Chapter 1

Interest-Based Data Dissemination in Opportunistic Mobile Networks: Design, Implementation and Evaluation

Wei Gao

Department of Electrical Engineering and Computer Science, The University of Tennessee, Knoxville, TN

Wenjie Hu

Department of Computer Science and Engineering, The Pennsylvania State University, University Park, PA

Guohong Cao

Department of Computer Science and Engineering, The Pennsylvania State University, University Park, PA

CONTENTS

This research focuses on providing pervasive data access to mobile users without support of cellular or Internet infrastructure. Two mobile users are able to communicate and share data with each other whenever they opportunistically move into the Bluetooth communication range of their smartphones. We designed and implemented our system on Android-based smartphones, and deployed our system to students at the Penn State University campus. Our system dynamically captures the interests of mobile users at runtime, and intelligently distribute to users with the data that they are interested in. It also provides a unique research facility for investigating the interests and the data access patterns of users in various mobile environments.

1.1 Introduction

With the recent technical advance and popularization of smartphones which are able to store, display and transmit various types of media content, it is important to provide pervasive data access to mobile users, and distribute media contents to these users promptly and efficiently. A straightforward solution is to provide such data access via the 3G cellular network infrastructure. However, the data access will create huge amount of data traffic for the 3G network, which imposes immense pressure on the limited spectrum and the backend resources of 3G networks, and hence deteriorating the quality of service. This also motivates research on offloading 3G traffic to other communication networks such as WiFi [23, 1], but the coverage of WiFi accessibility is still limited nowadays as reported in [1], especially for mobile users which keep moving.

To address this challenge, in this paper we develop a practical system which enable mobile users to autonomously transmit and share data with each other by exploiting the unused *opportunistic communication links* between the short-range radios (i.e., Bluetooth) of their smartphones. Two users are able to communicate when they opportunistically *contact* each other, i.e., moving into the Bluetooth communication range of each other. The networks consisting of these users are called *Opportunistic Mobile Networks*, which are also known as Delay Tolerant Networks (DTNs) [7] or Pocket Switched Networks (PSNs) [14].

The contributions of our work are two-fold. First, our system provides prompt and efficient access on categorized web data to mobile users, based on their interests in the data. More specifically, our system dynamically captures the interests of mobile users at runtime, and intelligently send them their interested data. The power constraints of smartphones are also taken into account, and we propose solutions to tradeoff between power consumption and data availability of mobile users. To the best of our knowledge, this is the first work to implement opportunistic data sharing/access among smartphones. Previous work has focused on analyzing the contact or communication patterns of mobile users based on the collected user behavior records [19, 18, 1], and furthermore propose data sharing schemes based on the analysis results [19, 3, 24]. However, these schemes have never been implemented or evaluated with practical user involvement and realistic media content.

Second, our system is going to be deployed over a large number of smartphone users in mobile environments, and hence provides a unique research facility for investigating the user interests and the data access patterns of mobile users. Comparatively, although a lot of experimental systems have been developed to monitor and record the behaviors of mobile users in various mobile environments, ranging from university campus (Dartmouth [17], MIT Reality [6], UCSD [20]), conference sites (INFOCOM [5]) to urban areas (DieselNet [2]), they are limited to record and analyze the contact process among mobile users. The characteristics of users' interest and data access patterns are generally neglected.

The rest of this paper is organized as follows. We motivate the proposed work with potential mobile environments and applications that benefit from our system in Section 1.2. The design and the implementation of our system are described in Sections 1.3 and 1.4 respectively. The experiment results are presented in Section 1.5. Section 1.6 summarizes the related work. Section 1.7 discusses future research directions and concludes the paper.

1.2 Motivation

The practical applicability of our system can generally be motivated by the following mobile environments.

Large-scale public commute systems, in which opportunistic communication links can be found among commuters on the bus, train, ferry or subway. First, due to the mobility and physical constraint of public transportation, 3G or WiFi network coverage is usually unavailable to commuters. For example, smartphones do not have

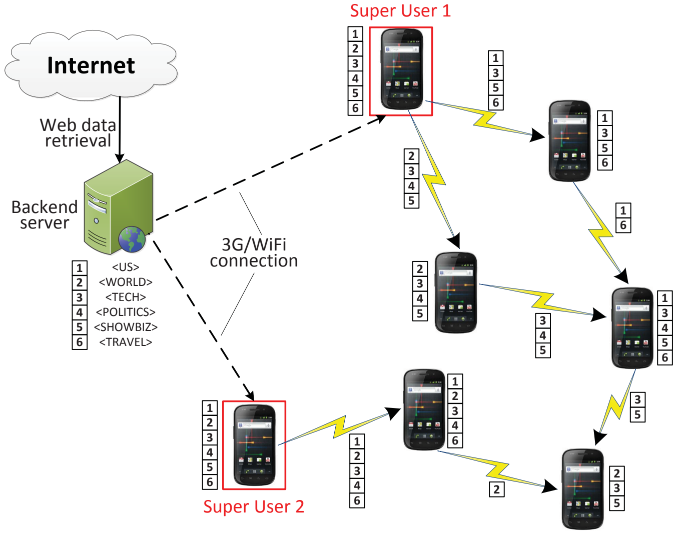


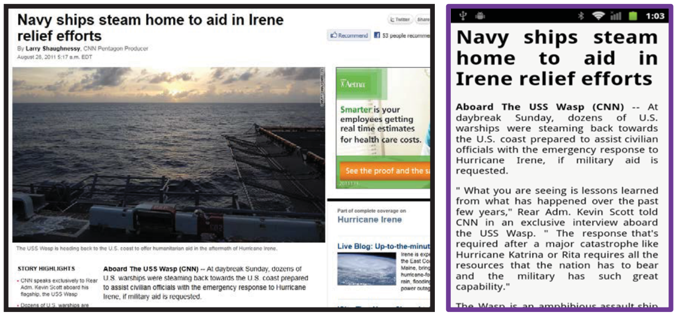
Figure 1.1: Overview

3G signal in the underground subway, and commuters on the train may frequently lose WiFi connections when the train is moving. Second, the population density of mobile users is usually high in such environments. This high density, on one hand, ensures the existence of opportunistic communication links among commuters. On the other hand, it ensures that a sufficient amount of data is available among commuters, and many users may share common interests. For example, most of the commuters will be interested in daily news, weather report or traffic information.

Public event, in which opportunistic communication links can be found among mobile users at stadium, museum, theater or shopping mall. Although 3G or WiFi network coverage may be available at these places, the high population density and traffic demand during the event - sports game, concert or holiday sale - may incur fierce competition for the limited channel bandwidth and significantly reduce data transmission rate. Particularly, most of the users will be interested in the same data during the event. For example, they may be interested in the live score of other concurrent matches during a NBA game, and they may be interested in the coupon or discount information during a holiday sale at a shopping mall. In these cases, the efficiency of data access can be significantly improved when data is shared among themselves.

