On Clustering Heterogeneous Social Media Objects with Outlier Links

Guo-Jun Qi
Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign
Urbana, IL
qi4@illinois.edu

Charu C. Aggarwal
IBM T.J. Watson Research Lab
Hawthorne, NY
charu@us.ibm.com

Thomas S. Huang
Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign
Urbana, IL
huang@ifp.uiuc.edu

ABSTRACT

The clustering of social media objects provides intrinsic understanding of the similarity relationships between documents, images, and their contextual sources. Both content and link structure provide important cues for an effective clustering algorithm of the underlying objects. While link information provides useful hints for improving the clustering process, it also contains a significant amount of noisy information. Therefore, a robust clustering algorithm is required to reduce the impact of noisy links. In order to address the aforementioned problems, we propose heterogeneous random fields to model the structure and content of social media networks. We design a probability measure on the social media networks which output a configuration of clusters that are consistent with both content and link structure. Furthermore, noisy links can also be detected, and their impact on the clustering algorithm can be significantly reduced. We conduct experiments on a real social media network and show the advantage of the method over other state-of-the-art algorithms.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Clustering

General Terms

Algorithms

Keywords

Social media networks, robust clustering, noisy links

1. INTRODUCTION

Social media networks represent social repositories for the sharing of multimedia objects between users, and a platform for interactions which are based on this content. The media objects may correspond to images, videos, or even text content. An example of such a network would be *Flickr* which provides a platform for the sharing of images between users. While the problem of clustering has been widely studied with the use of pure content, the social interactions in such networks provide vitally important information which can be used in order to improve the clustering process. The social links can be used as vital clues which can be used in order to cluster the media objects with consistent themes together. One interesting characteristic of such social media networks is that they provide a way to consistently cluster not just the social media objects, but also the users who contribute these objects. The content of the social media objects, and the user links to these objects provide mutually re-enforcing information which can be leveraged for a robust clustering process. In addition, user tagging information (or comments) are available to enhance the clustering process. A major challenge in this problem is that it requires us to simultaneously cluster data of many different kinds, such as images, user tags, user nodes and the links which represent the relationships between them. Clearly, an integrated and holistic approach to social media object clustering is required, which can help us understand the content themes in the different clusters with the help of user-centered social hints.

As illustrated in Figure 1, there are two types of links in generic social media networks - the links between the users and the social media objects, as well as the context links between media objects and their context objects such as user tags or comments. These links, along with the associated content of the media objects, play an important role in determining the clustering of the social media objects with a holistic approach. In this paper we will use a heterogeneous random field to model the natural clusters in the social media content and their underlying linkages in a seamless framework.

The social links between objects and users are often noisy, in which the links between users and objects could be spam, erroneous, or incidental links. Thus, the presence of such links may actually reduce the quality of the clustering, if such misleading social cues are used blindly. The detection and removal of such outlier links are useful in improving the underlying clustering quality. The traditional problem of clustering focusses on determining the outliers in the underlying *objects* rather than the links. However, since links form one of the key social cues in the clustering process, it implies

WSDM'12, February 8–12, 2012, Seattle, Washington, USA.

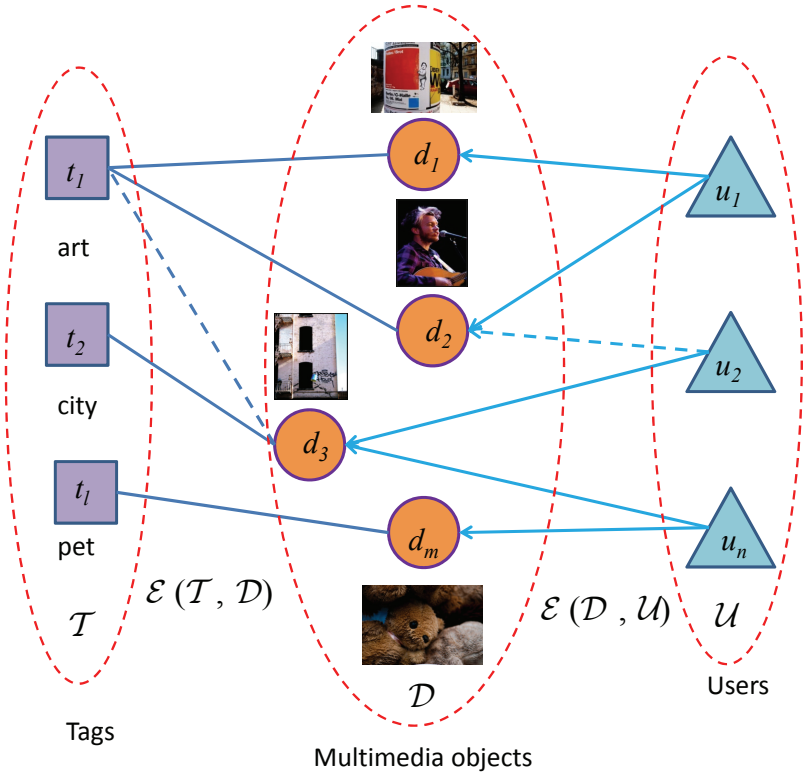


Figure 1: Illustration of social media network. Such network consists of three component graphs - the graphs of users, multimedia objects, and the annotated tags. Users indicate their preferences on media objects, while each object can be linked with multiple tags.

that the abnormalities in the linkages should be removed in order to enhance clustering quality. Our model will jointly detect the outlier links as well as model the underlying clustering structure. We will see that accurately identifying the abnormal links can improve the consistency of clustering the social media objects. On the other hand, a better understanding of the clustering structure can improve the accuracy of detecting of outlier links in the social networks. These two tasks enhance the effectiveness of each other, are therefore presented in a unified framework.

The remainder of this paper is organized as follows. In section 2, we introduce related work on clustering with social media networks. We formally define the problem in Section 3 and propose the heterogeneous random fields for clustering in Section 4. In Section 5 we present efficient inference algorithms for determining the cluster configuration with the highest probability. We present experiments on real data sets in Section 6. Finally, we present the conclusions in Section 7.

2. RELATED WORK

The problem of clustering has been studied extensively in the database, machine learning, multi-media and text literature. An extensive overview of clustering algorithms from a very generic context may be found in [14]. In the context of different kinds of multi-media data, especially text and images, a variety of content based algorithms have also been proposed [1, 5, 8, 13, 16, 20, ?]. However, such methods do not use the rich information which are available in a social network, such as the linkage information and user activity in the form of tags and comments.

