

Exploiting Latent Relevance for Relational Learning of Ubiquitous Things

Lina Yao
School of Computer Science
The University of Adelaide
Adelaide, SA 5005, Australia
lina.yao@adelaide.edu.au

Quan Z. Sheng
School of Computer Science
The University of Adelaide
Adelaide, SA 5005, Australia
qsheng@cs.adelaide.edu.au

ABSTRACT

With recent advances in radio-frequency identification (RFID), wireless sensor networks, and Web services, physical things are becoming an integral part of the emerging ubiquitous Web. While this integration offers many exciting opportunities such as efficient supply chains and improved environmental monitoring, it also presents many significant challenges. One such challenge lies in how to classify, discover, and manage ubiquitous things, which is critical for efficient and effective object search, recommendation, and composition. In this paper, we focus on automatically classifying ubiquitous things into manageable semantic category labels by exploiting the information hidden in interactions between users and ubiquitous things. We develop a novel approach to extract latent relevances by building a relational network of ubiquitous things (RNUbiT) where similar things are linked via virtual edges according to their latent relevances. A discriminative learning algorithm is also developed to automatically determine category labels for ubiquitous things. We conducted experiments using real-world data and the experimental results demonstrate the feasibility and validity of our proposed approach.

Categories and Subject Descriptors

H.3.5 [Information Storage and Retrieval]: Online Information Services; H.4.0 [Information Systems]: General

Keywords

Ubiquitous things discovery, Web of Things, multi-label classification, relational learning, modularity

1. INTRODUCTION

About two decades after Mark Weiser wrote his seminal article [3], we are one step closer to his vision of “ubiquitous computing” where computing power becomes invisibly integrated into the world around us and accessed through intelligent interfaces. The main driver lies in recent advances in identification technologies such as radio frequency identification (RFID), wireless sensors, and nanotechnology, which make processing power available in smaller

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

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

and smaller packages that can interact and connect. Indeed, our world is slowly evolving into an environment where everyday objects such as buildings, sidewalks, and commodities are readable, recognizable, addressable, and even controllable [10].

While it is well understood that such a ubiquitous environment offers the capability of integrating the information from both the physical world and the virtual one, which creates tremendous business opportunities such as efficient supply chains, independent living of elderly persons, and improved environmental monitoring, it also presents significant challenges [3]. With many things connected and interacted over the Web, there is an urgent need to efficiently index, organize, and manage these things for object search, recommendation, and effectively reveal interesting patterns from the things. Classifying ubiquitous things into manipulable semantic categories is an essential task to realize these applications.

Although classification of Web objects has been actively investigated in the last few years, especially Web page classification [9], these studies mostly focus on homogeneous objects and their classification algorithms are designed for some specific types of data. Classification of ubiquitous things is a much more challenging task due to the specific characteristics of things on the Web. Firstly, things cannot be easily represented in a meaningful feature space. They usually only have very short textual descriptions [5], which cannot provide enough semantic features. Secondly, ubiquitous things are normally diverse and noisy (e.g., some things have meaningful descriptions while many others do not) [3]. As a result, it becomes quite challenging to discover explicitly the relationships between heterogeneous things. Finally, correlations among things are not obvious and difficult to discover. This is due to the fact that things often exist in isolate settings and interconnections between them are usually limited.

The problem targeted in this paper can be formulated as predicting appropriate categories for a given thing on the Web. In general, there are two types of things: *labeled* things and *unlabeled* things. The goal of our work is to predict category label for unlabeled things by mining the latent relevance between things. The relevance between things are *implicit* and can be captured by exploring regularities of users' interactions with similar things. This task can be conducted by using a two-stage approach. Firstly, we propose to derive the latent relevances of things based on the log history records from three dimensions: i) *user-thing* interactions that record who has used the things; ii) *time-thing* interactions that record when the things are used; and iii) *location-thing* interactions that record where the things are used. Secondly, we adopt the relational learning technique to predict category labels for unlabeled things. We summarize our main contributions as the following:

- We study the problem of managing ubiquitous things, which have specific characteristics (e.g., short descriptions, diverse,

and noisy). We propose to investigate users' interactions with things from three different aspects: usage history, temporal, and geographical influence. Accordingly, we develop three graph presentations that appropriate corresponding relationships from users' interactions with things. These graphs lay the foundation for uncovering the latent relevance among ubiquitous things.