1.3 System Design

Our system provides mobile users with the access to categorized media news, which is dynamically retrieved by our backend server from the well-known news websites,



Navy ships steam home to aid in Irene relief efforts

By Larry Shaughnessy, CNN Pentagon Producer
August 28, 2011 5:17 a.m. EDT

The USS Wasp is heading back to the U.S. coast to offer humanitarian aid in the aftermath of Hurricane Irene.

STORY HIGHLIGHTS

- CNN speaks exclusively to Rear Adm. Kevin Scott aboard his flagship, the USS Wasp
- Dozens of U.S. warships are

Aboard The USS Wasp (CNN) -- At daybreak Sunday, dozens of U.S. warships were steaming back towards the U.S. coast prepared to assist civilian officials with the emergency response to Hurricane Irene, if military aid is requested.

Part of complete coverage on
Hurricane Irene

Live Blog: Up-to-the-minut

Irene is exp the East Co Maine, bring hurricane-fo rain, flooding power outag

Twitter
Share
Recommend
53 people recomme
Aetna
Smarter is your employees getting real time estimates for health care costs.
See the proof and the s
1:03

Navy ships steam home to aid in Irene relief efforts

Aboard The USS Wasp (CNN) -- At daybreak Sunday, dozens of U.S. warships were steaming back towards the U.S. coast prepared to assist civilian officials with the emergency response to Hurricane Irene, if military aid is requested.

" What you are seeing is lessons learned from what has happened over the past few years," Rear Adm. Kevin Scott told CNN in an exclusive interview aboard the USS Wasp. " The response that's required after a major catastrophe like Hurricane Katrina or Rita requires all the resources that the nation has to bear and the military has such great capability."

The Wasp is an amphibious assault ship

Figure 1.2: Comparison between the original webpage an the re-formatted news page from CNN

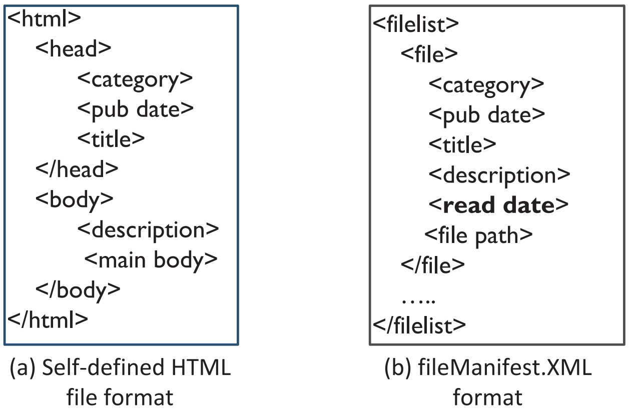
including CNN, New York Times and BBC, on a hourly basis. The raw data retrieved from these websites is classified into a finite number of categories, and is re-formatted for easy processing at smartphones. It is important to notice that, the functionality of our system is not limited to media news, and it can be applied to any categorized media content ranging from photos at Flickr, video clips at Youtube to music at iTunes.

Our system is developed on smartphones which are distributed to voluntary students at the Pennsylvania State University. The process of data access among mobile users is illustrated in Figure 1.1. The categorized news is accessible to mobile users via a subset of smartphones which are employed as the “super users”. A super user retrieves the up-to-date news from the backend server on a hourly basis, when it has 3G or WiFi connections available. Such news is then shared and transmitted among mobile users when they opportunistically contact each other, according to their interests in the news. In this way, we ensure that a user only receives the news that he is interested in. As illustrated in Figure 1.1, the amount of data available to a user will not decrease during the multi-hop data sharing process, because a user generally receives data from multiple users.

In general, the process of data sharing and transmission is transparent to the mobile users. A user only reads the news that he is interested in when his phone reports that new data is available.

1.3.1 Collecting Data from the Web

A web crawler is running at the backend server to collect news from the aforementioned websites periodically, in the form of HTML webpages. In general, these webpages are usually redundant and contain a large amount of stuff such as the CSS/frame specifications, outlinks, advertisements, Javascripts, etc, which are not related with the news content. As a result, these webpages are re-formatted by our system, so that the data size can be reduced significantly and the pieces of news are more convenient for the smartphones to process and display. The comparison be-



```
<html>
  <head>
      <category>
      <pub date>
      <title>
  </head>
  <body>
      <description>
       <main body>
  </body>
</html>
```

(a) Self-defined HTML file format

```
<filelist>
  <file>
    <category>
    <pub date>
    <title>
    <description>
    <read date>
   <file path>
  </file>
  …..
</filelist>
```

(b) fileManifest.XML format

Figure 1.3: Format of HTML and XML files

tween the original webpage and the re-formatted news page from CNN is illustrated in Figure 1.2.

The web crawler is responsible for re-formatting the webpages, which can be computationally expensive and hard to realize on smartphones. The important attributes of the news, such as the publishing date, title and category, are extracted from the webpage. These attributes, along with the main text of the news, is saved as a self-defined simple HTML file as shown in Figure 1.3(a) and transmitted to smartphones. Correspondingly, a data handler is implemented at each smartphone to manage the received HTML files. This management is realized by using a special XML file named `fileManifest.XML` as shown in Figure 1.3(b). Each HTML file (i.e., a piece of news) is mapped to a `<file>` element in `fileManifest.XML`, in which the attributes of the piece of news are also recorded.

To show the efficiency of such webpage re-formatting on reducing the file sizes, we analyzed the news webpages that we collected from CNN during one day. The average size of original webpages is around 47KB. As shown in Figure 1.4, the average size of HTML files after re-formatting is only 7.4% of the original webpages.

1.3.2 User Interest Profile

Each smartphone maintains a *User Interest Profile (UIP)*, which is manually initialized when the phone is first distributed to the user. Later on, the UIP of a user is dynamically updated by his phone whenever he reads the news that he is interested in. When two smartphones contact each other, they determine which news should be transmitted to each other based on their maintained UIPs.

