I_t^v[l] = 1\}_l | G_{t-1}, \{\mathcal{U}\})$ at time t obtained using variational method is used to update the prior distribution $p(I_{t+1}^v)$. Therefore, the errors introduced by the variational transformations can carry over time. We next show that even in this incremental setting where the parameters are updated at each time step (as opposed in the batch setting where all observations are available for updation), we can define upper and lower bounds on the posterior probabilities at time $t+1$ by incorporating the bounds on the prior inferred from time t. Interestingly, such a greedy solution suffices to minimize the sum of errors over time.

In particular, our goal is to compute a tight bound on the likelihood of the observed utterances ($P(\{\mathcal{U}^v\})$) over time. This entails identifying a set of variational parameters $\{E_{v,t}\}_{v,t}$ that provide the tightest bounds for $P(\{\mathcal{U}^v\} | \{E_{v,t}\})$. Since this computation is independent for each user, we ignore subscript over user v. In addition, we also ignore the signal from the friends ($\underset{w \in E_v}{\mathrm{f}}(sus_{v,w,j} * U_{t-1}^w[j])$) as this value is irrespective of user interest vector and can be computed in constant time for each utterance. Since $P(\{\mathcal{U}^v\} | \{E_{v,t}\})$ provides an upper bound for the likelihood of utterances $P(\{\mathcal{U}^v\})$, our goal is to minimize this value, getting as close as possible to the actual likelihood. The optimal choice of variational parameters up until time step t is given as:

$$
\{E_t\}^* = \arg\min_{\{E_t\}} P(\{U_t\} | \{E_t\}). \tag{12}
$$

Let us focus on the subset of variational parameters at a particular time step $t' \leq t$. We aim to show that solving Eq. 12 is equivalent to optimizing for the variational parameters $\varepsilon_0, \varepsilon_1, ..., \varepsilon_t$ at each time step $0, 1, ..., t$ independently in a greedy manner, i.e. $\{E_t\}^*[t'] = \arg\min_{\varepsilon_{t'}} P(U_{t'} | \varepsilon_{t'}, E_{t'-1})$ where $\{E_t\}^*[t']$ denotes the subset of observational parameters in set $\{E_t\}$ that correspond to time $t' \leq t$. For this purpose, we first show that $P(U_t | E_t)$ is a convex function. Note that the computation of negative findings (U^-) i.e. utterances *not* uttered by the user at the particular time step, can be performed in linear time and do not require transformation. Therefore, we focus on the computation of positive findings (U^+), i.e. the set of utterances the users uttered at the particular time step. Clearly, proving this for a particular time step t suffices to show that the more general problem of minimizing the bound for the joint distribution of each time step is also convex since the summation of convex functions is also convex.

Theorem 0.1. *$P(U_t^+ | E_t)$ is a convex function of the variational parameters E_t.*

Proof. To simplify the notation we assume that all of the positive utterances will be transformed. Our formulation can easily be extended to cover the cases where a subset of the utterances will be treated exactly since the variational parameter optimization is independent from exact treatments.

$$
P(U_t^+ | E_t) = \sum_i \left[\prod_j P_t(u_j^+ | i, \varepsilon_{t,j}) \right] \prod_l P_t(i_l | E_{t-1}). \tag{13}
$$

The convexity of $\left[\prod_j P_t(u_j^+ | i, \varepsilon_{t,j}) \right]$ follows from the literature (Jaakkola and Jordan 1999). Here, we show that $\prod_l P_t(i_l | E_{t-1})$ is a convex function of variational parameters E_{t-1}. Considering the update on prior distributions as given in Line 4 of Algorithm 1, we compute the transformed priors as:

$$
P_t(i_l | E_{t-1}) = \beta P_{t-1}(i_l | E_{t-2}) + (1 - \beta) P_{t-1}(i_l | U_{t-1}, \varepsilon_{t-1}). \tag{14}
$$