- We formulate the task of classification of ubiquitous things as a multi-label classification problem. We propose to construct a relational network of ubiquitous things and extract latent features for discriminative learning, which automatically determine better aligned category labels. In particular, we focus on community mining of ubiquitous things, which involves two main techniques. The Random Walk with Restart [13] is used to calculate the relevance of things and the modularity method [7] is extended to discover communities.
- We establish a testing environment where things are tagged by RFID and exposed on the Web using RESTful Web services. Using the data collected from this environment, we conduct experimental studies to demonstrate the feasibility of our proposed approach.

The remainder of the paper is organized as follows. Section 2 introduces the concept and construction of the relational network of things. Section 3 proposes a modularity-based method that applies to the relational network of things to predict category of unlabeled things, which eventually classifies things into the corresponding categories. Section 4 reports the experimental results and Section 5 discusses the related work and gives some concluding remarks.

2. RELATIONAL NETWORK OF UBIQUITOUS THINGS

In this section, we introduce the three graphs, and the relational network of ubiquitous things.

2.1 Relevance Graph Building

To extract features from latent relevance among similar things, we develop a novel algorithm that constructs a relational network of ubiquitous things (RNUbiT), which captures the relevance between things by considering three aspects: *user* (i.e., *who* uses the thing), *time* (i.e., *when* the thing is used), and *location* (i.e., *where* the thing is used). Three graphs, namely user-thing graph, *time-thing* graph, and *location-thing* graph are developed accordingly.

User-Thing Graph. A user-thing graph consists of users and things connected in accordance with the log history. Let $h(u_i, t_j, ts_k, l_m) \in \mathcal{H}$ denote a history record indicating that user u_i used thing (resource) t_j at time stamp ts_k in location l_m , where \mathcal{H} is the set of history records. Formally, a user-thing graph (see Figure 1 (a)), denoted by $\mathcal{G}_u = (\mathcal{U}, \mathcal{T}, \mathcal{I}_u, \mathcal{F})$, is an augmented bipartite graph. \mathcal{U} and \mathcal{T} are the sets of users and things respectively. $\mathcal{I}_u = \{(u, t) : (u, t) \in \mathcal{U} \times \mathcal{T}\}$ denotes the things accessed by user u , $\mathcal{F} = \{(u, u') : (u, u') \in \mathcal{U} \times \mathcal{U}\}$ denote the users' friendships. In this graph, each edge between users and things is associated with a weight $w_{i,j}$, referring to the frequency that thing t_j is used by user u_i .

To obtain the relevance r_{t_i, t_j}^u between a pair of things from the user-thing graph, we adopt the biased random walk on this augmented bipartite graph. Given a node on this graph, random walk is performed by randomly following one of its links to another node on the augment graph based on the transition probabilities of these links. After convergence, the stationary states probabilities for each

pair of nodes on this graph can be obtained by recursively implementing the random walk. Four transition probabilities matrices in the graph are: i) user-user transition $\mathcal{P}(u'|u)$, ii) thing-thing transition $\mathcal{P}(t'|t)$, iii) user-thing transition $\mathcal{P}(t|u)$, and iv) user-thing's transpose $\mathcal{P}(u|t)$. $\mathcal{P}(t|u)$ and its transpose matrix should be proportional to the times of a thing being used.

It should be noted that our user-thing graph also considers interactions between users (i.e., social links). We treat relationships between users as augmenting the bipartite graph with social links (i.e., \mathcal{F} in the user-thing graph) among users. The user-user social links \mathcal{F} reflect the homophily meaning that similar users may have similar interests. Therefore, instead of selecting the link uniformly at random, we set the selection as biased by the *similarity* of users. In our work, the user-user transition probability is calculated using:

$$\mathcal{P}(u|u') \propto \frac{e^{\alpha \cos(b(u), b(u'))}}{\sum_{u'' \in \Omega(u)} e^{\alpha \cos(b(u), b(u''))}} \quad (1)$$

where $\cos(b(u), b(u')) = \frac{b(u) \cdot b(u')}{|b(u)| |b(u')|}$. $\Omega(u)$ is the set of the user u 's friends (i.e., $u' \in \Omega(u)$), $b(u)$ is the binary vector of things used by user u , and α is a parameter that reflects the preference for transitioning to a user who used the same things [11].