On the other hand, from a linkage viewpoint, clusters may be considered as a group of nodes which are densely connected by edges in the networks. For example, a variety of node clustering algorithms for graphs have been proposed in [7, 12, 4, 6, 15, 18]. Some recent work [17, 21, 22, 23] uses a combination of content and link structure for clustering purpose, however, they do not consider the noisy links, and other kinds of social cues which are helpful for clustering social media objects. In particular, some recent work has been focussed on clustering web images [3, 10] with the use of surrounding text from the web page. However, this approach requires a copious amount of text unlike the tags and comments in social media, and also does not use social linkage structure for the clustering process. In this paper, we develop a robust clustering algorithm that is sensitive to the noisy links in the networks for determining the underlying clusters. Our approach is much more general than the methods discussed in earlier work in terms of combining data of many different kinds such as multi-media objects, tags, and social-linkage structure of actors to objects.

3. PROBLEM DEFINITION

In this paper, we consider jointly clustering the media objects, textual context objects, and users in social media networks. As a mathematical abstraction, we use a tri-

partite graph $\mathcal{G} = (\mathcal{U}, \mathcal{D}, \mathcal{T})$ to denote the social media networks, where $\mathcal{U} = \{u_1, u_2, \cdots, u_n\}$ is the set of users, $\mathcal{D} = \{d_1, d_2, \cdots, d_m\}$ is the set of social media objects, and $\mathcal{T} = \{t_1, t_2, \cdots, t_l\}$ is a set of tags or comment keywords made by users on the social media objects $\mathcal{D}$. The users and objects in $\mathcal{U}$ and $\mathcal{D}$ are connected between each other by a set of links $\mathcal{E}(\mathcal{U}, \mathcal{D}) = \{(u_i, d_j)\}$ which correspond to user u_i interest in object d_j. In addition, collections of links $E(\mathcal{D}, \mathcal{T})$ exist between $\mathcal{D}$ and $\mathcal{T}$, which means one object d_j in $\mathcal{D}$ is annotated with user tag keyword t_l if the link $(d_j, t_l) \in \mathcal{E}(\mathcal{D}, \mathcal{T})$. In the event that the comment or tag contains multiple keywords, then a link exists from multiple nodes to the corresponding objects. The content of each object d_j in $\mathcal{D}$ is also summarized by a feature vector $\mathbf{x}_j \in \mathbb{R}^d$ of dimensionality d. For example, this feature vector could be the image features such as the visual words, or the tf-idf scores of a text document. The goal is to cluster the social media objects, users, and tags in this tri-partite graph simultaneously by investigating the associated link structure between component graphs as well as the object content. Associated with each user u_i, object d_j and tag t_l, there will be variables $c(u_i)$, $c(d_j)$ and $c(t_l)$ taking the same set of values from $\{1, 2, \cdots, k\}$ to indicate their cluster membership. Intuitively, the links between graphs should connect the objects in the same cluster.

While the social media objects may be clustered purely with the use of content, this ignores the fact that similar social cues for different objects can provide very useful hints. For example, images which are preferred by similar users, or have similar tags are much more likely to belong to the same cluster. One complicating factor with such an approach is that the social links between social media objects, users, and tags are quite noisy and often inconsistent. In order to mitigate this negative effect, we propose a robust clustering algorithm that can detect and remove the noisy links during the clustering process. Associated with each link in $\mathcal{E}(\mathcal{U}, \mathcal{D})$ we introduce the binary variable $n(u_i, d_j)$ to indicate that the link $e(u_i, d_j)$ is noisy if it makes on the value of 1. Similarly, we introduce the variable $n(d_j, t_l)$ to indicate that the link $e(d_j, t_l)$ is noisy.

4. HETEROGENEOUS RANDOM FIELD MODEL

In this section, we present our heterogeneous random field model (HRF), to determine the clusters of social media objects. In order to facilitate the random field model, we introduce an energy function on the edges. We show that minimizing the energy function will yield the most probable cluster configuration on $\mathcal{G}$, which is consistent with the social media content and link structure. The noise on the links are detected and utilized in order to determine their relevance to the clustering process.

We define the following energy functions for model construction. The first type of energy function is defined on the social links connecting users and media objects.

$$\begin{aligned} &E_{ij}\left(c\left(u_i\right), c\left(d_j\right), n\left(u_i, d_j\right)\right) \\ &= n\left(u_i, d_j\right)\varepsilon + \left(1 - n\left(u_i, d_j\right)\right)\delta\left[\!\left[c\left(u_i\right) \neq c\left(d_j\right)\right]\!\right] \\ &= \begin{cases} \delta\left[\!\left[c\left(u_i\right) \neq c\left(d_j\right)\right]\!\right], n\left(u_i, d_j\right) = 0 \\ \varepsilon, n\left(u_i, d_j\right) = 1 \end{cases} \end{aligned} \tag{1}$$

Here, $\delta[\![\cdot]\!]$ is the indicator function which outputs 1 when the condition in $[\![\cdot]\!]$ holds true and 0 otherwise. The variable $\varepsilon \geq 0$ denotes the confidence level of the links. A lower value of ε indicates a high level of noise on the links. In the extreme case, when $\varepsilon = 0$, all the links will be judged by the HRF as noisy links since the energy function will be minimized by $n(u_i, d_j) = 1$. On the other hand, when $\varepsilon = 1$, the links are treated as very relevant and the cluster membership of u_i and d_j will be regulated by this. We note that the clustering process outputs not just the membership of social media objects to clusters, but also the users to the analogous clusters. Thus, such an approach can also be used to facilitate recommendations in social media networks.

The second type of energy function is defined on the links between objects and tags.

$$\begin{aligned} &E_{jl}\left(c\left(d_j\right), c\left(t_l\right), n\left(d_j, t_l\right)\right) \\ &= n\left(d_j, t_l\right)\varepsilon + \left(1 - n\left(d_j, t_l\right)\right)\delta\left[\!\left[c\left(d_j\right) \neq c\left(t_l\right)\right]\!\right] \\ &= \begin{cases} \delta\left[\!\left[c\left(d_j\right) \neq c\left(t_l\right)\right]\!\right], n\left(d_j, t_l\right) = 0 \\ \epsilon, n\left(d_j, t_l\right) = 1 \end{cases} \end{aligned} \tag{2}$$

Here $\epsilon \geq 0$ denotes the confidence level on the quality of the context links between objects and tags. A larger value of ϵ indicates higher quality of the underlying social cues based on the context information for the clustering process.

