Pharos: Social Map-Based Recommendation for Content-Centric Social Websites

Wentao Zheng† Michelle Zhou‡ Shiwan Zhao‡ Quan Yuan† Xiatian Zhang† Changyan Chi†
†IBM Research – China ‡IBM Research – Almaden

ABSTRACT
Recommendation technologies are widely used in online social websites (e.g., forums and blogs) to help users locate their interests among overwhelming amounts of information. However, it is difficult to make effective recommendations for new users (a.k.a. the cold start problem) due to a lack of user information (e.g., preferences and interests). Furthermore, the complexity of recommendation algorithms may not be easily explained, leaving users with trust issues in recommendation results. To tackle the above two challenges, we are building Pharos, a social map-based recommender system. A social map summarizes users’ content-related social behavior (e.g., reading, writing, and commenting) over time as a set of latent communities. Each community describes the content being discussed and the people involved. Discovering, ranking, and recommending “popular” latent communities on a social map, Pharos enables new users to grasp the dynamics of a social website, alleviating the cold start problem. In addition, the map can also be used as a context for making and explaining recommendations about people and content. We have deployed Pharos internally and the preliminary evaluation shows the usefulness of Pharos.

Author Keywords
Cold start, explanation, recommender systems, social websites, trust

ACM Classification Keywords
H.3.3 Information Search and Retrieval: Information filtering; H.5.3 Information Interfaces and Presentations: Group and Organization Interfaces—Collaborative computing

INTRODUCTION
In recent years, content-centric social websites (e.g., forums, wikis, and blogs) have flourished with an exponential growth of user-generated information. It becomes increasingly more difficult for people to navigate these sites and locate desired information. Thus, researchers have developed recommendation technologies to help people better find desired information at online social websites [4, 5]. However, there are two prevalent challenges when building such a recommender system. First, it is difficult to make effective recommendations if a system knows little about its users or the items to be recommended. This is also known as the cold start problem [10, 11]. In the case of a social website, a recommender system knows little about a user who is new to the website or has few connections to others. As a result, it would be difficult for the system to guess what a user is looking for and make effective recommendations. Second, it is difficult to explain recommendation rationales to end users to make the recommendation more trustworthy. A recommender system usually utilizes complex algorithms or inferences to make recommendations. It is difficult for average users to interpret and comprehend the process and entrust themselves to the recommended results [5, 7].

To address the above two challenges, we are building Pharos, a social map-based recommender system. Here, we use the term social map to refer to a dynamically generated raider’s map 1 of a content-centric social website. Such a social map summarizes users’ content-related social behavior (e.g., reading, writing, and commenting) over time as a set of “latent communities” (Figure 1-a). Each latent community characterizes the implicit connections among a set of users and the content they generated. It thus consists of two parts, explaining what is being talked about (content summarized in green) and who are involved (people summarized in blue). Based on the generated social map, Pharos then uses a set of criteria (e.g., people’s social status and content popularity) to recommend “hot” communities, “hot” content or people in each community. For example, Figure 1-b, c shows the lists of content and people being recommended respectively. As a result, a user especially a new user can use the social map to get a quick glimpse of a content-centric social website and learn about the website’s dynamics (e.g., popular content and people). In this sense, Pharos uses a social map to recommend “hot communities” and relevant “hot items” (i.e., content and people) to all users, alleviating the cold start problem. Furthermore, the social map provides a natural context for users to grasp the recommendation results (e.g., hot communities and hot items), since it helps explain the existence of latent communities and their characteristics (e.g., highly influential content and individuals).

In short, Pharos offers two unique contributions. First, it addresses the cold start problem in part by using a social map to summarize a social website, which in turn helps new users

1http://en.wikipedia.org/wiki/Magical_objects_in_Harry_Potter
quickly locate their interests. Second, it uses the social map to help explain recommendation results by means of content, people, and their relationships, helping improve recommendation trustworthiness.

RELATED WORK
Over the years, researchers have used different methods to address the cold start problem. For example, Schein et al. [10] explore several generative probabilistic models to help recommend new items to end users. Park et al. [9] introduce filterbots to make collaborative filtering algorithms more robust when dealing with new items. More recently, Park and Chu [8] take predictive feature-based regression models to tackle three types of cold-start problems: 1) recommendation on existing items for new users; 2) recommendation on new items for existing users; and 3) recommendation on new items for new users. Compared to these works, Pharos provides a social map to new users, which helps them quickly grasp a content-centric web site and locate their interests.