Time-Thing Graph. A time-thing graph (see Figure 1 (b)), denoted by $\mathcal{G}_t = (\mathcal{T}\mathcal{S}, \mathcal{T}, \mathcal{I}_t)$, is a bipartite graph where \mathcal{T} and $\mathcal{T}\mathcal{S}$ are the sets of things and times respectively. $\mathcal{I}_t = \{(t, ts) : (t, ts) \in \mathcal{T} \times \mathcal{T}\mathcal{S}\}$ denotes the things accessed at different times. In this graph, each edge is associated with a weight $w_{j,k}$, referring the frequency thing t_j is accessed at time ts_k .

Similarly, there are four transition matrices: i) time-time transition $\mathcal{P}(ts'|ts)$, ii) thing-thing transition $\mathcal{P}(t'|t)$, iii) time-thing transition $\mathcal{P}(ts|t)$, and iv) time-thing's transpose $\mathcal{P}(t|ts)$. The same process is implemented on the time-thing graph to derive the relevance between pair of things, r_{t_i, t_j}^t from temporal dimension.

Location-Thing Graph. A location-thing graph (see Figure 1 (c)) reflects the spatial information of the usage of things. We argue in this paper that *geographical influence* to user activities cannot be ignored, i.e., a user tends to interact with things nearby rather than the ones distant [1]. For example, if a user is at her office, she has higher probability of using office facilities such as telephone, printer, and seminar rooms. Formally, a location-thing graph, denoted by $\mathcal{G}_l = (\mathcal{L}, \mathcal{T}, \mathcal{I}_l)$, is a bipartite graph where \mathcal{T} and \mathcal{L} are the sets of things and locations respectively. $\mathcal{I}_l = \{(t, l) : (t, l) \in \mathcal{T} \times \mathcal{L}\}$ denotes that the things are accessed at different user locations. In this graph, each edge is associated with a weight $w_{j,m}$, indicating how often thing t_j is accessed at location l_m .

There are four transition matrices including: i) location-location transition $\mathcal{P}(l'|l)$, ii) thing-thing transition $\mathcal{P}(t'|t)$, iii) location-thing transition $\mathcal{P}(l|t)$, and iv) location-thing's transpose $\mathcal{P}(t|l)$. Similarly, the relevance r_{t_i, t_j}^l between pair of things from spatial dimension can be derived after applying the same process on the location-thing graph.

As we mentioned before, our model also considers the geographical influence to users' interactions with things. Inspired by the recent work on social networks (e.g., [1]) and location-based services (e.g., [16]), we propose to use *power law distribution* to model a thing's usage probability to the physical distance between the thing and the user, as the following:

$$z = \theta \cdot x_1^b + (1 - \theta) \cdot x_2^d \quad (2)$$

where b and d are parameters of a power law distribution, x_1 and x_2 denotes the distance between user u_i and thing t_j and the fre-

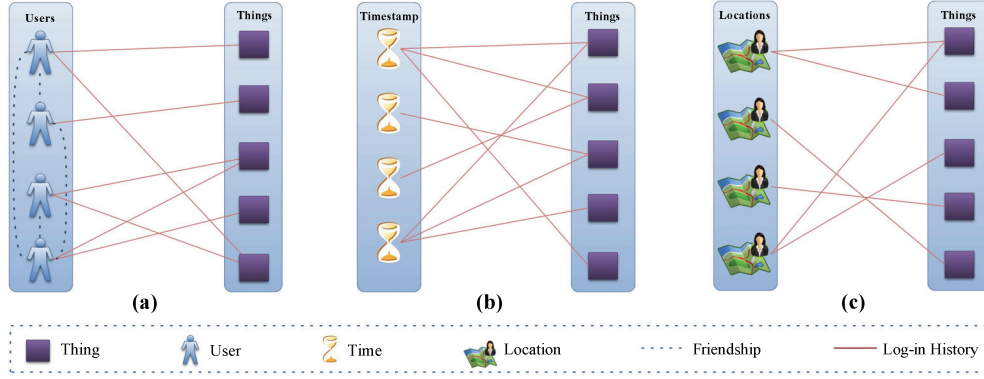


Figure 1: Three graph representations: (a) user-thing graph, (b) time-thing graph, and (c) location-thing graph

quency of this usage, respectively. θ is the weight for distance and frequency. z is the thing’s usage probability.

