Energy-constrained Wi-Fi Offloading Method Using Prefetching

Yoshihisa Onoue[†], Morihiko Tamai[†], Keiichi Yasumoto[†]
[†] Nara Institute of Science and Technology
Ikoma, Nara 630-0192, Japan Email: see http://ubi-lab.naist.jp/index-e.html

***Abstract*—The explosive growth of mobile data traffic causes immense pressure on the limited spectrum of cellular networks (3G/4G) and the deterioration in the quality of wireless communication. Even though mobile network operators deploy WiFi access points (WiFi APs) to offload the traffic from 3G/4G to WiFi, WiFi connectivity is far from ubiquitous. To increase opportunities of offloading, some methods leveraging delay tolerance have been proposed. These methods, however, cause delay sensitive applications like web browsing and streaming of video and audio to perform poorly. To address this problem, we introduce a WiFi offloading method that uses prefetching. While a user stays at a WiFi area, our method predicts the Web pages that will be requested by the user over 3G/4G after leaving the WiFi area, and prefetches the pages over WiFi. This allows the user to browse prefetched pages instantly without downloading them over 3G/4G. Simulation results show that our method was able to offload approximately 11% of data traffic, suppressing energy consumption within the amount consumed when performing communication only over 3G/4G.**

I. Introduction

Mobile data traffic is growing at a drastic rate [1]. This growth is due to increasing smartphone usage, the increasing number of subscribers, and the rapid development of mobile broadband technologies. This explosive growth of mobile data traffic causes immense pressure on the limited spectrum of cellular networks (3G/4G), and results in the deterioration in the quality of wireless communication. To cope with this problem, mobile network operators deploy WiFi access points (WiFi APs) to offload the traffic from 3G/4G to WiFi. However, because the coverage of a WiFi AP is relatively small and the number of deployed WiFi APs is limited, WiFi connectivity is far from ubiquitous.

To increase the opportunity of WiFi offloading, some existing studies leverage delay tolerance [2], [3], [4], [5]. Since it is not always necessary for applications such as e-mail clients, software updaters, and cloud backup services to immediately communicate with servers when demands for communication arise, these offloading methods refrain from immediate communication with the servers over 3G/4G, and delay the communication until a WiFi connectivity is available. However, the major contributors to the explosive growth of mobile data traffic are delay sensitive applications for browsing text, audio, and video contents on the Web [6]. Their usability may inevitably deteriorate because of these offloading methods. Thus, an offloading method that increases the opportunity of offloading without hurting the application usability is required.

For this purpose, we propose a prefetch-based WiFi offloading method. In this method, while a user stays at a WiFi area, the Web pages (containing text, image, video, and the like) that will be requested over 3G/4G after the user leaves the WiFi area are predicted. These pages are then prefetched in advance over WiFi. Because the prefetched pages are stored in the local storage of the user terminal, the user can immediately start browsing those pages without downloading them over 3G/4G. Because the remaining battery amount of the user terminal is one of the most critical resources, our method maximizes the data amount to be offloaded under the constraint on the energy consumption and adapts the constraint dynamically based on the user's demand for Web pages. Simulation results show that our method offloads approximately 11% of data traffic while satisfying the constraint on energy consumption.

II. Problem Formulation

A. Overview

Prefetching Web pages that the user will not request wastes a certain amount of energy. Thus, in terms of energy consumption, it is undesirable to prefetch all of the Web pages that the user may request. In our method, we introduce a constraint on the energy amount used for prefetching, and maximize the data volume to be offloaded under the constraint. In the following, we give a more formal description of the maximization problem considered in this paper.

B. Definitions and Assumptions

1) User Terminals: We assume that user terminals (e.g., smartphones) are equipped with WiFi and 3G/4G network interfaces, and that they have enough storage space to store prefetched content. We call a storage space that stores prefetched content a *cache*.

2) Applications: Applications for browsing content on the Web, such as Web browsers, RSS readers, and site-specific movie players, are installed in a user terminal, and these applications are implemented using a library that provides the functions of our prefetch-based offloading method.

3) Networks Environment: An area where a WiFi connectivity is available is called a *WiFi area*. While staying at a WiFi area, only WiFi is used for all of communications even if a 3G/4G connectivity is available. An area where only a 3G/4G connectivity is available is called a *3G/4G area*.

4) Web Pages: Each Web page, which consists of one or more types of resources such as HTML, CSS, JavaScript, video, audio, and the like, is a candidate for prefetching. We assume that each Web page is identified by a Uniform Resource Locator (URL).