The UIP of a mobile user is maintained with respective to the set of categories (denoted as $\mathcal{C}$) that the media news is classified. As shown in Figure 1.5, each category $j \in \mathcal{C}$ is associated with an interest index I_j which ranges from 1 to 100. The

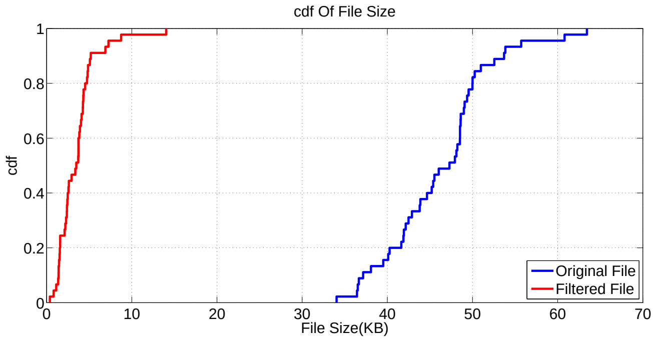
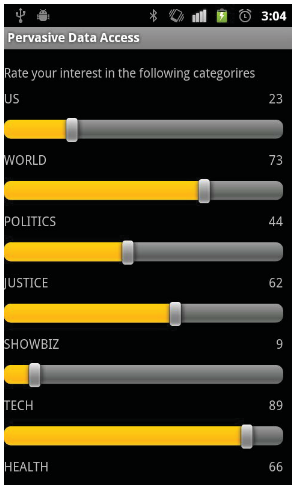


Figure 1.4: CDF of file size after filtering



Pervasive Data Access

Rate your interest in the following categorires

US 23

WORLD 73

POLITICS 44

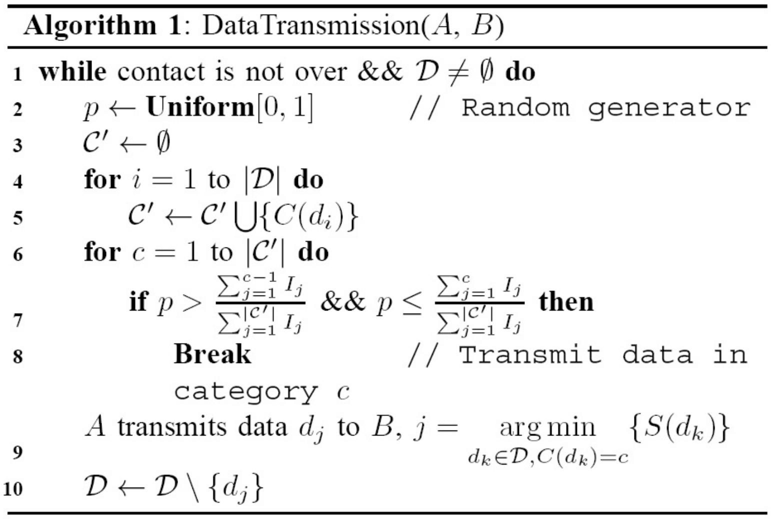
JUSTICE 62

SHOWBIZ 9

TECH 89

HEALTH 66

Figure 1.5: User Interest Profile (UIP)



```
Algorithm 1: DataTransmission($A$, $B$)
while contact is not over && $\mathcal{D} \neq \emptyset$ do
    $p \leftarrow$ Uniform$[0, 1]$        // Random generator
    $\mathcal{C}' \leftarrow \emptyset$
    for $i = 1$ to $|\mathcal{D}|$ do
        $\mathcal{C}' \leftarrow \mathcal{C}' \bigcup \{C(d_i)\}$
    for $c = 1$ to $|\mathcal{C}'|$ do
        if $p > \frac{\sum_{j=1}^{c-1} I_j}{\sum_{j=1}^{|\mathcal{C}'|} I_j}$ && $p \leq \frac{\sum_{j=1}^{c} I_j}{\sum_{j=1}^{|\mathcal{C}'|} I_j}$ then
            Break        // Transmit data in category $c$
    $A$ transmits data $d_j$ to $B$, $j = \underset{d_k \in \mathcal{D}, C(d_k)=c}{\arg\min} \{S(d_k)\}$
    $\mathcal{D} \leftarrow \mathcal{D} \setminus \{d_j\}$
```

Figure 1.6: Algorithm for data transmission

values of interest indices specified by users are internally normalized by our system to ensure that $\sum_{j=1}^{|\mathcal{C}|} I_j = 100$ at any time.

In order to update the UIP, a smartphone maintains N_j as the number of times that the user reads the news in the category j and N_j is initialized as 1. Whenever a user reads a piece of news belonging to the category $k \in \mathcal{C}$, the UIP of the user is updated as follows. First, the corresponding interest index I_k is updated to $I_k \cdot \frac{N_k+1}{N_k}$. Afterwards, the interest indices over all the categories are normalized to ensure that $\sum_{j=1}^{|\mathcal{C}|} I_j = 100$. More specifically, for each category $j \in \mathcal{C}$, we update I_j to $I_j \cdot \frac{100}{100+I_k \cdot \frac{1}{N_k}}$. Note that the value of I_k before update is used for this normalization.

It is easy to see that N_k monotonically increases as time elapses. As a result, the UIP of a user gradually becomes stable over time.

1.3.3 Data Transmission

1.3.3.1 Basic approach

Due to mobility, two users may only contact for a limited period of time, and it is usually hard to predict such contact duration. Therefore, when a smartphone A contacts another phone B, A needs to determine an appropriate sequence for transmitting the pieces of news that B does not have to B, so as to ensure that the news B is interested in can be transmitted before the contact ends.

A straightforward solution is to determine such sequence based on B's UIP over different categories of news. However, B may only receive news of few categories. Instead, we propose a probabilistic solution to determining such sequence over all the categories. Suppose that A has the set $\mathcal{D}$ of news that B does not have, the process of such data transmission is described by Algorithm 1.6, where $C(d_k)$ and $S(d_k)$ indicate the category and size of the piece of news d_k respectively, and I_j indicates the interest index of B at category j. In general, A determines which piece of news in $\mathcal{D}$ to be transmitted to B one by one before the contact with B ends, and the news in category c has the probability $I_c/\sum_{j=1}^{|\mathcal{C}'|} I_j$ to be transmitted next. After c is probabilistically determined, A transmits the piece of news in category c with the smallest size to B.

1.3.3.2 Considering data freshness

The freshness of different pieces of news is also considered in determining the data transmission sequence. For a piece of news d_i which is in the category $c_i \in \mathcal{C}$ and was generated at T_i, the freshness of d_i is calculated as:

$$F_i = \begin{cases} 1, & T_{now} - T_i < T_G \\ \frac{1}{\frac{T_{now}-T_i}{T_G}} = \frac{T_G}{T_{now}-T_i}, & other \end{cases} \tag{1.1}$$

where T_{now} is the current time and T_G is the time granularity for calculating data freshness. For example, if we set T_G to one day, all pieces of news generated in the current day will have $F_i = 1$, while the news generated yesterday will have $F_i \in [1/2, 1)$.