Ensuring recommendation trustworthiness is another popular research topic in building recommender systems. An early piece of work on recommendation explanation is done by Herlocker et al. [7]. They explore the utility of explanations in ACF (automated collaborative filtering) systems, by exposing neighbors and their ratings to end users. Guy et al. [5] study the effect of using social network information to explain recommended items. Vig et al. [13] introduce tag-based explanation, linking items and users via tag relevance and tag preference. Compared to these works, Pharos employs a social map, a summary of users’ content-related social behavior, to help explain recommendation results.

The increasing popularity of social websites leads to various work on social recommender systems. Chen et al. [2] present recommendation algorithms that can effectively expand users’ friend lists in a social network. Guy et al. [4] describe a social recommender that recommends people to join one’s social network. Most existing works rely on explicit social networks to make recommendations. In contrast, Pharos utilizes implicit social relationships dynamically extracted from users’ content-related social behaviors (i.e., latent communities in a social map) to make recommendations.

EXAMPLE APPLICATION
Pharos is designed to help users locate desired information at content-centric social websites, such as forums and blogs. Here, we use the IBM BlogCentral website (our internal blog website) as an example to show the use of Pharos.

Alice, a new employee of IBM Lotus wants to use BlogCentral to learn more about Lotus:

- After login, Alice is presented with a social map of BlogCentral (Figure 1-a). In this case, Pharos chooses to display seven “popular” communities discovered based on user behaviors from August 1st to 15th, 2009. She notices that one community seems talking about “lotus” (highlighted in brown).
- Alice then drills down on the “lotus” community. She gets a list of recommended blog entries (Figure 1-b). Each blog entry is associated with a list of people (e.g., authors and commentors). From the people listed, Alice notices that Bob is a very active blogger. She clicks on Bob to find out more about him. Then she gets a list of recommended bloggers with Bob ranked at the top (Figure 1-c). In addition, the social map highlights Bob’s activities in multiple communities (Figure 2).
- Based on the information displayed on the social map, Alice notices that Bob is also active in another community on the subject of “iphone” (highlighted in yellow). She clicks on that community and then gets a list of recommended blog entries. Again, Alice can now learn more about the content and people involved in the “iphone” community.

In summary, Alice is able to use the social map created by Pharos to understand the key activities of BlogCentral and easily learn the topics and people that she might be interested in.

PHAROS OVERVIEW
Pharos is designed to run on top of existing content-centric social websites. We assume that the target website always
has a database that contains all the site content and server logs that record user interaction behavior. As shown in Figure 3, Pharos has three key components: social map generator, recommender and visual explainer. Based on users’ content-related social behavior (e.g., reading, writing, and commenting), which are often provided by a website’s database and server logs, Pharos dynamically generates a social map. Based on the generated social map, the recommender uses several algorithms to recommend both people and content to end users. The recommendation is explained visually by means of communities in the context of a social map.

**SOCIAL MAP-BASED RECOMMENDATION**

Since the creation of a social map is the key of Pharos, here we focus on describing how Pharos dynamically creates a social map. We also briefly explain how Pharos recommends “hot” content or people in a selected community on a social map.

**Social Map Definition**

Before describing how Pharos generates a social map and uses it to make recommendations, first we give the formal definition of a social map and its components. A social map is a dynamically generated “marauder’s map” that summarizes users’ content-related social behavior of a content-centric social website. A social map consists of a set of identified latent communities. Each community consists of two parts: the content being discussed and the people involved. Just like communities in a real world, a piece of content or a person may belong to multiple communities at the same time. In addition, a social map evolves over time as people’s behaviors and content dynamically change. More precisely, a social map $M$ consists of $n$ latent communities, $M = \{C_1, \ldots, C_n\}$, where $C_i$ is the $i$-th community. Each latent community $C_i$ has two parts: $C_i = \{D_i, P_i\}$, where $D_i$ represents a set of data/content items; and $P_i$ represents a set of relevant people. Currently, content items in $D_i$ may be considered as user-generated documents (e.g., a blog entry or a forum post).