Let $C = \{c_1, c_2, \ldots, c_n\}$ be the set of Web pages that can be downloaded while a user stays at a WiFi area, and let $c_i.size$ be the size (in bytes) of $c_i \in C$. Let $C^p (\subseteq C)$ be the set of Web pages that are prefetched while the user stays at the WiFi area. After leaving the WiFi area, the user moves to a 3G/4G area, and let C^b be the set of Web pages browsed by the user while staying at the 3G/4G area.

5) Energy Consumption Model: The energy consumed for downloading one byte using the network interfaces of 3G/4G and WiFi are denoted by E^{3g} and E^{wifi}, respectively. We suppose that the actual values of E^{3g} and E^{wifi} are estimated using the method proposed in [7].

C. Problem Formulation

1) Constraint on Energy Consumption: We denote the total number of bytes prefetched while staying at a WiFi area by V^p. The total energy consumed by prefetching is calculated as:

$$E^{wifi} V^p = E^{wifi} \textstyle\sum_{c \in C^p} c.size. \qquad (1)$$

After leaving the WiFi area, the Web pages in C^p are eventually classified into two categories: (i) those that are browsed by the user (i.e., $C^p \cap C^b$) and (ii) those that are not (i.e., $C^p \setminus C^b$). Hereafter, we call the Web pages that are classified into the former as *consumed pages*, and that are classified into the latter as *unconsumed pages*.

Suppose that the consumed pages were not prefetched in advance. In this case, when the user starts to browse those pages in a 3G/4G area, the pages must be downloaded using the 3G/4G interface. We denote the total size of consumed pages by V^b. The total energy consumed for downloading V^b bytes of data via 3G/4G is calculated as:

$$E^{3g} V^b = E^{3g} \textstyle\sum_{c \in C^p \cap C^b} c.size. \qquad (2)$$

When the number of unconsumed pages becomes large, a large amount of energy will be wasted. To avoid this, we introduce the following constraint on the available energy to be used for prefetching:

$$E^{wifi} V^p \quad \leq \quad E^{3g} V^b. \qquad (3)$$

According to [8], the energy consumption of a WiFi interface is one-sixth of that of a 3G interface. Thus, it is possible to satisfy the constraint of inequality (3) by appropriately selecting the Web pages to be prefetched. In inequality (3), because V^b can be only determined after the user moves to the next WiFi area by leaving from 3G/4G area, we use a predicted value for V^b, which is denoted by $V^{b'}$. We will discuss in detail how to determine $V^{b'}$ in Section III. Using $V^{b'}$, inequality (3) can be rewritten as:

$$E^{wifi} V^p \quad \leq \quad E^{3g} V^{b'}. \qquad (4)$$

2) Objective Function: Our objective is to maximize the total data volume to be offloaded (i.e., the data volume retrieved from the cache) while satisfying the constraint on energy consumption specified by inequality (4). Let $c.size'$ be the estimated size of a Web page c, and let $EV(c)$ denote the expected number of bytes to be loaded from the cache in terms of c. $EV(c)$ is calculated as:

$$EV(c) = p^b(c)\, c.size', \qquad (5)$$

where $p^b(c)$, which is called the *browsing probability*, is the estimated probability of c being consumed by the user. Hereafter, we call the value of $EV(c)$ the *expected offloading volume* of c. We consider the following objective function:

$$\begin{array}{ll} \text{maximize} & \sum_{c_i \in C^p} EV(c_i) \\ \text{subject to} & (4). \end{array} \qquad (6)$$

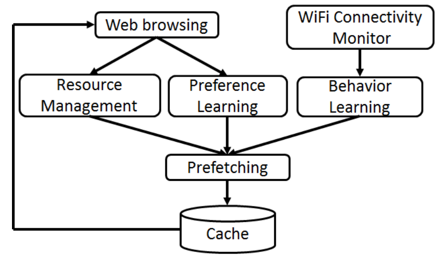

Fig. 1: System Architecture

III. Prefetch-based Offloading Method

A. System Architecture

Fig. 1 illustrates the architecture of the proposed system. The *preference learning module* (described in Subsections III-B and III-C) learns the user's preference to Web pages based on the user's history of the browsed pages. The *resource management module* (described in Subsection III-D) determines the constraint on the available energy amount for prefetching based on the history of the data volume of the browsed pages. The *prefetching module* (described in Subsection III-E) downloads a set of Web pages that will maximize the expected offloading volume under the constraint on the available energy amount.