It suffices to show that $P_{t-1}(i_l | U_t, \varepsilon_{t-1})$ is convex in ε_{t-1} due to induction and because the base case is independent from variational parameters. Note that, $P_{t-1}(i_l | U_t, \varepsilon_{t-1}) = \frac{P_{t-1}(i_l | U_t, \varepsilon_{t-1}) P_{t-1}(i_l | \varepsilon_{t-1})}{P_{t-1}(U_t | \varepsilon_{t-1})} \equiv P_{t-1}(U_t | i_l, \varepsilon_{t-1})$. This follows from the fact that $P_{t-1}(U_t | \varepsilon_{t-1})$ is the normalization term and can be ignored and $P_{t-1}(i_l | \varepsilon_{t-1})$ is in fact independent from ε_{t-1}. Since $P_{t-1}(U_t | i_l, \varepsilon_{t-1})$ is a convex function (Jaakkola and Jordan 1999), we can conclude that $P(U_t^+ | E_t)$ is convex in ε_{t-1}. □

Next, we show that the optimal choice of variational parameters ε_{t-m}, where $1 \leq m \leq t$, is the same at time step t as it is at time step $t - m$.

$$\frac{\partial}{\partial \varepsilon_{t-m}} \log P(U_t^+|E_t) = \frac{\partial}{\partial \varepsilon_{t-m}} \beta^{m-1}(1-\beta) \log P_{t-m}(i|U_t, \varepsilon_{t-m})$$
$$= \frac{\partial}{\partial \varepsilon_{t-m}} (\log P_{t-m}(U_t|i, \varepsilon_{t-m}) + \log P_{t-m}(i|\varepsilon_{t-m}))$$
$$= \frac{\partial}{\partial \varepsilon_{t-m}} \log P_{t-m}(U_t|i, \varepsilon_{t-m}). \quad (15)$$

The first line follows from the inductive nature of prior computation and the second from the Bayes rule. Since the choice of variational parameters ε_t is optimal for any time step $t' \geq t$, the choice of variational parameters that minimize the sum of errors over time is equivalent to the values computed in a greedy manner at each time step.

Experiments

We next describe the experiments we performed to assess if one could ascertain user interests using the model just introduced. The Twitter data set used in the experiments is described §. In §, we define utterances and interests in the Twitter context and provide our parameter choices. In §, we specify the three variants of our general model we study. In §, we present the results of performance evaluation from the user study as well as the analysis of model convergence. Finally, in §, we describe our experience from attempting to use Amazon Mechanical Turk to scale up the user study.

Data Set

We use the Twitter status updates from May 2011 to May 2012 for computing utterances. We made use of a sampled graph to identify the connections between the Twitter users. The graph data used in our experiments is a static snapshot of the follow relations on Twitter, therefore in our experiments we set $G = G_0 = G_1 = ... = G_t$. For evaluation purposes, we focused on a subset of users who tweeted at least fifty times between May 2011- May 2012.

Setup

Utterances We would like to use utterances that carry meaningful signals about interests. For this purpose, we use *entities*, i.e. capitalized n-grams in tweets, to denote phrases present in a tweet. A recent work also obtained promising results from using entities in classifying Twitter content (Michelson and Macskassy 2010). Some of the commonly used entities in our data set are person or place names (e.g. Russell Wilson, Giovanna Fletcher, Ayala Museum, West London), shows/movies (Die Hard, Real Housewives...), products/companies (Snow Leopard, Pizza Parlor...) or special days (National Preparedness Day, Blessed Christmas...). Repeated use of such entities can signal interest in various topics. For instance, a person mentioning various museums like Ayala Museum might be interested in arts and/or traveling. This approach can result in a large number of phrases, many of which are used only by a few users and might not be reliable. Therefore, we considered only entities that were used by at least 150 distinct users.