In addition, we model the feature vector $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m\}$ associated with the social media objects, given their cluster membership. For simplicity, we assume the objects are composed by a set of discrete features $\{w_1, w_2, \cdots, w_q\}$. For generality, we assume that such features may be of different kinds depending upon the kind of object at hand. For example, for an image object, the feature could correspond to visual words, whereas for a text object, it could correspond to the actual word. For each object d_j, each element of its content vector $\mathbf{x}_j = [x_{j1}, x_{j2}, \cdots, x_{jq}]^T$ counts the number of occurrences of this feature in the object. Now we use parameters $\gamma_k = \{\gamma_{k1}, \gamma_{k2}, \cdots, \gamma_{kq}\}$ to denote the probability of the features appearing in the k-th cluster, i.e., $\gamma_{cv} = P(w_v | c(d_j) = c), 1 \leq v \leq q$. Then the feature counts $\mathbf{x}_j$ of the object d_j are generated following the multi-nomial distribution as follows:

$$P\left(\mathbf{x}_j | c\left(d_j\right) = c\right) = \prod_{v=1}^{q} P\left(x_{jv} | c\left(d_j\right) = c\right) = \prod_{v=1}^{q} \gamma_{cv}^{x_{jv}} \tag{3}$$

Based on the above two types of energy functions and generative model of the feature vectors, we can define a random field on the heterogeneous tri-partite graph $\mathcal{G}$ as follows:

$$\begin{aligned} &P\left(\mathcal{C}, \mathcal{N}, \mathcal{X}\right) \propto \\ &\exp\left\{-\lambda \sum_{e\left(u_i, d_j\right) \in \mathcal{E}(\mathcal{U}, \mathcal{D})} E_{ij}\left(c\left(u_i\right), c\left(d_j\right), n\left(u_i, d_j\right)\right)\right\} \\ &\cdot \exp\left\{-\lambda \sum_{e\left(d_j, t_l\right) \in \mathcal{E}(\mathcal{D}, \mathcal{T})} E_{jl}\left(c\left(d_j\right), c\left(t_l\right), n\left(d_j, t_l\right)\right)\right\} \\ &\cdot \prod_{d_j \in \mathcal{D}} P\left(\mathbf{x}_j | c\left(d_j\right) = c\right) \end{aligned} \tag{4}$$

Here, $\mathcal{C} = \{c(u_i), c(d_j), c(t_l) | u_i \in \mathcal{U}, d_j \in \mathcal{D}, t_l \in \mathcal{T}\}$ is the configuration of the cluster membership over the whole graph $\mathcal{G}$, $\mathcal{N} = \{n(u_i, d_j), n(d_j, t_l) | e(u_i, d_j) \in \mathcal{E}(\mathcal{U}, \mathcal{D}), e(d_j, t_l) \in \mathcal{E}(\mathcal{D}, \mathcal{T})\}$ denotes the noisy links, and $\mathcal{X} = \{x_1, x_2, \cdots, x_m\}$ is the set of all feature vectors associated with the social media objects in the network. The optimal configuration of cluster memberships of the objects in $\mathcal{G}$, which is consistent with the link structure and object content, can be solved by

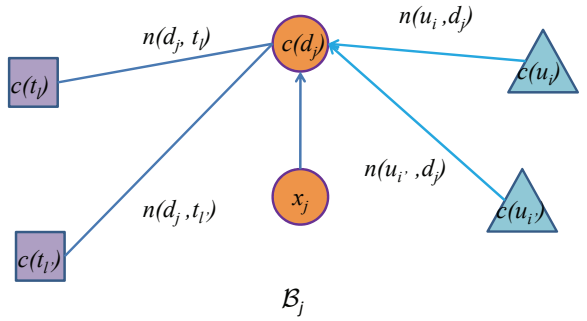


Figure 2: Illustration of the Markov blanket upon which $c(d_j)$ is dependent.

maximizing this probability measure given by $P(\mathcal{C}, \mathcal{N}, \mathcal{X})$. At the same time, the noise on the links can be detected by the edge-based variables, since these are included in the model.

4.1 Analysis

The probability measure given by Eq. (4) defines an interdependency structure between the cluster assignment $\mathcal{C}$, the noisy link detection $\mathcal{N}$ and the social media object content $\mathcal{D}$. Before giving the inference and learning algorithm of this probabilistic model, we discuss some of its properties which are beneficial to understanding the dependency between the cluster assignment task and the link relevance in the social media network.

From the probability measure in Eq. (4), we can obtain the conditional probability of the object d_j belonging to cluster k when the cluster membership of the other objects, and the corresponding link relevance variables:

$$\begin{aligned} &P(c(d_j) = c|\mathcal{C}\backslash c(d_j), \mathcal{N}, \mathcal{X}) \propto P(c(d_j) = c|\mathcal{B}_j) \\ &\propto \prod_{e(u_i,d_j)\in\mathcal{E}(\mathcal{U},\mathcal{D})} P(c(d_j) = c|n(u_i, d_j), c(u_i)) \\ &\cdot \prod_{e(d_j,t_l)\in\mathcal{E}(\mathcal{D},\mathcal{T})} P(c(d_j) = c|n(u_i, d_j), c(u_i)) \\ &\cdot P(\mathbf{x}_j|c(d_j) = c) \end{aligned} \tag{5}$$

Here, $\mathcal{C}\backslash c(d_j)$ denotes the variables of cluster membership of the objects excluding the object d_j, and $\mathcal{B}_j$ is the Markov Blanket upon which $c(d_j)$ is dependent. As illustrated in Figure 2, $c(d_j)$ only depends on the objects which are linked to d_j as well as the noisy link variables that connect d_j. Formally, we have

$$\begin{aligned} &\mathcal{B}_j = \{x_j, c(u_i), c(t_l), n(u_i, d_j), n(d_j, t_l)|e(u_i, d_j) \\ &\in E(U, D), e(d_j, t_l) \in E(D, T)\} \end{aligned} \tag{6}$$

For each link $e(u_i, d_j)$ in $\mathcal{E}(\mathcal{U}, \mathcal{D})$, there are the following two cases depending on the value of the associated link variable:

- when $n(u_i, d_j) = 1$, it indicates $e(u_i, d_j)$ is a noisy link. Then, we have:

$$P(c(d_j) = c|n(u_i, d_j), c(u_i)) \propto \exp\{-\lambda\varepsilon\} \tag{7}$$

  In this case, the membership of d_j is uniformly distributed over all the clusters since the cluster membership of its adjacent user u_i should not affect its membership since the link between the two is noisy.