**Social Map Creation**

Pharos take three steps to create a social map:

1. community extraction,
2. community ranking, and
3. community labeling.

To extract meaningful latent communities that can form a social map, potentially we can use three approaches: (1) we directly model user-content relationships by using co-clustering methods; (2) starting from people, we group them by their behavioral patterns or demographic characteristics (e.g., geographical information). We then follow people-content relationships to assign associated content to corresponding people groups; (3) starting from the content, we cluster content, and then assign associated people to corresponding content clusters. From our research experience, we decide to adopt the third approach for several reasons. First, due to the extremely sparse (less than 0.1% in our test bed) people-content relationship data, co-clustering methods (approach 1) normally would not work well. Second, a large portion of user behavioral data is contributed by anonymous users, leading to the difficulty of distinguishing different users and modeling user behavioral patterns (approach 2). Third, in a content-centric website, a user is more likely to be interested in content, rather than people. In other words, the third approach of treating the content as the first class citizen and summarizing user social behavior around content makes more sense to end users.

There are many approaches to content clustering. Because of its effectiveness in summarizing text content [3], we decide to leverage LDA (Latent Dirichlet Allocation) [1] to summarize all content into a set of topics. Naturally, topics form content clusters. Based on the probabilistic distributions of each content item over topics, we assign the membership of the content items to the most relevant topic (the probability score being the highest). At last, we assign associated people to the content clusters based on their content-related social behaviors.

Since the derived communities are not ranked, the second step is to rank derived communities based on their importance. The goal of this step is to identify “hot” communities to be presented on a social map. There are a number of metrics to measure the “hotness” of a community. Currently, we measure the “hotness” based on both content and people authorities. The key idea is that content items with more hits and replies are considered more popular, and people who
We deployed Pharos internally at IBM since August 2009 to help IBM BlogCentral users navigate blogs and locate their interests. The back-end system is implemented in Java, and the front-end user interface is implemented in HTML/JavaScript and Flex. To evaluate the usefulness of Pharos, we have conducted a preliminary evaluation.

Our evaluation consisted of two parts. First, we asked eight users to use Pharos at BlogCentral for a short period of time (several weeks). Second, we conducted a survey, asking eleven (11) users including the eight (8) Pharos pilot users about their opinions of BlogCentral and the usefulness of Pharos (if they were Pharos users).

The Survey Results
Our questionnaire contains eight questions, covering three categories: user profiling, viewpoints on current BlogCentral system and recommendations, and assessment of Pharos functions if they have used Pharos before.

From our survey results, 7 out of 11 people selected “read” as their major task in BlogCentral. When asked what they would like to see on the home page of the Blog Central website, 9 out of 11 mentioned that they would like to see an overview of “hot spots”; and 6 out of 11 thought an overview would be very useful. When commenting on the recommendations used by the current BlogCentral website, people’s opinions were inconsistent. Half of them thought they were useful while the other half did not think so. These answers indicated that our motivation in Pharos is valid: it is useful to provide an overview of a social website.

From the eight users who used Pharos for a short period of time, their answers were positive. In particular, they found the social map useful for them to grasp and track the activities at BlogCentral; and the social map also helped explain the recommendation results.

More User Feedback
Our pilot Pharos users also provided us more feedback. Several users commented that there was too much information displayed on a social map, which could be a burden for end users. They suggested that each community only display two or three labels to summarize people and content. On the other hand, a few users complained the labels displayed on a social map were too terse, and sometimes they had to guess what they meant.

CONCLUSIONS
In this paper, we present Pharos, a social map based recommendation, to address two challenges in recommender systems. First, it provides a social overview of a content-centric website along with “hot” content and people, helping new users quickly locate their interests and in part addressing the cold start problem. Second, it uses a social map to explain recommendations by means of relating content with user social activities, increasing recommendation’s trustworthiness. Pharos was deployed internally at IBM, helping bloggers navigate the IBM BlogCentral website. Our preliminary evaluation has showed that the social map helps end users grasp the overview of the website and comprehend recommendations results. Feedback from our pilot users also
points future directions to improve Pharos.

REFERENCES