Interests For defining the set of literals to use as interests, we first used Wikipedia categories. Indeed, Wikipedia has been used to classify tweets in an earlier work (Michelson and Macskassy 2010). However, our analysis of Twitter data revealed that Wikipedia categories tend to get out-of-date and do not keep up with the real time nature of Twitter. For example, 'Hosni Mubarak' is listed under the categories Egyptian Military Academy alumni, Living people, National Democratic Party, Egypt politicians, Knights Grand Cross of the Order of St Michael and St George, Attempted assassination survivors, Vice Presidents of Egypt, Egyptian Air Force air marshals, Egyptian Muslims, but none of these categories matched the intent of the tweets. We, therefore, use a taxonomy derived from the Open Directory Project (ODP). We used the categories in this taxonomy as they provide a clear, meaningful and broad coverage of various real-world interests. Examples include 'Society and Culture', 'Health and Wellness' and 'Careers and Employment'. Hosni Mubarak gets classified under 'Activism and Social Issues', which is much closer to the intent of Twitter usage.

Model Parameters We next describe how we set various parameters in our experiments.

Activity level of a user, p^v_{active}: We compute the activity level of a user as the relative number of tweets that the user posts in comparison to all other tweeters in the network.

Susceptibility of a user to his friends, $sus_{v,w}$ and aggregation function f*:* We set the susceptibility measure of user v to the same value for all of his friends w. This value is computed as the ratio of the total number of re-tweets v makes to the total number of his tweets and denote it as sus^v. We adopt *argmax* as the aggregator given the studies that demonstrate the diminishing and near-constant effect of multiple friend adoptions (Steeg, Ghosh, and Lerman 2011). Thus, the social influence is captured as $1 - \underset{w \in E_v}{\text{argmax}} (sus_{v,w} * U^w_{t-1}[j])$.

Probability of an utterance given the interest, $\alpha_{j,l}$: We first classify the entities into the ODP taxonomy that provides the probability of a particular interest (i_l) for a particular *entity* (u_j). The $\alpha_{j,l}$ values are then computed through the Bayes rule:

$$\alpha_{j,l} = p(U[j] = 1|I[l] = 1) = \frac{p(I[l]|U[j])p(U[j])}{\sum_{j=1}^{j=M} p(U[j] = 1, I[l] = 1)}. \quad (16)$$

Note that the $p(I[l]|U[j])$ values computed by the ODP classifier could conceivably be used to impute the interests of a user. This classifier also does not take the characteristics of the user into consideration. Therefore, we use this classifier merely to compute $\alpha_{j,l}$.

Hysteresis parameter, β: We set $\beta = 0.25$ throughout our experiments. Our experiments show that this choice allows interests to evolve over time, without causing large fluctuations.

Initialization of prior distributions: The prior distribution for an interest i_l for all users at time 0 is computed as $\sum_j \alpha_{j,l} * p(u_j)$, where $p(u_j)$ is the frequency of phrase u_j in our dataset.

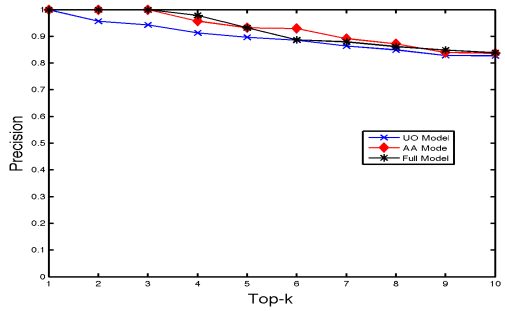


Figure 2: Precision of inferring interests

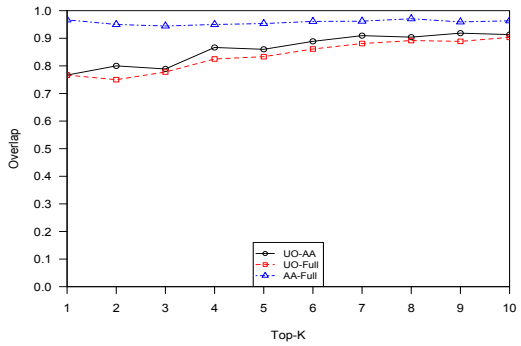


Figure 3: Overlap of Top-k Interests Identified by Different Models

Variants of Our Model

We studied the following variants:

Utterences Only (UO Model): This is the simplest variant that takes into account only user's utterances to infer his interests. This model is depicted in Figure 1(a), with observation likelihood given by Eq. 1.