- when $n(u_i, d_j) = 0$, it is a normal link, and we have:

$$P(c(d_j) = c|n(u_i, d_j), c(u_i)) \propto \exp\{-\lambda\delta[\![c(u_i) \neq c]\!]\} \tag{8}$$

  In this case, when the linked user u_i also belongs to the cluster k, the belief of the object d_i being in cluster k is enhanced; otherwise, such belief will be reduced.

On the other hand, the conditional probability of noisy link variables $n(u_i, d_j)$ and $n(d_j, t_l)$ is defined as follows:

$$P(n(u_i, d_j)|\mathcal{C}, \mathcal{N}\backslash n(u_i, d_j), \mathcal{X}) \propto P(n(u_i, d_j)|c(u_i), c(d_j)) \tag{9}$$

with $c(u_i)$ and $c(d_j)$ in its Markov blanket. We differentiate two different cases for this conditional probability

- When $c(u_i) = c(d_j)$, user u_i and social media object d_j belong to the same cluster, and we have

$$P(n(u_i, d_j) = 0|c(u_i), c(d_j)) = \frac{1}{1 + \exp(-\lambda\varepsilon)}$$

  The posterior that $e(u_i, d_j)$ is not a noisy link is greater than 0.5 in this case. This implies that the belief that the link is not noisy is greater than the opposite belief that the link is noisy.

- When $c(u_i) \neq c(d_j)$, user u_i and object d_j are in different clusters, we have:

$$P(n(u_i, d_j) = 0|c(u_i), c(d_j)) = \frac{1}{\exp(\lambda(1 - \varepsilon)) + 1}$$

  This enhances the belief that $e(u_i, d_j)$ is a noisy link. Actually, when $0 \leq \varepsilon \leq 1$, the posterior probability of $e(u_i, d_j)$ being a normal link is less than 0.5. While $\varepsilon > 1$, the probability that $e(u_i, d_j)$ is a normal link is greater than 0.5. It is reasonable since a larger ε means a stronger belief on normal links. When $\varepsilon \to +\infty$, $P(n(u_i, d_j) = 0|c(u_i), c(d_j)) \to 1$.

A similar discussion can be applied to links between social media objects and tags. The above property of the probability measure is consistent with our intuition that the objects connected by normal links should have similar cluster membership and the belief of a normal link can be enhanced by the linked objects belong to the same cluster.

5. INFERENCE AND PARAMETRIC ESTIMATION WITH HRF

In this section, we present an efficient algorithm to infer the most probable configuration of clusters on social media networks based on the probability measure in Equation (4), as well as the model parameters.

5.1 Inference

The most probable configuration of clusters and the detection of noisy links in social media networks can be jointly inferred as follows:

$$\mathcal{C}^\star, \mathcal{N}^\star = \arg\max_{\mathcal{C},\mathcal{N}} P(\mathcal{C}, \mathcal{N}, \mathcal{X})$$

It is an NP-hard problem to find the exact solution to the above optimization problem. Fortunately, efficient algorithms exist to find the approximate solutions to the HRF model. In general, we use the Gibbs Sampling [11] algorithm to sample a sequence of values for the variables in

Algorithm 1 Gibbs Sampling of Heterogeneous Random Fields

input the number of sampling iterations T.
  Initialize $s(u_i,c), s(d_j,c), s(t_l,c) \leftarrow 0$ for all $c \in \{1,2,\cdots,K\}$, $u_i \in \mathcal{U}, d_j \in \mathcal{D}, t_l \in \mathcal{T}$; and $s(e(u_i,d_j),0), s(e(u_i,d_j),1), s(e(d_j,t_l),0), s(e(d_j,t_l),1) \leftarrow 0$ for each $e(u_i,d_j) \in \mathcal{E}(\mathcal{U},\mathcal{D})$ and $e(d_j,t_l) in \mathcal{E}(\mathcal{D},\mathcal{T})$.
  Initialize the variables in $\mathcal{C}$ and $\mathcal{N}$.
  for $t = 1,\cdots,T$ **do**
    for each u_i in $\mathcal{U}$ **do**
      Sample $c(u_i)$ from $\{1,2,\cdots,k\}$ according to the posterior probability $P(c(u_i)|\mathcal{C}\backslash c(u_i),\mathcal{N},\mathcal{X})$.
      $s(u_i,c(u_i)) \leftarrow s(u_i,c(u_i)) + 1$
    end for
    for each d_j in $\mathcal{D}$ **do**
      Sample $c(d_j)$ from $\{1,2,\cdots,k\}$ according to the posterior probability $P(c(d_j)|\mathcal{C}\backslash c(d_j),\mathcal{N},\mathcal{X})$.
      $s(d_j,c(d_j)) \leftarrow s(d_j,c(d_j)) + 1$.
    end for
    for each t_l in $\mathcal{D}$ **do**
      Sample $c(t_l)$ from $\{1,2,\cdots,k\}$ according to the posterior probability $P(c(t_l)|\mathcal{C}\backslash c(t_l),\mathcal{N},\mathcal{X})$.
      $s(t_l,c(t_l)) \leftarrow s(t_l,c(t_l)) + 1$.
    end for
    for each $e(u_i,d_j)$ in $\mathcal{E}(\mathcal{U},\mathcal{D})$ **do**
      Sample $n(u_i,d_j)$ from $\{0,1\}$ according to the posterior probability $P(n(u_i,d_j)|\mathcal{C},\mathcal{N}\backslash n(u_i,d_j),\mathcal{X})$.
      $s(e(u_i,d_j),n(u_i,d_j)) \leftarrow s(e(u_i,d_j),n(u_i,d_j)) + 1$.
    end for
    for each $e(d_j,t_l)$ in $\mathcal{E}(\mathcal{U},\mathcal{D})$ **do**
      Sample $n(d_j,t_l)$ from $\{0,1\}$ according to the posterior probability $P(n(d_j,t_l)|\mathcal{C},\mathcal{N}\backslash n(d_j,t_l),\mathcal{X})$.
      $s(e(d_j,t_l),n(d_j,t_l)) \leftarrow s(e(d_j,t_l),n(d_j,t_l)) + 1$.
    end for
  end for
output $s(u_i,\cdot), s(d_j,\cdot), s(t_l,\cdot), s(e(u_i,d_j),\cdot), s(e(d_j,t_l),\cdot)$

HRF. Then the most probable cluster configuration can be obtained by the most frequent samples for each cluster variables in $\mathcal{C}$. Moreover, these sampled variables will also be used to estimate the model parameters as in the following subsection.