In general, we prioritize to transmit the most important pieces of news first within the limited contact duration between mobile users. The importance of a piece of news takes both user interest and data freshness into account, and the importance P_i for a piece of news d_i is defined as

$$P_i = mI_i/100 + (1-m)F_i, \tag{1.2}$$

where m is used to adjust the weight between user interest and data freshness. As a result, each piece of news can be mapped into the square area shown in Figure 1.7. In data transmission, we introduce interest threshold S_I, freshness threshold S_F and Priority Boundary Line (PBL) to realize the balance between user interest and data freshness. S_I and S_F are the lower bounds of user interest and data freshness. If a piece of news is too old, or attracts little interest from a user, it will fall into area A and not to be transmitted. Otherwise, the news corresponding to area B are more fresh and attract interests of most users. These data are prioritized for being transmitted in the first place.

Determining transmission priority for pieces of news falling into the areas C and D is more complicated. We divide C and D into smaller areas based on PBL. We believe that users are still interested in some data in the areas $C1$ and $D1$. We setup the PBL to be across (S_I, S_F) and with a gradient determined by m. We will only send data whose priority is above the PBL.

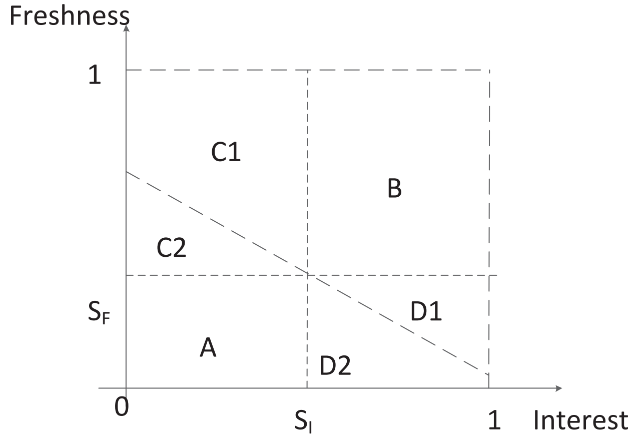


Figure 1.7: Data transmission priority considering user interests and data freshness simultaneously

1.3.4 Multi-Party Data Transmission

Furthermore, we explore the possibility of multi-party communication using Bluetooth radios of smartphones, i.e., a user transmits data with multiple peers simultaneously. Such multi-party communication is particularly challenging in Android-based smartphones due to the hardware limitation, which can be elaborated in the following three perspectives. First, current smartphones do not support reuse of the Bluetooth radio. In a TCP/IP connection, a user can use multiple threads in the OS to support multi-party communication with multiple counterparts simultaneously. However, such reusability is not applicable to Bluetooth communication. Due to the hardware limitation of Bluetooth radios in smartphones, it can only support one connection at one time, and will not respond to other connection requests given an existing Bluetooth connection. Second, Bluetooth radio can not check the channel availability before transmission, like what CSMA/CA does in WiFi. Third, broadcast is basically not supported by Bluetooth.

In this paper, we propose three new schemes to support multi-party data transmission using Bluetooth radio of smartphones.

- *Sensing before sending:* When a smartphone wants to communicate with others via its Bluetooth radio, it will first check the status of Bluetooth radio. If the Bluetooth radio is available, it will wait for a specific time period and start the connection. If not, it will wait for the end of previous communication.

- *E2E back after failing:* If a smartphone initializes a Bluetooth connection but fails, it believes that the other phone is busy for another communication and then backoff for a random time period $t_{back} \in [0, \mathcal{W}]$ before trying again. If

```
Algorithm 2: LoadBalancing($\mathcal{S}$, $\mathcal{S}'$, $\mathcal{U}$, $K$)
$n \leftarrow 0, \mathcal{U}' \leftarrow \emptyset$
while $n < K$ do
    $j \leftarrow \arg\max_{k \in \mathcal{S}' \setminus \mathcal{U}} \{E(k)\}$
    $j$ is selected as the super user
    $\mathcal{U} \leftarrow \mathcal{U} \bigcup \{j\}, \mathcal{U}' \leftarrow \mathcal{U}' \bigcup \{j\}, \mathcal{S}' \leftarrow \mathcal{S}' \setminus \{j\}$
    $n \leftarrow n + 1$
    if $\mathcal{U} == \mathcal{S}$ then
        $\mathcal{U} \leftarrow \mathcal{U}'$        // Every user has been
        selected, so reset $\mathcal{U}$
```

Figure 1.8: Algorithm for load balancing

it fails the *n*-th time, it will select a back time $t_{back} \in [0, 2^n\mathcal{W}]$. After 5th try, the back window will remain unchanged and not be enlarged again.

- *Switching connection:* The Bluetooth connection between two smartphones will be terminated if there is no more file to transmit or after transmitting S_N number of files. After termination, the two smartphones will not connect to each other again during the next time period S_A, to give other smartphones the opportunity to communicate.

1.3.5 Power Constraint of Smartphones

Smartphones usually consume more power than traditional cellphones due to their advanced functionalities on running complicated applications and processing various media contents. Hence, the battery power on smartphones is usually limited and should be taken into account during the system design. In our previous work [13, 11], we have made initial efforts on reducing the power consumption of smartphones by optimizing their contact detections. We aim to further address the power constraints of smartphones via workload balancing and adjustment of data access strategies.

1.3.5.1 Balancing the Workload of Super Users

As illustrated in Figure 1.1, the super users are responsible for retrieving the up-to-date news from the backend server on a hourly basis, and hence consume their battery power much faster. To balance the workload of super users, in our solution, all mobile users (denoted as set $\mathcal{S}$) in the network are scheduled to be super users in a round-robin manner, and the super users are changed daily.

Due to the difficulty of distributed coordination among mobile users, super users are selected at the backend server according to the status of the smartphones' battery power. Each smartphone reports its remaining battery power to the server at 3AM

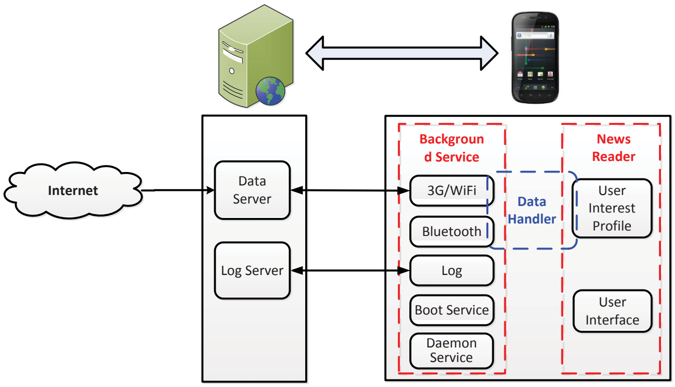


Figure 1.9: System implementation

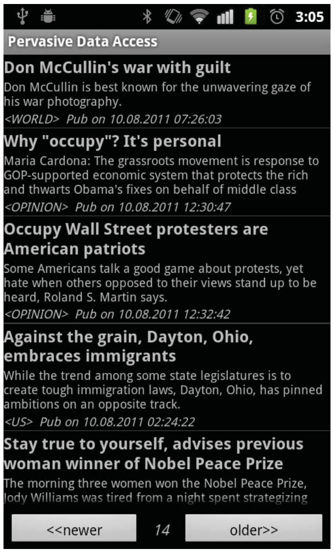
everyday via 3G or WiFi connections. The server selects K super users from the set $\mathcal{S}'$ including all the users having reported to the server. The selection process is described by Algorithm 1.8, where $E(k)$ indicates the remaining battery power of user k, and $\mathcal{U}$ indicates the set of users that have been selected as the super user recently and will not be the super user again in the near future. In general, the users with the maximum remaining battery power of their smartphones are selected as the super users.