Before starting prefetching, the system needs to collect the URLs of the Web pages that the user is expected to browse. This process (hereafter called *exploring process*) can be realized through a polling performed periodically while staying at a WiFi area. However, additional energy consumed by the exploring process becomes large if the user stays at the WiFi area for a long time. To reduce this energy overhead, the *behavior learning module* (described in Subsection III-F) predicts the time when the user leaves from the current WiFi area, so that the exploring process and subsequent prefetching can be performed only once per stay at a WiFi area just before leaving.

B. Determining Web Pages to Prefetch

Our system takes a learning period that lasts for a certain duration in days (e.g., 30 days) to learn the Web pages that the user has accessed frequently (e.g., the front page of a news site, a portal site like Yahoo!, and the like) by monitoring the user's browsing history. We call these pages *root pages*. After the learning period, while staying at a WiFi area, the system accesses the root pages by executing the exploring process and collects URLs appearing in those pages (e.g., URLs of individual news articles). Then, the system selects the pages to be prefetched from the collected URLs.

C. Calculating Expected Offloading Volume

Through the exploring process, a set of candidate pages for prefetching are collected. By evaluating each collected page in terms of its browsing probability, the system decides the pages to be prefetched. In order to calculate the expected offloading volume of c (i.e., $EV(c)$ in equation (5)), both the browsing probability of c and the estimated size of c are required. In the following, we describe how to calculate these values.

1) Calculating Browsing Probability: The browsing probability of c is represented by the product of the probability that a root page (e.g., the front page of a news site) containing the URL of c is accessed by the user and the probability that the user browses the page c (e.g., a news article). Let $p(c^r)$ denote the probability that the user accesses a root page c^r at least once while staying in a 3G/4G area, and let $p(c|c^r)$ denote the probability that the user accesses a Web page c linked from c^r. Then, the browsing probability of c is calculated as:

$$p^b(c) = p(c^r)\, p(c|c^r). \tag{7}$$

In order to calculate $p(c|c^r)$, we adopt the method proposed in [9]. Using this method our system learns the keywords that the user is interested in by analyzing the anchor texts that the user clicked during Web browsing. After the learning period (e.g., 30 days), our system starts to perform prefetch-based offloading. When the user stays at a WiFi area, the system explores all root pages and extracts all anchor texts appearing in the root pages. Then, the system compares the anchor texts with keywords one by one. Finally, for each Web page appearing in c^r, the system can calculate the browsing probability.

2) Estimating the size of a Web page: The size of a Web page varies depending on the type of content (e.g., news articles, video content, and the like). We regard a root page as a collection of the links to the same type of Web pages. Thus, during the learning period, our system calculates the average size of a Web page for each root page, and this size is used as the estimated size of a Web page linked to the corresponding root page.

D. Determining the Data Volume to be Prefetched

The constraint on available energy amount for prefetching, which is specified by inequality (4), leads to the following equivalent inequality:

$$V^p \leq \frac{E^{3g}}{E^{wifi}} V^{b'}. \tag{8}$$

This gives the constraint on the data volume that can be downloaded by prefetching over WiFi. In order to determine the right hand side of inequality (8), $V^{b'}$, which is the predicted value of V^b, must be determined. We derive $V^{b'}$ from the average value of V^b as follows. First, whenever the user stays in 3G/4G areas, the system records the URLs and sizes of the Web pages accessed by the user in the areas. Next, the system compares these URLs with the URLs that were obtained during the exploring process. Then, the system estimates the total data volume of Web pages that could be prefetched while at a WiFi area. The system repeats these steps, and calculates the average value of the total data volumes recorded during the last certain period of time. Finally, the system uses it as the predicted value of V^b.

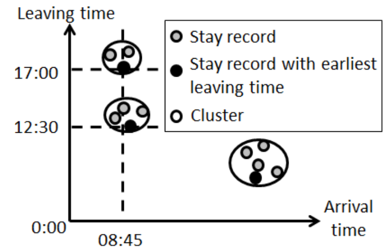

Fig. 2: Learning of the user's behavior patterns

E. Maximizing Expected Offloading Volume

As stated in Subsection II-C2, the system selects the optimal set of Web pages that maximizes the total expected offloading volume within the constraint on the available energy amount. This combinatorial optimization problem is regarded as a knapsack problem: the volume of the knapsack corresponds to the limitation of the data volume that can be downloaded by prefetching (i.e., the right hand side of inequality (8)); the weight of each item to be put in the knapsack corresponds to the estimated size of each Web page; the value of each item corresponds to the expected offloading volume of each Web page. We can solve this problem using existing algorithms for the knapsack problem such as dynamic programming.