Activeness Aware (AA Model): In this model, every user has his own activeness parameter. This corresponds to the model depicted in Figure 1(b), with observation likelihood given by Eq. 2.

Full Model: This is our complete model depicted in Figure 1(c), with observation likelihood given by Eq. 3. This model captures both the user's activeness and susceptibility to the influence from his friends.

Evaluation

Methodology The most direct and accurate evaluation of inferred latent interests of a particular user would arguably be to confirm the results from the user for whom inference is performed. We, therefore, attempted to perform exactly this task. The procedure was carried out as follows. We first identified the Twitter users with email addresses by mining the user profiles for email address patterns. From such users, 500 were randomly selected to include in the user study. For each of these 500 users, the top-10 interests identified through *UO, AA,* and *Full* Models are curated. Refer to the list of interests identified by model m for user v as L_v^m, and let $L_v = \cup_m L_v^m$.

We emailed to each user v, through an automated email client, a request for participation in the study. The email first introduces the research project and next provides an embedded php form. The form includes $|L_v|$ questions. Each question is of the form: "Are you interested in *¡interest¿*?", where *¡interest¿* is one of the interests from L_v. The user provides his response by selecting between 'Yes' and 'No'; there was no default option. We also included the text of tweets made by the user as an attachment to the email to refresh the user's memory. Thirty users responded to our request.

Implementation All three variants are implemented through Infer.NET, a framework for running Bayesian inference in graphical models (Minka et al. Microsoft Research Cambridge 2009). The inference task for all three variants are embarrassingly parallel, i.e. $p(I_t^v|G_{t-1},\{\mathcal{U}^v\})$ computation for each v is independent. Therefore we use Dryad (Isard et al. 2007) infrastructure, a general-purpose distributed execution engine for coarse-grain data-parallel applications, to parallelize this task and gain significant improvement in efficiency. In addition to using Dryad as the infrastructure, we use DryadLINQ (Yu et al. 2008) as our programming framework. DryadLINQ provides useful extensions on the LINQ framework and compiles LINQ programs into distributed computations running on the Dryad cluster-computing infrastructure to enable a programming model for large scale distributed computing. Such a technique allows for inferring user interests at scale.

Results We next summarize the key results from our user study.

Precision Results: Figure 2 shows the plot of the precision of the three variants of our model. Precision for a method for the top-k interests for a particular user is computed as the fraction of interests identified that are verified through the user study. We then average precision values across the thirty users to produce the plot.

We see that all three models have high precision. The UO model considers all utterances simultaneously and disentangles the true underlying interest by allowing all interests to compete to explain the utterances. Due to this probabilistic disentanglement, the model is able to infer the interests with high precision. The AA model accounts for differences in the activity of the users by modeling their activity levels. Hence, we find improvements in precision over the UO model, particularly for less active users. The Full model additionally takes into account the influence of user's friends. The model would exhibit greater delta in precision for less susceptible users, but only amongst those with equal activity level. Our test population did not contain many such users, resulting in similar precision for the Full and AA models.