First, all the variables in $\mathcal{C}$ and $\mathcal{N}$ are randomly initialized. Then, in each sampling step, one variable is sampled based on the conditional probability of the current variable, given that others are fixed. In Eqn (5), we computes the conditional probability of cluster variables for each object given the other variables (including the other cluster variables in $\mathcal{C}$ and the link variables in $\mathcal{N}$) are fixed. A new variable is sampled according to this conditional probability. Algorithm 1 summarizes this sampling process and stores the number of sampled values in $s(u_i,c), s(d_j,c), s(t_l,c)$ and $s(e(u_i,d_j),0), s(e(u_i,d_j),1), s(e(d_j,t_l),0), s(e(d_j,t_l),0)$ for each variables in $\mathcal{C}$ and $\mathcal{N}$. Accordingly, the most probable cluster configuration can be inferred by the most frequently sampled cluster of each object as follows:

$$\begin{aligned} c^\star(u_i) &= \arg\max_c s(u_i,c) \\ c^\star(d_j) &= \arg\max_c s(d_j,c) \\ c^\star(t_l) &= \arg\max_c s(t_l,c) \end{aligned} \tag{10}$$

5.2 Parametric Estimation

The model parameters of the proposed HRF include $\Gamma = \{\gamma_c\}, 1 \le q \le k$, for the multinomial distribution of each cluster. The parameters can be obtained by maximizing the likelihood of the model from the observed social media objects $\mathcal{X}$ as follows:

$$\Gamma^\star = \arg\max_\Gamma \log P(\mathcal{X}|\Gamma) = \arg\max_\Gamma \sum_{\mathcal{C},\mathcal{N}} \log P(\mathcal{C},\mathcal{N},\mathcal{X}|\Gamma) \tag{11}$$

It is intractable to directly optimize the above likelihood to obtain $\Gamma^\star$ since the marginalization of the hidden variables $\mathcal{C}$ and $\mathcal{N}$ involve an exponentially large number of terms. In this subsection, we use Expectation-Maximization (EM) [9] algorithm based on Gibbs Sampling results in the above subsection to obtain an efficient solution to the model parameters. The parameter Γ is first initialized by $\Gamma^{(0)}$. At each step τ, an expectation of the complete joint distribution with respect to the posterior $P(\mathcal{C},\mathcal{N}|\mathcal{X},\Gamma^{(\tau)})$ is computed as follows:

$$Q\left(\Gamma|\Gamma^{(\tau)}\right) = \sum_{\mathcal{C},\mathcal{N}} P(\mathcal{C},\mathcal{N}|\mathcal{X},\Gamma^{(\tau)}) \log P(\mathcal{C},\mathcal{N},\mathcal{X}|\Gamma)$$

The new parameters are then updated by maximizing this expectation as follows:

$$\Gamma^{(\tau+1)} = \arg\max_{C,N} Q\left(\Gamma|\Gamma^{(\tau)}\right)$$

It is not trivial to compute $Q\left(\Gamma|\Gamma^{(\tau)}\right)$. Fortunately, we can approximate this expectation by the sampled values in Algorithm 1. It is not difficult to see that the model parameters Γ only depend on the variables $c(d_j)$ regarding the social media object clusters. Then, we have:

$$\begin{aligned} & Q\left(\Gamma|\Gamma^{(\tau)}\right) \\ &= \sum_{d_j \in D} \sum_{c=1}^{k} \frac{s(d_j,c) \log P(x_j|c(d_j)=c,\gamma_c)}{\sum_{c=1}^{k} s(d_j,c)} + \text{Const} \\ &= \sum_{d_j \in D} \sum_{c=1}^{k} \frac{s(d_j,c) \sum_{v=1}^{q} x_{jv} \log \gamma_{cv}}{\sum_{c=1}^{k} s(d_j,c)} + \text{Const} \end{aligned}$$

All the constant terms in the parameter set are merged into Const term above. Then we have:

$$\gamma_{cv}^{(\tau+1)} \leftarrow \frac{\sum_{d_j \in D} s(d_j,c) x_{jv}}{\sum_{d_j \in D} \sum_{c=1}^{k} s(d_j,c) x_{jv}}$$

for $1 \le c \le k, 1 \le v \le q$. It is equivalent to soft-counting the mean of the number of feature occurrences for each cluster in $\mathcal{D}$ based on the sampling results. The above update rule can iterate until convergence.

6. EXPERIMENTS

In this section, we will compare the effectiveness of the *HRF* model with the other state-of-the-art multimedia clustering algorithms on the social media networks. In the following, we will describe the data sets, performance metrics and the experimental setup in detail.

Group Name	Favored Images in the Group	Top 10 tags in the group
Family		family portrait children fun girl love hmmlargeart baby tucson child
Street Art		streetart art street tag graf pochoir peinture urban bombe arosol
Folk Music		music folk folkmusic concert festival guitar lastfm live performance singer
Magic City		birmingham alabama al iphone iphoneography snow hdr stanroth urban downtown
Pet Portrait		dog pet cat portrait cute nikon animal puppy chien feline

Figure 3: Illustration of favored images and top 10 user tags in five user groups of *Flickr* social media network data set.

6.1 Data Set

We collected the following *Flickr* Social Network Data Set for evaluation purpose.

- *Flickr Social Network Data Set:* This data set contained 121 popular *Flickr* user groups, including "family", "auto" , "concerts", "pet portraits", "kids and nature", "street art," "wide party," "folk music," "magic city," "party favors", "British politics", "youth basketball", "fast food", "fancy dress party", and "great sky." These groups are collected using the keyword-based group search functionality provided by *Flickr*. The most popular tags were used as queries. This social media network has 13,826 users in these 121 groups, and each user can join more than one group. We note that users have the ability to mark their favored images in these groups. We use these favored images in order to create a graph of users in which the edges reflect an interest in the same image. In order to enable this, a total of 36,300 favored images were collected from *Flickr*. Since these images belong to user groups, we were able to use their group membership as the ground truth of the clustering. In order to construct the social media network, two users are linked by edges if they favor the same images. For each image, users also tag some keywords to describe its content. The user tags are stemmed and the stop words and meaningless keywords are removed. This results in 5,000 user tags in this *Flickr* data set. Figure 3 illustrates some favored images and top 10 user tags in five user groups of *Flickr* social media network data set. In general, the user tags provided a richly descriptive characterization of the underlying images as well as user sharing intention. These favored images and their associated user tags are collected to represent the edge content in social media graph. We also extract visual features in order to construct a multi-dimensional representation for image content. These include 8000 dimensional bag of words based on SIFT descriptions.