Equation 2 can be transformed in log-log scale to fit a linear model, into the following equation:

$$\log z = \log(\theta) + b \log x_1 + \log(1 - \theta) + d \log x_2 \quad (3)$$

Let $z' = \log z$, $x'_1 = \log x_1$, $x'_2 = \log x_2$, $\log(\theta)$ as w_0 , b as w_1 , $\log(1 - \theta)$ as w_2 and d as w_3 , we can have:

$$z' = \mathbf{w}^T \mathbf{x} \quad (4)$$

where w_0 , w_1 , w_2 , and w_3 are the linear coefficients, which are collectively denoted by \mathbf{w} , and $\mathbf{x} = \{x'_1; x'_2\}$. In order to avoid over-fitting, we approach the weight coefficients (i.e., \mathbf{w}) by least square error method and add a penalty term to discourage the coefficients from reaching large values, as the following:

$$\tilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{i=1}^n (z'_i - z''_i)^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2 \quad (5)$$

where $E(\mathbf{w})$ denotes the loss function, n represents the cardinality of input dataset, z''_i is the ground truth corresponding to z'_i , and λ is the regularization term. The optimal values of power law parameters (i.e., b , d in Equation 2) form the setting that minimizes the loss function $E(\mathbf{w})$. In our work, we use stochastic gradient descent to solve this optimization problem. Interested readers are referred to [2] for more details.

2.2 Relational Network of Ubiquitous Things

A relational network of ubiquitous things (RNUbiT) is constructed in a way that each thing is connected to things with top-k relevance values. The relevance between pair of things t_i and t_j in RNUbiT, denoted by r_{t_i, t_j} , is calculated as:

$$r_{t_i, t_j} = \gamma r_{t_i, t_j}^u + \epsilon r_{t_i, t_j}^t + \eta r_{t_i, t_j}^l \quad (6)$$

where $\gamma + \epsilon + \eta = 1$, $\forall \gamma, \epsilon, \eta \in [0, 1]$. γ , ϵ and η are parameters that can be used to control preferences over three different aspects (i.e., user, time, and location). An RNUbiT is denoted by $\mathcal{G}_{rnt} = (\mathcal{T}, \mathcal{E}, \mathcal{W})$. For each thing $t_i \in \mathcal{T}$, let \mathcal{T}_i denote the set of related things to t_i . $\mathcal{E} = \{e(x, i) | \forall t_i \in \mathcal{T}, t_x \in \mathcal{T}_i^k\}$, where $e(x, i)$ is an edge from t_x to t_i . Each edge is associated with a weight $w_{t_x, t_i} \in \mathcal{W}$ with the relevance value.

3. RELATIONAL LEARNING

Things usually hold multiple relations. For instance, a thing might be shared among its owner, owner’s friends, co-workers, or

family members. It might be connected to other things based on functionality or non-functionality attributes. Detecting such relations is related to the task of *community detection* [7].

There are multiple methodologies to learn the communities from a graph such as collective classification and latent space model [4, 12]. In our work, we adopt the *modularity-based* methods to unveil the implicit structures of the things network. Briefly, modularity is a metric proposed by Newman et al. [7] to evaluate the goodness of a partition of undirected graphs. The reason that we choose this method is that modularity has been shown to be an effective quantity to measure community structure in many complex networks.

Modularity \mathcal{Q} is like a statistical test that the null model is a uniform random graph model, where one vertex connects to others with uniform probability. It is a measure of how far the interaction deviates from a uniform random graph with the same degree distribution. Modularity is defined as:

$$\mathcal{Q} = \frac{1}{2m} \sum_{ij} \left[\mathcal{A}_{ij} - \frac{d_i d_j}{2m} \right] \delta(s_i, s_j) \quad (7)$$