F. Determining the Time to perform Prefetching

As described in Subsection III-A, to avoid the energy overhead caused by frequent execution of the exploring process, our system predicts the time when the user leaves the current WiFi area and executes the exploring process only once at a certain time point before leaving from the WiFi area.

Because people's daily schedules are fixed to some extent, the time to enter a specific location and the time to leave the location will not change dramatically depending on the day. We focus attention on this routine of behavior. Every time the user enters into and leaves from a WiFi area, our system records the arrival time and the leaving time and stores these timestamps for a certain period of time. We call each pair of the timestamps a *stay record*. In order to learn the relationship between arrival time and leaving time, the system performs a clustering algorithm to group stay records obtained during the last certain period of time on a two-dimensional plane as shown in Fig. 2. Because it is difficult to determine the number of clusters in advance, we use Mean Shift Clustering method, which does not require the number of clusters to be pre-determined.

Prefetching must be finished before the user leaves the current WiFi area. Thus, when the user enters a WiFi area at a time (say t_0), the system finds a cluster (say S_i) that contains two members whose timestamps of the arrival time (say t_1 and t_2) satisfy $t_1 \leq t_0 \leq t_2$. Then, the leaving time of the user is predicted as the leaving time of the stay record contained in S_i with the earliest leaving time. If there is no such cluster, the system fails to predict the leaving time, and prefetching is not performed. Moreover, if there are multiple clusters that satisfy the above condition, the system performs the exploring process and subsequent prefetching before each predicted leaving time. For example, in Fig. 2, if the user arrives at a WiFi area at

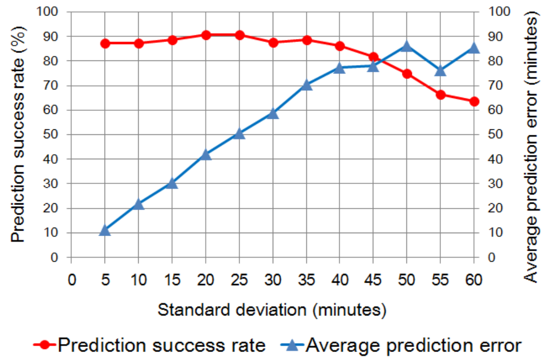

Fig. 3: Prediction accuracy of leaving time

about 8:45 am, then the predicted leaving times are obtained as 12:30 pm and 17:00 pm.

IV. Evaluation

In this section, we report the simulation results on the performance of the proposed offloading method. We first examine how accurately our method predicts the time when a user leaves from a WiFi area. After that, we examine how much data volume our method can offload as well as the results of consumed energy.

A. *Prediction Accuracy of Leaving Time from WiFi Area*

In this simulation, we evaluate the prediction accuracy of the leaving time from a WiFi area. For this purpose, we measure the success rate of the prediction of the leaving time. For successful prediction, it is necessary that there is at least one cluster found for a given arrival time as described in Subsection III-F, and that the user actually leaves at a certain time point after the predicted leaving time. In addition, we measure the prediction error, i.e., the difference between the predicted leaving time and the actual leaving time.

1) Simulation Scenario: We simulate a user who enters a WiFi area at 9:00 am on average and leaves the WiFi area at 17:00 pm on average. On each day, the arrival time and the leaving time are slightly varied by adding a random noise having a normal distribution. Note that the smaller the standard deviation of the normal distribution is, the more punctual the user becomes. We set the learning period to 30 days. We measure the performance of the prediction in three days after the learning period.

2) Simulation Results: Fig. 3 shows the results obtained by varying the standard deviation of the normal distribution. The smaller the standard deviation, the higher the success rate of prediction. As the standard deviation becomes larger, the number of clusters and their sizes become larger. For this reason, in most of the cases, the prediction succeeds. However, because the probability that the user leaves from the WiFi area earlier than the predicted time becomes higher, the success rate decreases slightly. The average prediction error increases with the increase of the standard deviation. From these results, we conclude that the leaving time can be predicted accurately, if the user is punctual (e.g., office workers and students who commute by public transportation on weekdays).

B. *Performance of Offloading*

To evaluate the performance of offloading, we measure the ratio of the offloaded data volume to the total data volume downloaded at the 3G/4G area. We also measure the consumed energy by prefetching.