Overlap: Figure 3 presents the overlap of interests identified by different models. The overlap between two models M_1 and M_2 for the top-k interests of given user is defined

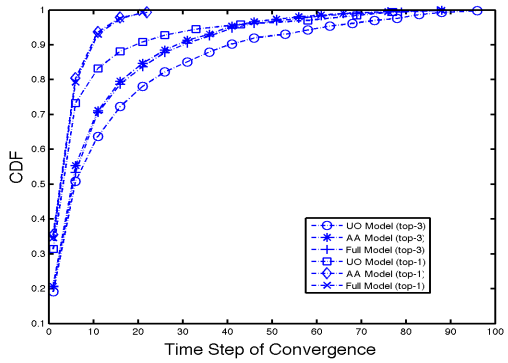


Figure 4: Convergence Analysis

as the size of the intersection set of the top-k interests of M_1 and M_2, normalized by k. The overlap between the two models M_1 and M_2 is then defined as the mean overlap over all users. It is instructive to examine Figure 3 in conjunction with Figure 2. We see that the interests identified by *AA* and *Full* models in Figure 3 have large overlap. No wonder, we see little gain in precision as we advance from *AA* to *Full Model* in Figure 2. On the other hand, the *UO Model* fails to identify some interests that *AA* and *Full* models do identify, leading to a lower performance of the *UO Model* in Figure 2.

Convergence: We also investigated how quickly our model learns user interests. We present the results for the interests identified for each of our 500 users. For each interest, we identify the confidence the model has at the last iteration (identified by the prior probability of that particular interest at the last time step) and determine the time step at which the inference reached 95% of this value. We mark this time step as the convergence step and average the time steps needed for the top-1 and top-3 interests across users. In Figure 4, we plot the cumulative distribution of these values for the three variants of our model. The results show that *AA Model* and the *Full Model* converge quite fast while the convergence is relatively slower for *UO Model*. The inference converges much faster for the top-1 interest. In general, the results indicate that our inference technique learns stable user interests in a fast manner. For instance, the full model provides a close approximation of the final probability of the top-1 interest for 90% of the users within only 10 iterations.

Mechanical Turk Study

A somewhat unsatisfactory aspect of the results just presented is the relatively small number of users who volunteered to participate in the user study. We next report on our attempt to validate our results at a larger scale, using the Amazon Mechanical Turk platform (Tur 2011). Although Mechanical Turk has been reported to have been successfully used in tasks such as classification (Ipeirotis 2010), ranking (Heilman and Smith 2010), translation (Callison-Burch 2009) and even social experiments (Wang, Suri, and Watts 2012), we are not aware of any work reporting the use of this platform for identifying latent interests of a user from his utterances. We note that identifying latent interests from noisy user tweets is an inherently hard task. Moreover, mapping this problem to the Mechanical Turk platform and guaranteeing best effort from judges is quite challenging.

Experiment Setup Mechanical Turk is an online labor market where workers are recruited for the execution of tasks (called HITs, acronym for Human Intelligence Tasks) in exchange for a wage. Mechanical Turk workers can be required to have certain "qualifications" prior to completing a HIT to control the quality of experiments. Throughout our experiments we necessitate workers to have at least 95% approval rate. A HIT in our experiments consists of the judge reading the tweets of a particular user and answering binary questions where each question is of the form :"is this user interested in X?", where X is one of the interests identified by *UO*, *AA* or *Full Model*. Given all judgments, we define the prediction of the judges for a particular question of a particular user as the majority vote.

Results We rely on two measures in quantifying the quality of Mechanical Turk experiments: 1) precision/recall/accuracy w.r.t. a ground truth data (precision experiment) and 2) inter-rater reliability that captures how consistent judges are in their judgements (consistency experiment).

Precision Experiment: For this particular experiment, one HIT is created for each user who responded to our email survey (described in §). Each HIT provides the set of tweets of a user v and includes binary questions about the validity of the top six interests from L_v. Note that we already have the ground truth on these interests from the survey results. Given this ground truth, we can compute the accuracy, precision and recall of the Mechanical Turk judgments. We vary the number of judges per hit from 5 to 40 and plot the average values across all the users in Figure 5.