Where \mathcal{A}_{ij} is the adjacent matrix on the graph, m is the number of edges of the matrix, d_i and d_j denote the degree of vertex i and out-degree of vertex j , and $\delta(s_i, s_j)$ are the Kronecker delta function that takes the value 1 if node t_i and t_j belong to the same community, 0 otherwise. A larger modularity \mathcal{Q} indicates denser within-group interaction. The modularity-based algorithm aims to find a community structure such that \mathcal{Q} is maximized. Newman [7] proposes an efficient solution by reformulating \mathcal{Q} as:

$$\mathcal{Q} = \frac{1}{2m} \mathcal{S}^T \mathbf{B} \mathcal{S} \quad (8)$$

where \mathcal{S} is the binary matrix indicating which community each node belongs to. \mathbf{B} is the modularity function with entries are as follows:

$$\mathcal{B}_{ij} = \mathcal{A}_{ij} - \frac{d_i d_j}{2m} \quad (9)$$

However, since RNUbiT is weighted and directed graph, we need to make some modification on \mathcal{Q} to solve it. This involves two steps. In the first step, we extend \mathcal{B} to directed graph. Based on [6], we rewrite the modularity matrix \mathcal{B} as the following:

$$\mathcal{B}'_{ij} = \mathcal{A}_{ij} - \frac{d_i^{in} d_j^{out}}{2m} \quad (10)$$

where $d_i^{in} d_j^{out}$ are the in-degrees and out-degrees of all the nodes on the RNUbiT graph.

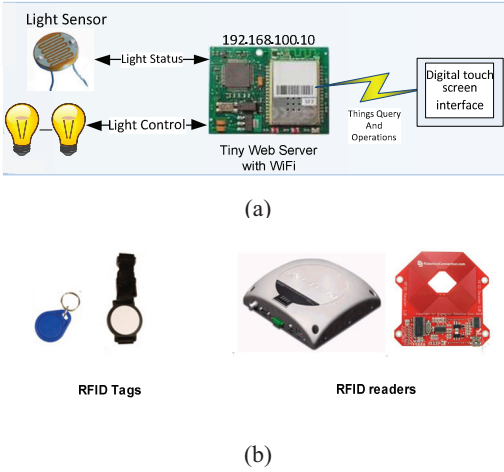


Figure 2: (a) An implemented ubiquitous things: RFID tagged lights; (b) RFID tags and readers used in the implementation

In the second step, we extend \mathcal{B}' to weighted graphs. To do so, we do the further modification based on Equation 10. According to [7], it can be rewritten further as below:

$$\mathcal{B}''_{ij} = \mathcal{W}_{ij} - \frac{w_i^{in} w_j^{out}}{2m} \quad (11)$$

where \mathcal{W}_{ij} is the sum of weights of all edges in RNUbiT graph replacing the adjacency matrix \mathcal{A} , w_i^{in} and w_j^{out} are the sum of the weights of incoming edges adjacent to vertex t_i and the outgoing edges adjacent to vertex t_j on the RNUbiT graph respectively.

After these two steps, it should be noted that different from undirected situation, \mathbf{B}'' is not symmetric. To use the spectral optimization method proposed by Newman in [7], we restore symmetry by adding \mathbf{B}'' to its own transpose, thereby the new \mathcal{Q}_{new} is:

$$\mathcal{Q}_{new} = \frac{1}{4m} \mathcal{S}^T (\mathbf{B}'' + \mathbf{B}''^T) \mathcal{S} \quad (12)$$

We then is able to calculate all the eigenvectors corresponding to the top- n positive eigenvalue of $\mathbf{B}'' + \mathbf{B}''^T$ and assign communities based on the elements of the eigenvector.

After obtaining the latent features of things based on the modularity technique, we feed them into Support Vector Machine (SVM) [12] to do the supervised learning. It is noted that these latent features can be easily combined with other features of things (e.g., textual features and contextual features) to enhance the classification performance.

4. IMPLEMENTATION AND EXPERIMENTS

4.1 Implementation

We set up a testing environment that consists of several different physical places including one author’s home (e.g., kitchen, living room). Things (e.g., desktop computers, backpacks, TV, coffee machine) are tagged using radio-frequency identification (RFID) technology. RFID tags are captured by two fixed RFID readers and ten portable RFID readers (worn by volunteers when interacting with things). Figure 2 (a) shows one application that we implemented to detect and control lights and Figure 2 (b) shows the RFID devices that we used in the implementation.

When users interact with things, RFID events and data are captured by RFID readers. These raw data have to be preprocessed (e.g., cleaning, transformation, integration) before storing at the

repository. This task greatly benefits from our four years extensive experience in a large RFID research project [14]. In our implementation, things are exposed on the Web using RESTful Web services, which can be discovered and accessed from a Web-based interface. To determine user locations, we also develop a module that implements an RFID-based localization algorithm [8].