1) Simulation Scenario: Before the simulation, we measured E^{wifi} and E^{3g} by PowerTutor [7] using SONY XPERIA acro HD. Based on the result, we set the ratio of E^{wifi} to E^{3g} to be $1\colon 4.5$. To simulate the update interval of Web pages and to generate the size of each Web page, we monitored four blog sites and measured the average update interval and the average size of each Web page. We regard the front pages of the four Web sites as root pages in the simulation. The visiting probabilities of the root pages and the browsing probability of each Web page contained in the root pages are determined randomly by the uniform distribution. We use the same behavior patterns for arriving and leaving the WiFi area as in the previous simulation. We set the standard deviation (described in Subsection IV-A1) to 60 minutes. We take seven days as a learning period for $V^{b'}$. In this simulation, we assume that the user browses a constant number of Web pages each day. We measure the performance of offloading three days after the learning period.

We evaluate our method by comparing it with other methods that prefetch the Top N pages in terms of the expected offloading volume every time the user enters a WiFi area. We call these methods *Top-N methods*, and measure the performance of Top-10, Top-50, and Top-100.

2) Simulation Results: Fig. 4 shows the ratio of the offloaded data volume to the total data volume that is consumed (i.e., the sum of the sizes of the Web pages browsed) by the user at the 3G/4G area. On average, Top-50 and Top-100 methods achieved 12.4 % and 19.4 % of offloading, respectively, while the proposed method achieved 10.8 % of offloading. Our proposed method tries to satisfy the constraint on energy consumption, thus, the fraction it offloads is smaller than those of Top-50 and Top-100. However, as mentioned later, by using Top-50 and Top-100, the energy consumption is dramatically increased compared to a case in which prefetching is not performed. Moreover, as the number of browsed pages increases, the offloaded volumes of the Top-N methods decrease, while the proposed method maintains the performance at certain level. This is because the Top-N methods prefetch only a fixed number of pages, and the ratio of the offloaded volume becomes smaller when the number of browsed pages increases. On the other hand, our proposed method dynamically adapts the offloading volume based on the user's demand (i.e., the number of the browsed pages).

Fig. 5 shows the energy consumption ratio. *3G* in the figure is a method that does not prefetch at all, and the Web pages requested by the user at 3G/4G areas are downloaded using 3G. When the number of browsed pages is 10, the Top-100 method consumes more than twice as much energy as 3G, and the Top-50 method consumes more than one and a half times as much energy as 3G. In the top-N methods, when the difference between the number of browsed pages and the number of pages prefetched is large, they tend to consume a large amount of energy. On the other hand, in the proposed method, because of the dynamic adaptation of the data volume

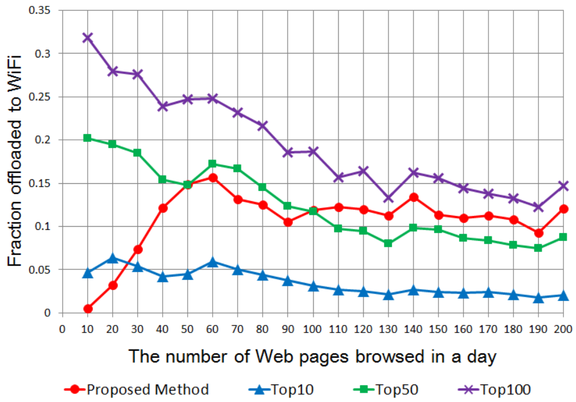

Fig. 4: The ratio of offloaded data volume to the total data volume that is consumed by user at 3G/4G area

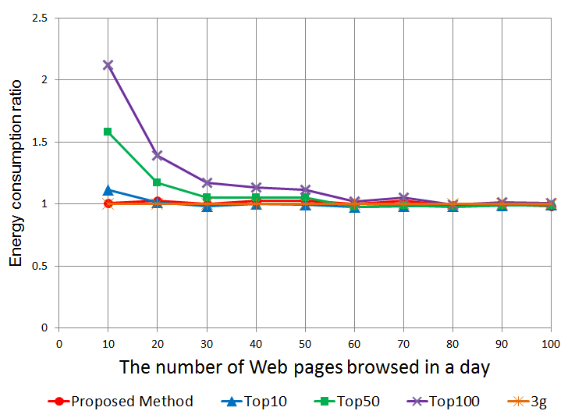

Fig. 5: Energy consumption in comparison to the method that uses only 3G

to be prefetched, the energy consumed by the proposed method is approximately 2% less than that of 3G on average.