6.2 Performance Metrics

As mentioned in the previous section, each image is associated with cluster labels in addition to the content. These labels were used as the ground truth for measuring the effectiveness of the clustering process. Two metrics are used in the experiments. These were the *pairwise F-measure (P-*

Algorithm 2 Parametric Estimation for HRF model

input the sampled squences $s(u_i, \cdot), s(d_j, \cdot), s(t_l, \cdot), s(e(u_i, d_j), \cdot), s(e(d_j, t_l), \cdot)$ in Algorithm 1.

Uniformly initialize $\gamma_{cv}^{(0)} \leftarrow \frac{1}{q}$.

set $\tau \leftarrow 0$.

repeat

Sample a sequence of values from the current model, and count the occurrence of clusters for each object in $s(d_j, c)$.

Update the model parameters as

$$\gamma_{cv}^{(\tau+1)} \leftarrow \frac{\sum_{d_j \in D} s(d_j, c) x_{jv}}{\sum_{d_j \in D} \sum_{c=1}^{k} s(d_j, c) x_{jv}}$$

$\tau \leftarrow \tau + 1$.

until Convergence

output Model parameters Γ.

WF) and *average cluster purity (ACP)* respectively. These metrics are both supervised metrics, which are constructed with the use of the cluster ground truth (or class labels) collected in the data sets. Since clustering is an unsupervised problem, the ground truth information was not used during the clustering process. The class information about the communities is only used for evaluation purposes. This provides a robust evidentiary measure about the quality of the clustering.

Pairwise Precision, Recall and F-measure. We adopt the commonly used pairwise precision and recall measures for clustering algorithms [21], in order to create a meaningful measure. Let G denote the set of images that share one cluster class. Similarly, let H denote the set of images that are assigned to the same cluster by the algorithm. Then, we can compute the pairwise precision and recall as follows:

$$\text{pr} = \frac{|H \cap G|}{|H|}, \text{rc} = \frac{|H \cap G|}{|G|}$$

The afore-mentioned measures of precision and recall can be used in order to define the *pairwise F-measure* as follows:

$$\text{PWF} = \frac{2 \times \text{pr} \times \text{rc}}{\text{pr} + \text{rc}}$$

A higher value of the *pairwise F-measure (PWF)* suggests that the underlying clustering is of good quality.

Average Cluster Purity: The average cluster purity is computed as the average percentage of the *dominant community* in the different clusters. Formally, let $\mathcal{C} = \{C_1, \cdots, C_K\}$ be the k clusters determined by the algorithms. Let us assume that the number of points in C_i, are denoted by n_i. The corresponding set of n_i vertices is denoted by $\{v_{1,i}, \cdots, v_{n_i,i}\}$. Let $M_{l,i}$ denote the set of communities that $v_{l,i}$ truly belongs to in the ground truth of labels. Then, the *average cluster purity (ACP)* is defined as follows:

$$ACP = \frac{1}{k} \sum_{i=1}^{k} \sum_{l=1}^{n_i} \frac{\delta\left(dom_i \in M_{l,i}\right)}{n_i}$$

Here, $\delta(\cdot)$ is an indicator function, which indicates whether the dominant class dom_i of cluster C_i matches with at least one of the labels for a vertex.

6.3 Compared Algorithms

In order to validate the effectiveness of our algorithms, we used the following baselines:

- We used a *pure content-based approach* where we are simply clustering the documents on the edges, with the use of a clustering approach. We used the *LDA-WORD* [2] algorithms in order to cluster the content of the data sets.

- We used some *link-based techniques* in which we cluster the nodes using known structural methods. In particular, we tested with the use of the normalized cut (*NCUT*) [19] which is a spectral clustering algorithm. In this algorithm, we consider two types of links - the user favor link and tag links. Specifically, for user favor links, two images are considered to be linked if they are favored by the same user. For tag links, two images are linked together if they are annotated by the same tag. The link weights are defined by the number of times that two images are either co-favored or annotated by the same tag keywords. We apply *NCUT* on a graph that combines these two kinds of links. The link structure is used to partition the images without any content information.

- We also compare with the clustering algorithm with both social links and the image content. In particular, we tested with a graph theoretical clustering algorithm, which simultaneously integrates visual and textual features in a trigraph for efficient Web image clustering [17]. This algorithm is referred to as *Consistent Isoperimetric High-order Co-clustering (*CIHC*)*, and it uses the user tags as well as image content to partition the object in social media networks.

We describe some implementation details about the initialization of the *HRF* clustering model. The Gibbs sampling approach needs to be be initialized with an initial cluster configuration. For this purpose, we apply the k-means clustering algorithm on the content of *Flickr* images without any link structure. Such content-based initialization may affect the first few iterations of Gibbs sampling process. However, the link structure will be gradually incorporated into the modeling process after several iterations.

6.4 Results

The results for the different algorithms on the *Flickr* image network are illustrated in Table 1. The value of the parameter λ was fixed at 0.5. We present the results in terms of pairwise precision, recall and F-measure. The proposed *HRF* clustering algorithm outperforms the other algorithms, including the pure content and pure link-based algorithms, and also the algorithms which combine both link and content. This demonstrates that the combination of the different kinds of content and linkage information in a social media network provides more effective results. For example, *CIHC*, which combines both image content and the links to user tags in a trigraph, outperforms the *LDA-WORD* and *NCUT*. However, the *CIHC* algorithm is not quite as effective at incorporating different kinds of heterogeneous

Table 1: Comparison of different clustering algorithms on *Flickr* social media data set.

Algorithms		*Flickr* Social Media Data Set			
		precision	recall	PWF	ACP
Content	*LDA-WORD*	0.4350	0.5569	0.4885	0.4989
Link only	*NCUT*	0.4960	0.7189	0.5870	0.5237
content + tags	*CIHC*	0.5504	0.7484	0.6343	0.6403
Our approach	*HRF*	**0.6350**	**0.8526**	**0.7279**	**0.8296**

content such as the tags, the social structure and the object content. As the result, the *HRF* algorithm outperforms *CIHC* as well. An interesting observation is that the pure link-based clustering algorithm *NCUT* obtain better performances than pure content-based information. This suggests that the linkage information in the networks often contain more useful semantic information for the clustering process than the content.