The server is responsible for notifying each selected super user about the selection result. Since the execution time of Algorithm 1.8 is negligibly short, any selected super user can be notified. At the same time, the super users of the previous day automatically revoke themselves at 3AM.

1.3.5.2 Tradeoff between Power Consumption and Data Availability

Our system also enables the mobile user to flexibly tradeoff between the power consumption of data transmission of their smartphones and the availability of media news. Such tradeoff is realized by a user-controlled parameter λ valued in the range $[0,1]$. $\lambda = 0$ means that the user wants to conserve the maximum amount of battery power and does not want to receive any data, while $\lambda = 1$ means that the user wants all the available data regardingless of the transmission overhead.

Such tradeoff can be easily integrated with the process of determining data transmission sequence described in Algorithm 1.6. More specifically, every time when a piece of news d_j is picked, A only has probability λ_B to send this news to B, where λ_B is the tradeoff parameter specified by B. In practice, a user can adjust λ at any time according to the power status of the smartphone.



Pervasive Data Access

Don McCullin's war with guilt
Don McCullin is best known for the unwavering gaze of his war photography.
<WORLD> Pub on 10.08.2011 07:26:03

Why "occupy"? It's personal
Maria Cardona: The grassroots movement is response to GOP-supported economic system that protects the rich and thwarts Obama's fixes on behalf of middle class
<OPINION> Pub on 10.08.2011 12:30:47

Occupy Wall Street protesters are American patriots
Some Americans talk a good game about protests, yet hate when others opposed to their views stand up to be heard, Roland S. Martin says.
<OPINION> Pub on 10.08.2011 12:32:42

Against the grain, Dayton, Ohio, embraces immigrants
While the trend among some state legislatures is to create tough immigration laws, Dayton, Ohio, has pinned ambitions on an opposite track.
<US> Pub on 10.08.2011 02:24:22

Stay true to yourself, advises previous woman winner of Nobel Peace Prize
The morning three women won the Nobel Peace Prize, Jody Williams was tired from a night spent strategizing

Figure 1.10: News Reader

1.4 System Implementation

The implementation of our system is illustrated by Figure 1.9. At the server side, the web crawler retrieves data from the Internet as described in Section 1.3 and transmits the re-formatted news to the super users. The log module is responsible for recording the contact and data access behaviors of mobile users. At the smartphone side, the handling of system execution and data transmission is implemented as background system services. These services then interact with the news reader. The news reader displays the received news to the user as shown in Figure 1.10 and maintains the UIP of user. Such interaction is realized via data handler which manages the received news.

1.4.1 Development Platform

The system was developed on the Samsung Nexus S phone, which is equipped with a 1GHz processor to support efficient processing of media contents, and a 16GB SD card is pre-installed on the phone to provide sufficient storage space. Both AT&T 3G and 802.11 b/g/n networks are supported on this phone.

Our system is implemented based on the Android 2.3.3 (Gingerbread) OS. It would be desirable if our system could be developed as a stand-alone application which is independent from the OS kernel, and could be downloaded and running on any smartphone. However, it is challenging to realize this in practice due to the following system requirements:

- Our system is required to be continuously running at background for a long period of time without user involvement, so as to enable the smartphones to share the media news with each other. However, this capability is not supported by some mobile OS for third-party applications. For example, iOS on iPhone only enables the system applications to be running at background, and other applications can only be "waken" up by push notifications. Instead, we implemented this capability on the Android OS by developing specific system applications.

- A smartphone in our system should be able to periodically detect the existence of other smartphones within the communication range of its Bluetooth radio. This requires the phone's Bluetooth interface to be persistently set at the *discoverable* mode. However, the current Android OS limits the Bluetooth interface to be at discoverable mode for up to 120secs. Active user involvement is required periodically to resume the discoverable mode. In our implementation, the Android OS kernel is modified to address this limitation.

1.4.2 System Implementation

In this section, we describe how the aforementioned challenges are addressed to ensure that our system can be running correctly on the Android OS.

1.4.2.1 Discoverable Mode of Bluetooth

We require the Bluetooth interface of a smartphone to be persistently set at the discoverable mode so that the phone can periodically detect other nearby peers. However, in Android OS, whenever the Bluetooth interface is set at the discoverable mode, a system timer with 120secs is triggered, and the discoverable mode is set off when the timer expires. Our solution to solve this problem is shown in Figure 1.11. We remove this limit and keep monitoring the system timer. The timer is reset whenever it expires, so that the discoverable mode is resumed as soon as it has been set off.

This modification, by itself, cannot completely ensure that Bluetooth is always in discoverable mode, as users or other application processes can still turn Bluetooth off. We develop a specific daemon service to solve this problem, which is described in Section 1.4.2.3.

1.4.2.2 Removing the System Dialog requesting Bluetooth Permission

In the default workflow of Android OS, each time when a user application requests to open the Bluetooth interface, the Android OS will pop up a dialog requesting user permission to do that, and hence user involvement is required for setting up the Bluetooth discoverable mode. To address this problem, we modified the kernel of Android OS, so that each time when a request is sent to the OS to setup the Bluetooth discoverable mode, a parameter T is specified to indicate the elapsed time we request for the discoverable mode. In particular, we set T to be a negative value, so that we