With the largest number of judges (40), this experiment results in precision of 0.93, accuracy of 0.47 and recall of 0.46. Note that there are 155 positive labels (interests identified as being correct through email surveys) and 25 negative labels (interests identified as being incorrect through email surveys) in our data set. Therefore, a random classifier that labels a question "is this user interested in X?" as 'Yes' half of the time would have an expected recall and accuracy of 0.5 and precision of 0.86 indicating that the Mechanical Turk judgements are qualitatively almost random.

Consistency Experiment: For the consistency experiment, we use Fleiss Kappa measure (Gwet 2012) to compute inter-rater reliability. This measure is commonly used for assessing the reliability of agreement between a fixed number of raters when assigning categorical ratings to a number of items. Fleiss Kappa measure (κ) for N number of subjects to be categorized into k categories with n number of ratings per subject is defined to be:

$$\kappa = \frac{\bar{P} - \bar{P}_e}{1 - \bar{P}_e}, \tag{17}$$

where $\bar{P} - \bar{P}_e$ captures the true agreement and $1 - \bar{P}_e$ captures the amount agreement that would be attained by chance. Let

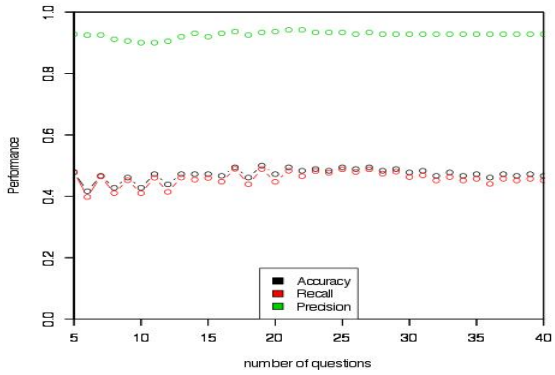


Figure 5: Evaluation of Mechanical Turk judgments

n_{ij} represent the number of raters who assigned the j^{th} category to the i^{th}. $\bar{P}$ and $\bar{P}_e$ can be computed as follows:

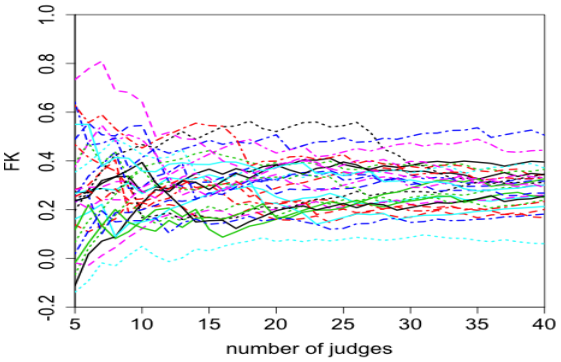
$$\bar{P}_e = \sum_{j=1}^{k} (\frac{1}{Nn} \sum_{i=1}^{N} n_{ij})^2$$
$$\bar{P} = \frac{1}{Nn(n-1)} (\sum_{i=1}^{N} \sum_{j=1}^{k} n_{ij}^2 - Nn) \quad (18)$$

In our case, the number of raters n corresponds to the number of Turkers who rate the interests of a user, the number of subjects N correspond to the number of interests of a user on which we obtain judgments, and the number of categories k is 2 corresponding to the two possible outcomes ('Yes' or 'No'). We employ the same HIT as we used in the precision experiments for the top six interests of every user for whom we had survey results and get 40 judgments per HIT.