Our algorithm, which combines not only content but also the context and social links, achieves the best performances. This is because in social media networks, in addition to the content, both tag and social links provide important network structural information for the discovering the image clusters. In addition to modeling content and links, the proposed *HRF* algorithm detects and removes the noisy links from the clustering process. This improves the robustness of clustering algorithm, which avoids the negative effect of noisy links. Therefore, the *HRF* algorithm is able to determine the useful clustering by separating out the noise from the useful clustering hints. This is particularly important in social media networks which are known to be noisy for mining purposes.

One interesting aspect of the clustering algorithm is that it outputs user clusters, tag clusters, and object cluster simultaneously. This can be very useful for a wide variety of applications, because the tag clusters can be used in order to determine the semantic interpretability of the images, and the user clusters can be leveraged in order to do group recommendations. This is particularly useful in cases where the multi-media objects are available on social sharing platforms. For example, Figure 4 also illustrates five image clusters and the associated tag clusters obtained by the *HRF* clustering algorithm. We can find that the images and associated tags are well clustered based on their content and descriptive tag words. It is evident that the tag keywords provide the descriptive theme in each cluster very well. For example, the images in the first cluster correspond to restaurants, and the corresponding tag keywords reflect this semantic theme well. Such objects can be recommended to the corresponding users in this cluster. Furthermore, new images which are tagged with the corresponding keywords can also be immediately recommended to the users in the cluster. Alternatively, the new users containing the appropriate frequent keywords in their comments or profile can be recommended the appropriate images. Thus, this clustering process not only improves the quality of the clustering, but also provides useful social and semantic interpretability which can be leveraged for a wide variety of applications.

6.5 Parameter Sensitivity

We also tested the sensitivity of the *HRF* model to different choices of the parameter λ. This is a particularly important parameter, because it regulates the importance of linkage information in the clustering process. For the purposes of our previously presented results in Table 1, we set the value of λ to 0.5, so that an approximately equal amount of importance is placed on linkage and content. Nevertheless, it is interesting to test how the quality of clustering is influenced by varying the value of this parameter. We illustrate the variation of the algorithm with λ in Figure 5. The value of λ is illustrated on the X-axis, and it varies from 0.1 to 0.8 with 0.1 as the step size. The *F-measure* and *cluster purity* measures of clustering quality are illustrated on the Y-axis in the different charts. It is evident from the results that when little link information is incorporated ($\lambda = 0.1$), the *HRF* model does not perform well on the data sets. However, as λ increases, more link information is combined together with content and it performs better. However, the effectiveness starts reducing after a certain point, because the use of a value of λ which is too large discounts the importance of the content information. This verifies that both content and link information do help in modeling the clustering structure. One observation is that the two different measures provide slightly different peaks in terms of clustering quality, though they both suggest robustness within a wide range of parameter values. We find that in the interval $[0.1, 0.8]$, the proposed *HRF* model consistently outperforms the other algorithms in terms of both the ACP and PWF measures.

Table 2: Comparison of computing time spent by different clustering algorithms on *Flickr* social media data set.

Algorithm	Computing Time
LDA-WORD	457 min
NCUT	375 min
CIHC	421 min
HRF	189 min

6.6 Computational Efficiency

Finally, we also compare the computing time spent by different algorithms for performing the clustering. The comparison is performed on a Server platform with Intel Xeon CPU 2.4 GHz and 33 GM physical memory. For the sake of fair comparison, we apply all the algorithm to compute 50 clusters. Table 2 reports the computational time of the different algorithms. We find that the algorithm *HRF* is much faster than the other algorithms. In fact the the *HRF* algorithm was *twice* as fast as the next fastest algorithm, corresponding to the *NCUT* method. This is in spite of the fact that the *NCUT* method works with only the link struc-

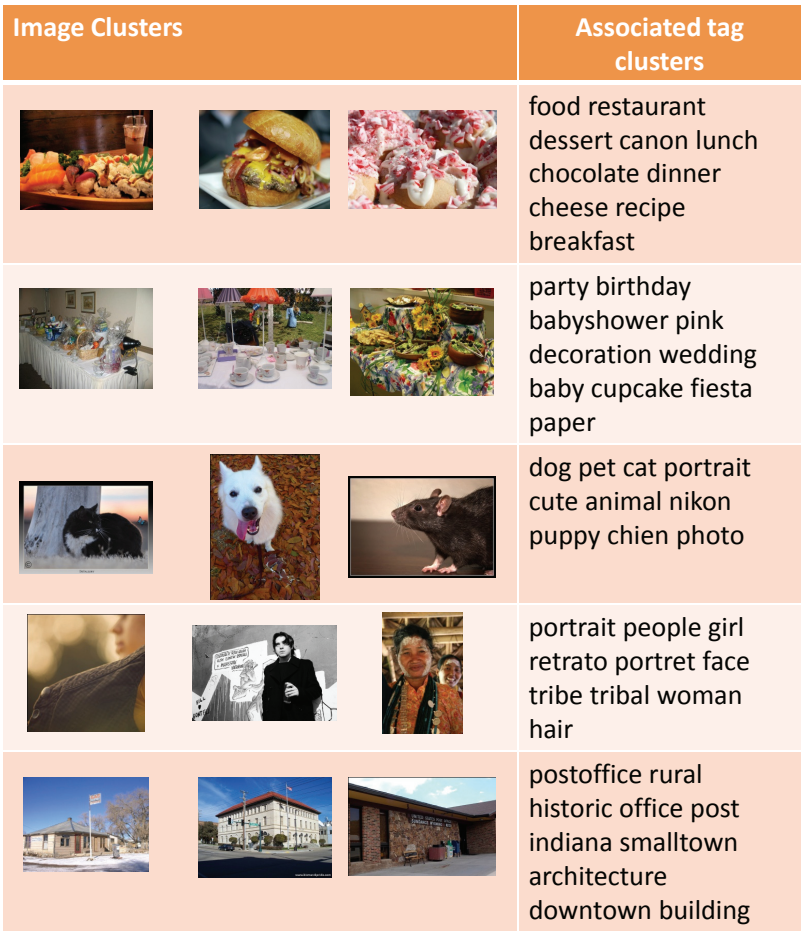


Figure 4: Illustration of five image and tag clusters obtained by the *HRF* clustering algorithm.

ture, whereas our approach uses both the content and links for clustering. The *LDA-WORD* algorithm was the slowest, and was about 2.5 times slower than the *HRF* method. The efficiency of our approach is because of our use of the Gibbs sampling strategy, which is able to provide robust results for samples of reasonable size. Thus, the approach presented in this paper provides an effective tradeoff between the quality and efficiency of the results.