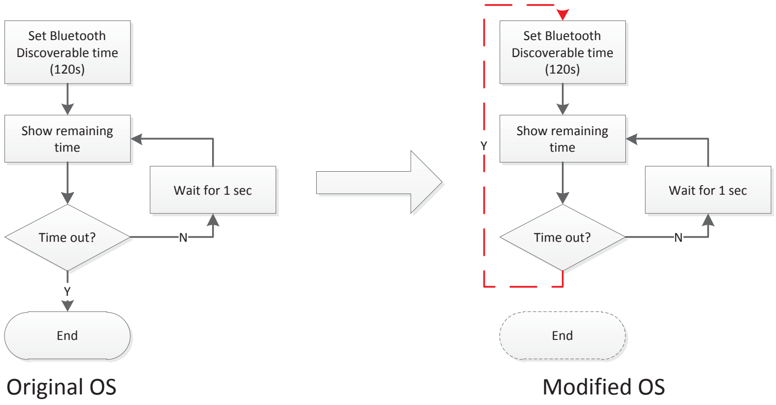


Figure 1.11: Extending Bluetooth discoverable time

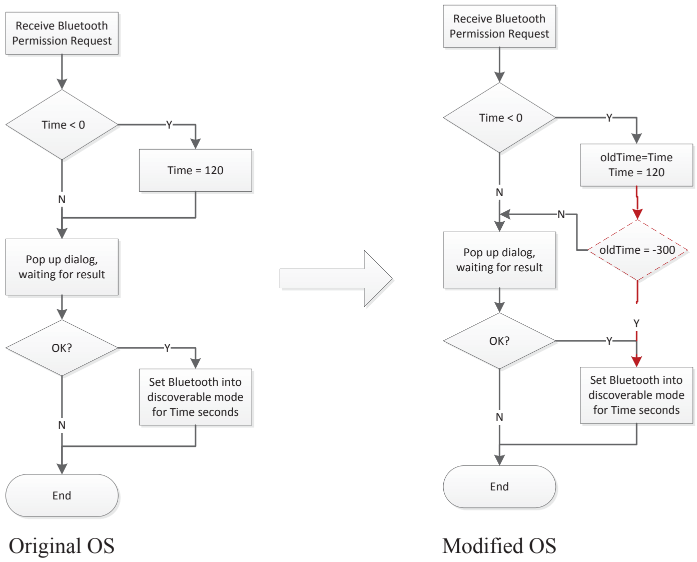
are able to modify the system workflow and skip over the system dialog requesting user permission. The elapsed time for Bluetooth discoverable mode is then set as 120 seconds automatically, as shown in Figure 1.12.

1.4.2.3 Boot and Daemon Services

We developed boot service as a special system application running in the background, which is started with the Android OS and automatically starts our system afterwards. A daemon service is developed to make sure that our system is resilient to unexpected failures of smartphones, such as power depletion, system crash or user mistakes. The boot service performs the following checking tasks every 30 seconds:

- *Checking Bluetooth discoverable mode:* If Bluetooth is not in the discoverable mode, it sends a request to enable it for -300 seconds. Thus, the system will set Bluetooth into discoverable mode without interfering the user.
- *Checking Bluetooth server status:* In order to start Bluetooth communication as soon as two devices contact, the Bluetooth server socket of the client is active all the time. If not, we will restart the Bluetooth server socket.
- *Checking neighbor:* The daemon service will also scan for neighbors. If there is one neighbor M available and the Bluetooth device is not connected to any others, it connects to M automatically.

The system should work immediately after the phone is booted without user action. Thus, the boot service also listens to the specific signal that the system generates when being booted. Then, it triggers the daemon service to start, and our system starts to work properly.


Receive Bluetooth Permission Request
Time < 0
Y
Time = 120
N
Pop up dialog, waiting for result
OK?
Y
Set Bluetooth into discoverable mode for Time seconds
N
End
Receive Bluetooth Permission Request
Time < 0
Y
oldTime=Time
Time = 120
N
N
oldTime = -300
Pop up dialog, waiting for result
Y
OK?
Y
Set Bluetooth into discoverable mode for Time seconds
N
End

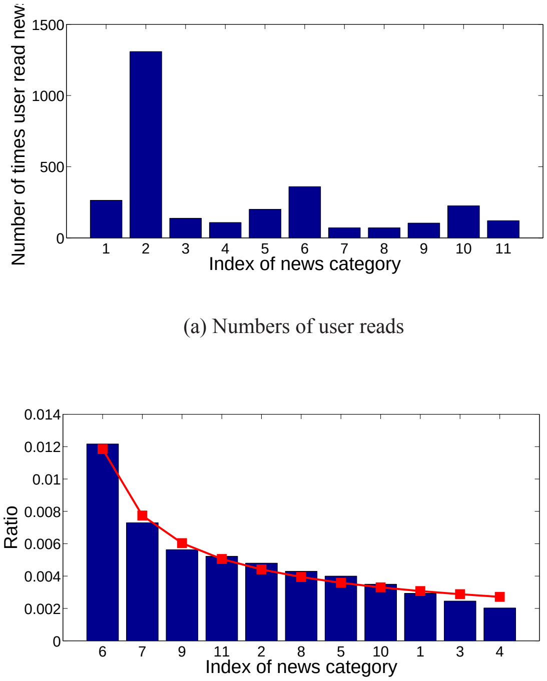
Original OS

Modified OS

Figure 1.12: Remove Bluetooth permission request dialog

Table 1.1: Amount of news in different categories

Index	news categories	amount of news
1	US	5500
2	WORLD	13755
3	POLITICS	3969
4	JUSTICE	3915
5	SHOWBIZ	3992
6	TECHNOLOGY	2754
7	HEALTH	1199
8	LIVING	1816
9	TRAVEL	1547
10	OPINION	3946
11	OTHER	3396



(a) Numbers of user reads

(b) User interest pattern

Figure 1.13: Behaviors of user data access over different news categories

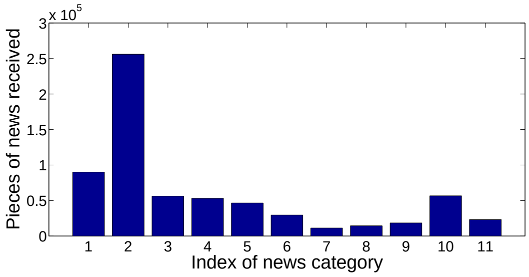
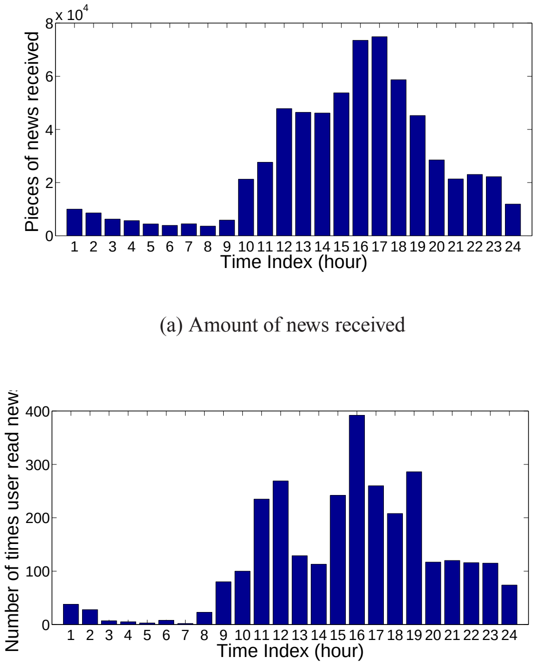