V. Related work

Several recent studies have proposed WiFi offloading methods levaraging delay tolerance [2], [3], [4], [5]. These methods refrain from immediate communication over 3G/4G, and delay the communication until a WiFi connectivity is available. According to [10], the longer the delay is, the more data can be offloaded. Thus, existing methods are effective for delay tolerance applications such as email and file transfer. On the other hand, such communication delay can spoil the usability of delay sensitive applications such as Web browsing and streaming contents. Our proposed method , in contrast, predicts and prefetches Web content over WiFi. Thus, it can offload the data traffic while improving application response time.

Since the bandwidth of cellular networks is limited compared to that of fixed broadband networks, cellular networks tend to have a large communication delay. To mitigate the delay and to improve the application's responsiveness, several prefetching methods that use not only WiFi but also 3G/4G networks have been proposed [11], [12]. On the other hand, our method aims to improve offloading performance from 3G/4G to WiFi.

A system to predict the length of stay at WiFi hotspots is proposed in [13]. The system monitors users' behavior using their smartphones, and detects users who are likely to leave from a WiFi hotspot in a short time period. The system, however, causes the user terminal to consume energy quickly by monitoring of the user's behavior and by transmitting the sensed data from the terminal to the server. In contrast, our proposed method only monitors the WiFi connectivity to learn and to predict the behavior pattern of the user. Thus, in terms of the monitoring overhead, we believe that our approach has almost no impact on the battery lifetime.

VI. Conclusion

In this paper, we introduce a WiFi offloading method that predicts the Web pages that will be requested at 3G/4G areas, and prefetches them while WiFi is available. This method not only offloads data traffic from 3G to WiFi but also reduces the delay caused by the downloading of the Web pages. The simulation results show that our method offloaded approximately 11% of data traffic while satisfying the constraint on the energy consumption.

Acknowledgment

This work was partly supported by JSPS KAKENHI Grant Number 25730059.

References

[1] Cisco, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2012–2019," http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.html.

[2] A. Balasubramanian, R. Mahajan, and A. Venkataramani, "Augmenting Mobile 3G Using WiFi," In *Proc. of MobiSys'10*, pp. 209–222, 2010.

[3] V. A. Siris and D.Kalyvas, "Enhancing mobile data offloading with mobility prediction and prefetching," In *Proc. of MobiArch'12*, pp. 17–22, 2012.

[4] J. Eriksson, H. Balakrishnan, and S. Madden, "Cabernet: Vehicular Content Delivery Using WiFi," In *Proc. of MobiCom'08*, pp. 199–210, 2008.

[5] S. Dimatteoy, P. Huiy, B. Hanyz, and V. O.K. Li, "Cellular Trafc Ofoading through WiFi Networks," In *Proc. of MASS'11*, pp. 192–201, 2011.

[6] Ericsson, "TRAFFIC AND MARKET DATA REPORT," http://hugin.info/1061/R/1561267/483187.pdf.

[7] L. Zhang, B. Tiwana, Z. Qian, Z. Wangand, R. P. Dick, Z. M. Mao, and L. Yang A, "Accurate online power estimation and automatic battery behavior based power model generation for smartphones," In *Proc. of CODES/ISSS'10*, pp. 105–114, 2010.

[8] N. Balasubramanian, A.Balasubramanian, and A. Venkataramani, "Energy consumption in mobile phones: a measurement study and implications for network applications," In *Proc. of IMC'09*, pp. 280–293, 2009.

[9] P. Venketesh and R.venkatesan, "Adaptive Web Prefetching Scheme using Link Anchor Information," *J. of Applied Information Systems*, Vol. 2, No. 1, pp. 39–46, 2012.

[10] K. Lee, I. Rhee, J. Lee, and S. Chong, "Mobile Data Offloading: How Much Can WiFi Deliver?," In *Proc. of CoNext'10*, pp. 425–426, 2010.

[11] B. D. Higgins, J. Flinn, T. J. Giuli, B. Noble, C. Peplin, and D. Watson, "Informed mobile prefetching," In *Proc. of MobiSys'12*, pp. 155–168, 2012.

[12] D. Lymberopoulos, O. Riva, K. Strauss, A. Mittal, and A. Ntoulas, "PocketWeb: instant web browsing for mobile devices," In *Proc. of ASPLOS'12*, pp. 1–12, 2012.

[13] J. G. Manweiler, N. K. Santhapuri, R. R. Choudhury, and S. Nelakuditi, "Predicting Length of Stay at WiFi Hotspots," In *Proc. of Infocom'13*, pp. 3102–3110, 2013.