4.2 Experimental Study

In this section, we report two experiments. The first experiment studies the impact of different training sets on the performance of our approach. The second experiment studies the impacts on coupling additional information with the derived latent features.

In our experiments, we tagged 127 different things in six categories. This dataset serves as the ground-truth dataset in our experiments. Ten volunteers participated in the data collection phase by interacting with RFID tagged things for a period of four months, generating 20,179 records on the interactions of the ubiquitous things tagged in the experiments. We adopt the widely used performance metrics to evaluate the performance of our proposed approach: *hamming loss*, *one-error*, *coverage*, and *average precision* [16].

Impact of Training Ratio. This experiment studies the impact of different training ratio on the performance of our approach. We randomly removed the category tags of a certain percentage, ranging over 10% to 90%, of things from each category of the ground-truth dataset, which were used to test our approach while the rest were used as the training set. This process was repeated 10 times and the average results were recorded. Figure 3 shows the results of the home category. The other categories have the similar results.

From the figure we can see that better performance is achieved with higher training sets. Interestingly, there are no significant performance improvement when the training sets are larger than 50%.

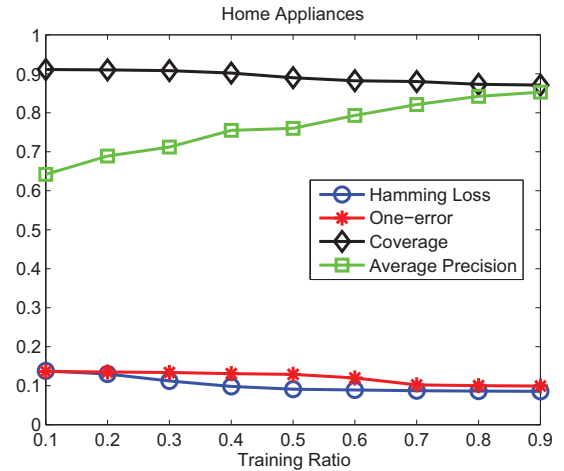


Figure 3: Impacts of training ratio

Coupling with Other Features. Our approach is feature-based in the sense that we consider the category probability of ubiquitous things that are derived from our algorithm. If other explicit information is available about the things (e.g., textual description), it would be useful to couple this information with the derived latent features. We expect that this combination will lead to a more accurate classification performance.

In our experiment, we consider the textual description of things as another explicit feature. It is worth reminding that we expose things as RESTful Web services and each thing has a brief webpage describing its basic information. By analyzing the textual descrip-

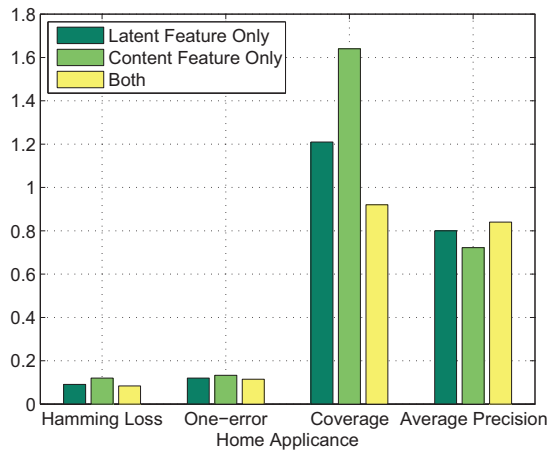


Figure 4: Combine with other features

tion, it is possible to extract the most common terms that represent the corresponding thing. We exploit Term Frequency/Inverse Document Frequency (TF/IDF)—a common method in IR for generating a robust set of representative keywords from a corpus of documents—to analyze things’ descriptions. Figure 4 depicts the performances of three scenarios: i) latent features only, ii) content only, and iii) the combination of the both. As we mentioned earlier, description of things are normally short and noisy, it is therefore not surprising that the performance based on content only is even worse than the latent feature-based approach. On the other hand, by combining the two together, the performance is increased, as shown in Figure 4. It should be noted that for the metrics of hamming-loss, one-error, and coverage, higher value means worse performance.