Figure 1.14: Amounts of news received by users



(a) Amount of news received

(b) Numbers of user reads

Figure 1.15: Temporal patterns of data access behaviors of mobile users

1.5 Experimental Results

We have deployed the system with 20 graduate students in the Departments of Computer Science and Engineering (CSE) and the Information Science and Technology (IST) at the Pennsylvania State University, to evaluate their user interests and data access patterns during the time period between September 2011 and March 2012. A total number of 45,807 pieces of media news in 11 categories has been retrieved from the CNN news website and shared among mobile users. The amounts of news in different categories are listed in Table 1.1. We observe that over 40% of the news are found in the categories of "US" and "WORLD", and other news basically are evenly distributed among the remaining nine categories. During the experiment, these news have been transmitted and shared 655,794 times among the 20 users, and have been read 2,970 times by the users.

1.5.1 Data Access Patterns of Mobile Users

As shown in Figure 1.13(a), the news read by users are unevenly distributed over different categories, and this distribution is determined by the amount of news avail-

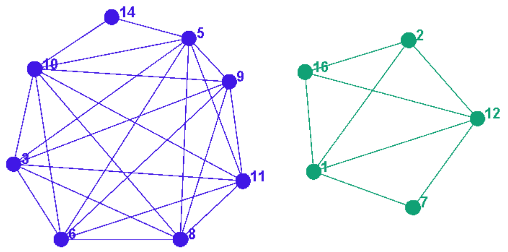


Figure 1.16: Social community structure among mobile users, 81730secs is used as the threshold

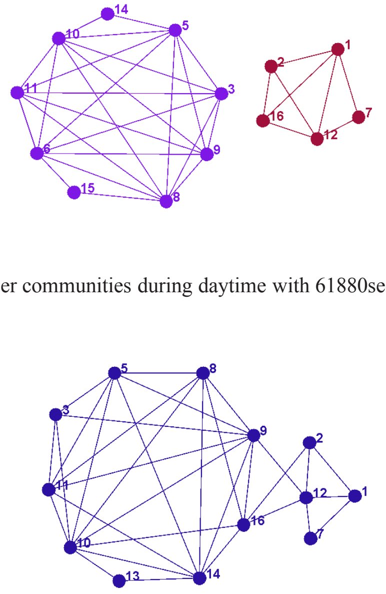
able in different categories. Furthermore, we evaluate the user interest and the data access patterns of mobile users using the ratio of the number of times users read the news over the amount of news available in the network. This ratio over different news categories are shown in Figure 1.13(b). Surprisingly, we observe that variation of this ratio over different news categories can be accurately approximated by power-law curve fitting, which is consistent with the well-known result that the user access pattern over web contents exhibits Zipf characteristics [4].

To better interpret the data access patterns of mobile users, we also evaluate the amount of news that users receive during the experiment periods separately, and the evaluation results are shown in Figure 1.14. We observe that the distribution of the amount of news that users receive is also skewed, and consistent with the data access patterns shown in Figure 1.13(a). In comparison, the result in Figure 1.13(b) provides more accurate characterization of users' data access patterns.

We are also interested in the data access behaviors of mobile users during different time periods in a day. Figure 1.15(a) shows that more than 90% of the news are received by smartphones during the time period between 9AM and 8PM, which indicates that most opportunistic contacts happen among mobile users when they are on campus during daytime. Moreover, Figure 1.15(b) shows that more than 45% of the news are read by users during two specific time periods, i.e., 10AM-12PM and 3-7PM. This result highlights the possible heterogeneity of user interests over time, and motivates us to further explore the temporal differentiation of user interest profiles for more accurate identification of the data that users are interested in.

1.5.2 Characterization of Social Communities

Students involved in our experiment also exhibit noticeable social correlations with each other, when they stay in the CSE department building. As a result, we also explore the social community structure among these mobile users during the experiment period. Social community structure in mobile environments is characterized based on contact patterns of mobile users, such that a community consists of mobile users that frequently contact each other. Palla et al. [22] defines a *k*-clique community as a union of all *k*-cliques (complete subgraphs of size *k*) that can reach each other



(a) User communities during daytime with 61880secs as threshold

(b) User communities during nighttime with 5029secs as threshold

Figure 1.17: User communities during different time periods

through a series of adjacent *k*-cliques. We employ distributed method for detecting *k*-clique communities using the cumulative node contact duration as the threshold for community detection [15]. The result of community detection with $k = 3$ is shown in Figure 1.16. Surprisingly, we observe that students are naturally classified into two groups according to their affiliations with the CSE or IST departments. This community structure therefore indicates the consistency between users' social connectivity and their contact patterns in practice.

Moreover, we are also interested in the changes of such social community structure over time. As shown in Figure 1.17, we evaluated the social community structure among mobile users during daytime and nighttime, respectively. Figure 1.17(a) shows that the social community structure during daytime is similar with that in Figure 1.16, and indicates that users' interactions during daytime are the main factor determining social community structure in the network. In contrast, Figure 1.17(b) shows distinct characteristics of the community structure. Basically, the two communities merge with each other during nighttime. This indicates that social correlation among mobile users are more casual during nighttime, so that all the mobile users can be grouped into a large community.

1.6 Related Work

Data access in opportunistic mobile networks can be provided in various ways. Data can be actively disseminated by the data source to appropriate users based on their interest profiles [8]. Publish/subscribe systems [25] are most commonly used for such data dissemination, and they usually exploit social community structures among mobile users to determine the brokers. Caching is another way to provide data access among mobile users. Distributed determination of caching policies or locations for minimizing data access delay has been studied [24, 16], but they generally rely on specific assumptions to simplify the network conditions. In [24], it is assumed that all the mobile devices contact each other with the same rate. In [16], mobile users are artificially partitioned into several classes such that users in the same class are statistically identical. Recent research efforts [9] developed more generic solutions by assuming the heterogeneity of users' contact and interest patterns, and further propose to cache data at specific network locations which can be easily accessed by other users in the network. The freshness of data cached at mobile users has also been taken care of in a fully distributed manner [10].

Recent research efforts have been focusing on exploring the realistic characteristics of mobile network environments and user behaviors for efficient design and implementation of data access systems [21, 12]. [18, 1] focus on investigating the coverage of WiFi network in urban areas, and further improve data accessibility by transferring 3G traffic to WiFi. [19] studied the traffic patterns of commuters in metropolitan subway systems, which is exploited for intentional media sharing. However, none of these systems has been implemented with realistic media content being transmitted among mobile users, and the efficiency of data access is only evaluated via trace-based simulations with synthetic network traffic pattern.