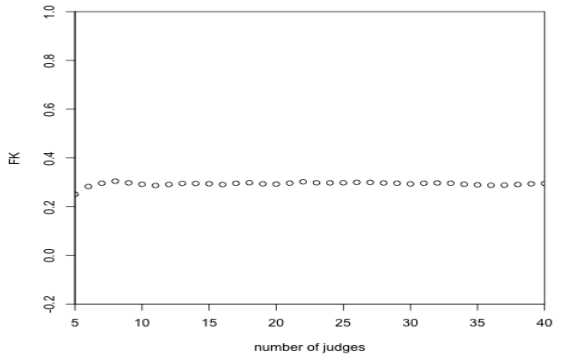
Figure 6(a) provides the change in κ for each of the thirty users as we increase the number of judges. In addition, Figure 6(b) provides an overview of the results by presenting the average κ across the thirty users. These two plots show that below 10 judges, the κ values are unstable, but they stabilize afterwards. The range of κ lies between $0.1 - 0.6$ (all below substantial agreement). Unfortunately, even for larger number of judges, the average agreement is around 0.3, indicating that the judges cannot agree on what interests a particular user has.

To further ensure that the lack of agreement was not something specific to the thirty users who participated in our survey, we performed another experiment, increasing the number of HITS (number of Twitter users whose inferred interests are evaluated through Mechanical Turk judges) to 100. The corresponding plot is given in Figure 6(c). We see that the average agreement is still very low.

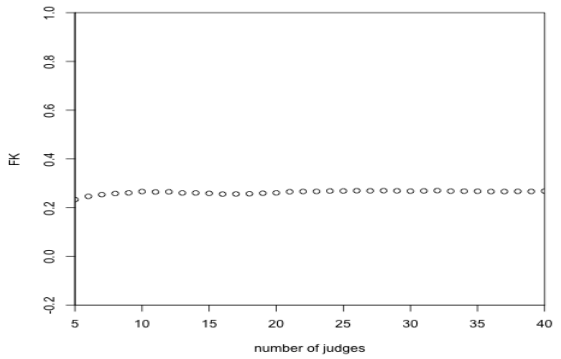
Concluding Remarks on the Mechanical Turk Study While our user study based on directly asking the users if the inferred interests matched their actual interests demonstrated the high precision of our techniques, we wanted to expand the evaluation to cover a much larger number of users. We extensively evaluated the feasibility of using the



(a) κ per HIT (30 HITS for 30 email survey responders)



(b) Average κ (Average over 30 email survey responders)



(c) Average κ over 100 Twitter users

Figure 6: Fleiss Kappa Analysis of Mechanical Turk

Mechanical Turk platform for this purpose. Unfortunately, various experiments presented here show that identifying latent interests of some other person based on his utterances is a hard task for human beings. We in fact tried various permutations such as reordering the tweets shown to the judges and reordering on interests in the HIT, but found the performance to be still poor. Even though one can perform this task in large scale, possibly for millions of users for a nominal cost, the quality of results would be questionable at best.

Conclusion

We studied whether the latent interests of users can be inferred using observational data available through online social networks. We proposed a novel probabilistic model of social data for this purpose. Inference in this model uncovers the interests of the users taking into account their utterances, their activeness level in the network, and susceptibility to their friends' influence. To allow for the fact that the interests change over time, we provided an online inference algorithm that balances between the current estimate of interests and the previous estimate. As a side benefit, our study advances the state of the art by applying variational methods in an online manner and proving optimality with respect to an equivalent batch solution. Our approach is unsupervised and can easily scale to a large number of interests and users.

We applied our model on Twitter data and found that one can accurately pinpoint interest of Twitter users using our methodology. By evaluating the methodology through a user study, we provide the most direct measurement of the goodness of our inference technique. The results show a precision of 0.9 for the top-5 interests. We also investigated the possibility of using the Amazon Mechanical Turk platform to evaluate our methodology in large scale. The extensive evaluation performed through various experiments shows the shortcomings of the Mechanical Turk platform for this task and highlights the importance of rigorous evaluation, even if small scale, over shaky large-scale evaluations.

References

Ahmed, A.; Low, Y.; Aly, M.; Josifovski, V.; and Smola, A. J. 2011. Scalable distributed inference of dynamic user interests for behavioral targeting. In *KDD '11*, 114–122. ACM.