7. CONCLUSIONS AND SUMMARY

In this paper, we present a robust clustering algorithm on social media networks based on heterogeneous random fields. We use a combination of linkage information and social cues in order to perform the clustering. These are used to infer the highest probability cluster configuration in an efficient way. The algorithm is able to explicitly detect the noisy links, which is particularly important in the noisy social media scenario. We present experimental results, which network demonstrate the effectiveness of our algorithm compared to other state-of-the-art methods.

Acknowledgments

Research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on. This work was also supported in part by National Science Foundation (NSF) Grant IIS-1144111. Guo-Jun Qi was supported in part by an IBM fellowship.

8. REFERENCES

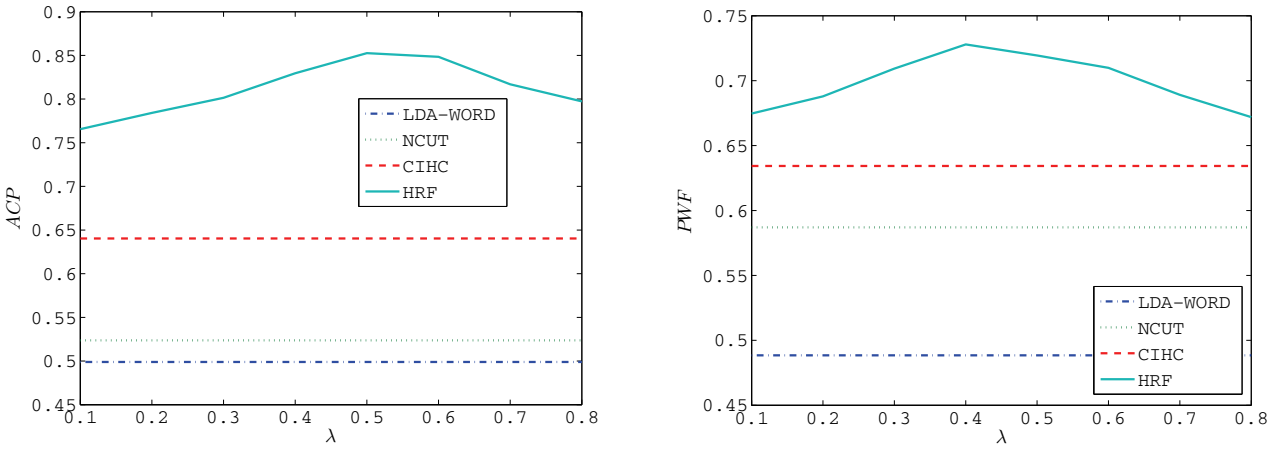
[1] K. Barnard, P. Duygulu, and D. A. Forsyth. Clustering art. In *CVPR Conference*, 2001.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, (3):993–1022, January 2003.

[3] D. Cai, X. He, Z. Li, W.-Y. Ma, and J.-R. Wen. Hierarchical clustering of www image search results using visual, textual, and link information. In *ACM Multimedia Conference*, 2004.

[4] D. Chakrabarti, R. Kumar, and A. Tomkins. Evolutionary clustering. In *KDD*, 2006.

[5] Y. Chen, J. Z. Wang, and R. Krovetz. Clue: Cluster-based retrieval of images by unsupervised



(a) (b)

Figure 5: Illustration of performance sensitivity with respect to the parameter λ for the *HRF* model in terms of pairwise F-measure (PWF) in subfigure (a) and cluster purity (ACP) in subfigure (b).

learning. In *IEEE Transactions on Image Processing, 14(8)*, 2005.

[6] Y. Chi, X. Song, D. Zhou, K. Hino, and B. Tseng. Evolutionary spectral clustering by incorporating temporal smoothness. In *KDD*, 2007.

[7] A. Clauset, M. E. J. Newman, and C. Moore. Finding community structure in very large networks. In *Phys.Rev. E 70, 066111*, 2004.

[8] D. Cutting, D. Karger, J. Pedersen, and J. Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. In *Proceedings of the SIGIR*, 1992.

[9] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum-likelihood from incomplete data via the em algorithm. *J. Royal Statist. Soc.*, B(39):1–38, 1977.

[10] B. Gao, T.-Y. Liu, T. Qin, X. Zheng, Q.-S. Cheng, and W.-Y. Ma. Web image clustering by consistent utilization of visual features and surrounding texts. In *ACM Multimedia Conference*, 2005.

[11] A. E. Gelfand and A. F. M. Smith. Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, (85(410)):398ÍC409, 1990.

[12] D. Gibson, R. Kumar, and A. Tomkins. Discovering large dense subgraphs in massive graphs. In *VLDB*, 2005.

[13] J. Goldberger, S. Gordon, and H. Greenspan. Unsupervised image-set clustering using an information theoretic framework. In *IEEE Transactions on Image Processing, 15*, 2006.

[14] A. Jain and R. C. Dubes. Algorithms for clustering data. In *Prentice Hall*, 1988.

[15] Y.-R. Lin, J. Sun, P. Castro, R. B. Konuru, H. Sundaram, and A. Kelliher. Extracting community structure through relational hypergraphs. In *WWW Conference*, 2009.

[16] G. Qiu. Image ad feature co-clustering. In *ICPR Conference*, 2004.

[17] M. Rege, M. Dong, and J. Hua. Graph theoretical framework for simultaneously integrating visual and textual features for efficient web image clustering. In *Proc. of International Conference on World Wide Web*, 2008.

[18] V. Satuluri and S. Parthasarathy. Scalable graph clustering using stochastic flows: applications to community discovery. In *KDD Conference*, 2009.

[19] J. Shi and J. Malik. Normalized cuts and image segmentation, August 2000.

[20] C. Silverstein and J. Pedersen. Almost-constant time clustering of arbitrary corpus sets. In *Proceedings of the SIGIR*, 1997.

[21] T. Yang, R. Jin, Y. Chi, and S. Zhu. Combining link and content for community detection: a discriminative approach. In *ACM KDD Conference*, 2009.

[22] Y. Zhou, H. Cheng, and J. X. Yu. Graph clustering based on structural/attribute similarities. In *VLDB Conference*, 2009.

[23] S. Zhu, K. Yu, Y. Chi, and Y. Gong. Combining content and link for classification using matrix factorization. In *Proceedings of the 30th ACM SIGIR Conference on Research and Development in Information Retrieval*, 2007.