1.7 Conclusion & Future Directions

In this paper, we focused on developing a practical system which provides pervasive data access to smartphone users without support of cellular or Internet infrastructure. We designed and implemented our system on Android-based smartphones, and deployed our system to a number of students at the Pennsylvania State University. Our system is able to efficiently share the media news that users are interested in when they opportunistically contact each other, and can also be used for capturing and analyzing the interest and data access patterns of mobile users.

In the near future, we plan to deploy our system on a large number of smartphones (more than 100), and distribute these phones to voluntary students from various departments of Penn State University. Being different from our system deployment described in Section 1.5, the increase of system scale may significantly change the characteristics of user interests and their data access patterns. We expect to have a large amount of traces which record the realistic data sharing and access behaviors of users in various mobile environments. These traces could provide unique research facility for mobile network researchers, and motivate future research on providing efficient data access with respective to realistic user interests.

We plan to improve the data access efficiency of our system by taking the temporal and spatial variations of user interests into account. In Section 1.3, our system determines the data that a user would be interested in based on his cumulative UIP over a long period of time. However, as shown in Figure 1.15, the interest of a mobile user may vary during different time periods in a day or at different places. For example, a student in Computer Science may be generally interested in information technology, but will more likely be interested in food during lunch time or when he is at a restaurant. The consideration of such temporal/spatial differentiations of user interest leads to more accurate identification of the data that users are interested in, and further improves the efficiency of data access.

We also plan to add security protection functionality into our system to protect mobile users during data access and sharing. We notice that the peer-to-peer data access can be utilized by malicious attackers to distribute malicious code, virus or worms among mobile users, and can also facilitate spammers to spread advertisements or other unwanted information. Particularly, the speed of such spread can be significantly accelerated due to the high population density in the application scenarios described in Section 1.2. We plan to develop efficient distributed authentication methods and network trust architecture to ensure the authenticity and integrity of the shared web data. The development of these security protection methods will also take the social interactions among mobile users into account.

References

[1] A. Balasubramanian, R. Mahajan, and A. Venkataramani. Augmenting Mobile 3G Using WiFi. In *Proceedings of MobiSys*, pages 209–222, 2010.

[2] A. Balasubramanian, R. Mahajan, A. Venkataramani, B. Levine, and J. Zahorjan. Interactive wifi connectivity for moving vehicles. In *Proceedings of ACM SIGCOMM*, pages 427–438, 2008.

[3] C. Boldrini, M. Conti, and A. Passarella. ContentPlace: Social-Aware Data Dissemination in Opportunistic Networks. In *Proceedings of MSWiM*, pages 203–210, 2008.

[4] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker. Web caching and Zipf-like distributions: Evidence and implications. In *Proceedings of INFOCOM*, volume 1, 1999.

[5] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott. Impact of Human Mobility on Opportunistic Forwarding Algorithms. *IEEE Trans. on Mobile Computing*, 6(6):606–620, 2007.

[6] N. Eagle and A. Pentland. Reality Mining: Sensing Complex Social Systems. *Personal and Ubiquitous Computing*, 10(4):255–268, 2006.

[7] K. Fall. A Delay-Tolerant Network Architecture for Challenged Internets. *Proc. SIGCOMM*, pages 27–34, 2003.

[8] W. Gao and G. Cao. User-centric data dissemination in disruption tolerant networks. In *Proceedings of INFOCOM*, 2011.

[9] W. Gao, G. Cao, A. Iyengar, and M. Srivatsa. Supporting cooperative caching in disruption tolerant networks. In *Proceedings of Int’l Conf. on Distributed Computing Systems (ICDCS)*, 2011.

[10] W. Gao, G. Cao, M. Srivatsa, and A. Iyengar. Distributed maintenance of cache freshness in opportunistic mobile networks. In *Proceedings of the 32nd IEEE International Conference on Distributed Computing Systems (ICDCS)*, pages 132–141, 2012.

[11] W. Gao and Q. Li. Wakeup scheduling for energy-efficient communication in opportunistic mobile networks. In *Proceedings of IEEE INFOCOM*, 2013.

[12] B. Han and A. Srinivasan. Your friends have more friends than you do: Identifying influential mobile users through random walks. In *Proceedings of the 13th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 5–14, 2012.

[13] W. Hu, G. Cao, S. Krishanamurthy, and P. Mohapatra. Mobility-assisted energy-aware user contact detection in mobile social networks. In *Proceedings of IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2013.

[14] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot. Pocket switched networks and human mobility in conference environments. In *Proceedings of the 2005 ACM SIGCOMM workshop on Delay-Tolerant Networking (WDTN)*, pages 244–251. ACM, 2005.

[15] P. Hui, E. Yoneki, S. Chan, and J. Crowcroft. Distributed community detection in delay tolerant networks. *Proc. MobiArch*, 2007.

[16] S. Ioannidis, L. Massoulie, and A. Chaintreau. Distributed caching over heterogeneous mobile networks. In *Proceedings of the ACM SIGMETRICS*, pages 311–322, 2010.

[17] M. Kim, D. Kotz, and S. Kim. Extracting a Mobility Model from Real User Traces. *Proc. INFOCOM*, 2006.

[18] K. Lee, I. Rhee, J. Lee, S. Chong, and Y. Yi. Mobile data offloading: How much can wifi deliver? In *Proceedings of ACM CoNEXT*, 2010.

[19] L. McNamara, C. Mascolo, and L. Capra. Media sharing based on colocation prediction in urban transport. In *Proceedings of MobiCom*, pages 58–69, 2008.

[20] M. McNett and G. Voelker. Access and mobility of wireless PDA users. *ACM SIGMOBILE CCR*, 9(2):40–55, 2005.

[21] N. P. Nguyen, T. N. Dinh, S. Tokala, and M. T. Thai. Overlapping communities in dynamic networks: Their detection and mobile applications. In *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 85–96, 2011.

[22] G. Palla, I. Derényi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, 2005.

[23] M. R. Ra, J. Paek, A. B. Sharma, R. Govindan, M. H. Krieger, and M. J. Neely. Energy-Delay Tradeoffs in Smartphone Applications. In *Proceedings of MobiSys*, pages 255–270. ACM, 2010.

[24] J. Reich and A. Chaintreau. The Age of Impatience: Optimal Replication Schemes for Opportunistic Networks. In *Proceedings of ACM CoNEXT*, pages 85–96, 2009.

[25] E. Yoneki, P. Hui, S. Chan, and J. Crowcroft. A socio-aware overlay for publish/subscribe communication in delay tolerant networks. *Proc. MSWiM*, pages 225–234, 2007.