5. DISCUSSION AND CONCLUSION

The emerging “Web of Things” will bridge the physical and digital worlds together, and ubiquitous things will become an integral part of our daily lives. This paper focuses on efficient management of large number of things on the Web. In particular, we propose classifying ubiquitous things using *relational learning* methods based on a two-stage approach: i) deriving the latent relevances of things by exploiting the information of log-in records in terms of user dimension, temporal dimension, and spatial dimension, which are used to build a relational network of things, and ii) applying the modularity-optimization technique to classify things based on the generated relational network of things.

Relational learning refers to the classification when objects or entities present multiple relations [12]. One main technique on relational learning is based on Markov assumption, where the labels of a node in a relational network are determined by the labels of nodes in its neighborhood. Collective inference [16] and semi-supervised learning on graphs [17] work on this assumption, which is constructed based on the relational features of labeled data, followed by an iterative process (e.g., relaxation labeling method) to determine the class labels for the unlabeled data. In [16], Ye et al. apply this methodology in Location-based Social Network for deriving the label probability for places. The authors use the collective classification method that learns the labels from the neighborhood, which only includes the nodes which hold the top-k relevance with the prediction node. In [15], Xu et al. propose a non-parametric infinite hidden relational model to capture the autocorrelation. However, both approaches, unfortunately, are not suitable for networks where the number of ubiquitous things might be large

due to their complexity and high computational cost for inference. Moreover, data collected from ubiquitous things are expected to be noisy. In this paper, we extend this model to the relational network of things where a log-in record not only indicates user and temporal information, but also location information. As a result, a better performance in deriving latent features from the relational network of things can be achieved.

We view the work presented in this paper as a first step towards efficient management of ubiquitous things. Our ongoing work includes further evaluating the approach in a large-scale environment. Another challenge lies on handling the dynamic nature of ubiquitous things. How to efficiently update the relational model in large scale environments remains a challenge.

6. REFERENCES

- [1] L. Backstrom et al. Find Me if You Can: Improving Geographical Prediction with Social and Spatial Proximity. In *Proc. of the 19th Intl. World Wide Web Conf.*, 2010.
- [2] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- [3] A. Ferscha. 20 Years Past Weiser: What’s Next? *IEEE Pervasive Computing*, 11(1):52–60, 2012.
- [4] D. Jensen et al. Why Collective Inference Improves Relational Classification. In *Proc. of the 10th Intl. Conf. on Knowledge Discovery and Data Mining*, 2004.
- [5] T. Kindberg and et al. People, Places, Things: Web Presence for the Real World. *Mobile Networks and Applications*, 7(5):365–376, 2002.
- [6] E. Leicht and M. Newman. Community structure in directed networks. *Physical Review Letters*, 100(11), 2008.
- [7] M. Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E*, 74(3), 2006.
- [8] L. Ni, Y. Liu, Y. Lau, and A. Patil. LANDMARC: Indoor Location Sensing Using Active RFID. *Wireless Networks*, 10(6):701–710, 2004.
- [9] X. Qi and B. Davison. Web Page Classification: Features and Algorithms. *ACM Computing Surveys*, 41(2):1–31, 2009.
- [10] Q. Sheng et al. Ubiquitous RFID: Where are we? *Information Systems Frontiers*, 12(5):485–490, 2010.
- [11] A. P. Singh and et al. Recommendations Using Absorbing Random Walks. In *Proc. of the North East Student Colloquium on Artificial Intelligence*, 2007.
- [12] L. Tang and H. Liu. Relational Learning via Latent Social Dimensions. In *Proc. of the 15th Intl. Conf. on Knowledge Discovery and Data Mining*, 2009.
- [13] H. Tong, C. Faloutsos, and J. Pan. Random walk with restart: fast solutions and applications. *Knowledge and Information Systems*, 14(3):327–346, 2008.
- [14] Y. Wu, Q. Z. Sheng, and D. C. Ranasinghe. Facilitating Efficient Object Tracking in Large-Scale Traceability Networks. *The Computer Journal (Oxford)*, 54(12):2053–2071, 2011.
- [15] Z. Xu et al. Nonparametric Relational Learning for Social Network Analysis. In *Proc. of the 14th Intl. Conf. on Knowledge Discovery and Data Mining*, 2008.
- [16] M. Ye et al. On the Semantic Annotation of Places in Location-based Social Networks. In *Proc. of the 17th Intl. Conf. on Knowledge Discovery and Data Mining*, 2011.
- [17] X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions. In *Proc. of the 20th Intl. Conf. on Machine Learning*, 2003.