Bing, L.; Lam, W.; and Wong, T.-L. 2011. Using query log and social tagging to refine queries based on latent topics. In *CIKM '11*, 583–592. ACM.

Callison-Burch, C. 2009. Fast, cheap, and creative: evaluating translation quality using amazon's mechanical turk. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, 286–295. Association for Computational Linguistics.

Chen, Y.; Pavlov, D.; and Canny, J. F. 2009. Large-scale behavioral targeting. In *KDD '09*, 209–218. ACM.

Goffman, E. 1959. *The Presentation of Self in Everyday Life*. Anchor, 1 edition.

Gwet, K. 2012. *Handbook of Inter-Rater Reliability (3rd Edition): The Definitive Guide to Measuring the Extent of Agreement Among Multiple Raters.* Advanced Analytics, LLC.

Hassan, A.; Jones, R.; and Klinkner, K. L. 2010. Beyond dcg: user behavior as a predictor of a successful search. In *WSDM '10*, 221–230. ACM.

Heilman, M., and Smith, N. A. 2010. Rating computer-generated questions with mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, CSLDAMT '10, 35–40. Stroudsburg, PA, USA: Association for Computational Linguistics.

Ipeirotis, P. G. 2010. Analyzing the amazon mechanical turk marketplace. *XRDS* 17(2):16–21.

Isard, M.; Budiu, M.; Yu, Y.; Birrell, A.; and Fetterly, D. 2007. Dryad: distributed data-parallel programs from sequential building blocks. *ACM SIGOPS Operating Systems Review* 41(3):59–72.

Jaakkola, T. S., and Jordan, M. I. 1999. Variational methods and the qmr-dt database. *Journal of Artificial Intelligence* 10:291–322.

Kim, H. R., and Chan, P. K. 2003. Learning implicit user interest hierarchy for context in personalization. In *IUI '03*, 101–108. ACM.

Liu, K., and Tang, L. 2011. Large-scale behavioral targeting with a social twist. In *CIKM*, 1815–1824.

Michelson, M., and Macskassy, S. A. 2010. Discovering users' topics of interest on twitter: a first look. In *AND '10*, 73–80. ACM.

Minka, T.; Winn, J.; Guiver, J.; and Kannan, A. Microsoft Research Cambridge, 2009. Infer.net 2.3.

Pearl, J. 1986. Fusion, propagation, and structuring in belief networks. *Artificial Intelligence* 29.

Pennacchiotti, M., and Popescu, A.-M. 2011. Democrats, republicans and starbucks afficionados: user classification in twitter. In *KDD '11*, 430–438. ACM.

Rao, D.; Yarowsky, D.; Shreevats, A.; and Gupta, M. 2010. Classifying latent user attributes in twitter. In *SMUC '10*, 37–44. ACM.

Steeg, G. V.; Ghosh, R.; and Lerman, K. 2011. What stops social epidemics? In *ICWSM*.

Sugiyama, K.; Hatano, K.; and Yoshikawa, M. 2004. Adaptive web search based on user profile constructed without any effort from users. In *WWW '04*, 675–684. ACM.

2011. *Amazon Mechanical Turk, Requester Best Practices Guide*. Amazon Web Services.

Wang, J.; Suri, S.; and Watts, D. J. 2012. Cooperation and assortativity with dynamic partner updating. *Proceedings of the National Academy of Sciences*.

Weng, J.; Lim, E.-P.; Jiang, J.; and He, Q. 2010. Twitterrank: finding topic-sensitive influential twitterers. In *WSDM '10*, 261–270. ACM.

Yu, Y.; Isard, M.; Fetterly, D.; Budiu, M.; Erlingsson, Ú.; Gunda, P. K.; and Currey, J. 2008. Dryadlinq: A system for general-purpose distributed data-parallel computing using a high-level language. In *OSDI*, volume 8, 1–14.