Towards Ubiquitous Computing with Call Prediction *

Santi Phithakkitnukoon[†○] *santi@mit.edu*
Ram Dantu[‡] *rdantu@unt.edu*
[†]SENSEable City Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA
[‡]Department of Computer Science and Engineering, University of North Texas, Denton, TX, USA

With the long-awaited era of the pervasive computing approaches, the handheld devices such as personal mobile phones begin to evolve into ubiquitous computing devices. At this early stage of the evolution, we propose a model of a call predictor based on the naïve Bayesian classifier. As an incoming call predictor, our model makes use of the user's call history to generate a list of numbers/contacts that are the most likely to be the callers within the next hour. On the other hand, when the user wants to make an outgoing call (e.g., user flips open the phone or unlocks the phone, etc.), the outgoing call predictor generates a list of number/contacts to be called. Our model has been evaluated with the real-life call logs and it shows a promising result in accuracy.

I. Introduction

With the rapid development of telecommunication technologies and the fast-growing number of users on the networks, the mobile phone has moved beyond being a simple phone. It has become a mobile workstation and integrated into many parts of people's lives. At this early stage of the pervasive-computing era, the handheld devices become the precursors to a phase of ambient computing that is always on, personalized, context-sensitive, and highly interactive.

Mobile (personal) phones record the history of our lives in the form of the call logs. By utilizing call logs in computing human (user) behavior, we can enhance the usability of the phone as it is becoming more than just a voice communication device and evolving into an intelligent assistant to its user.

In this paper, we design and evaluate a model that makes use of the call logs to predict incoming as well as outgoing calls. With our model, the personal phone will become even more personal as it learns and recognizes its user's calling behavior as well as the associated users' (callers' and callees') in order to provide the most accurate prediction of the future caller and callee for the user. In this way, the mobile phone becomes more personalized and sensitive to the user's context and needs.

*This work is supported by NSF under grants CNS-0751205, CNS-0821736, CNS-0619871, and CNS-0551694. ○This work was done while the author was a Ph.D. student at the University of North Texas

II. Call Prediction

Predicting incoming calls can be very useful for planning and scheduling (*e.g.,* it can be used to avoid unwanted calls and schedule time for wanted calls). People normally check weather forecast before leaving homes and watch for signs of approaching storms to prepare and schedule their days accordingly. Knowing what is coming next gives us supplemental time to think, prepare, and optimize our solutions. We believe that incoming call prediction can be useful for daily planning and it may become an important element as an initiative decision support for our daily life scheduling.

Quite often in our daily lives, we find ourselves in a situation where we wish to know who will be calling in the next hour so we could schedule (plan) things out accordingly. In many occasions, we know for certain that we will not be available to accept any incoming calls over the next hour (*e.g.,* having a flight, attending a class, having a meeting) thus we wish to know who will be calling during the next hour so we could perhaps make a call to the persons to inform of our next-hour schedule as we do not wish to miss any important future calls, which could be too important calls to miss.

Likewise, predicting outgoing calls can be useful for many applications such as enhancing mobile phone's usability by providing a list of the most likely contacts/numbers to be dialed when user wants to make a call. Such that it reduces the searching time as well as enable better life synchronization for the user.

Our call predictor makes use of the user's call his-

Figure 1: CPL user interface.

tory *e.g.,* call identifications, time of calls, day of calls, frequency of calls, and last received/made numbers, to build a probabilistic model of calling behavior. The calling behavior model is then used to generate a list of numbers/contacts that are the most likely to be the callers for the next hour (as an incoming call predictor) or a list of numbers/contacts to be dialed (as an outgoing call predictor).

The list can be presented to the user in a number of different ways for different purposes. We envisage the predictor as a "Call Predicted List (CPL)," *i.e.,* a list that anticipates the most likely callers/callees and gives these numbers/contacts higher precedence on the list. Figure 1 shows an example of the envisaged CPL where the most likely callers/callees are listed higher on the list.

As an incoming call predictor, CPL can be integrated with voice spam detector [1] and nuisance detector [2] to create a Call Firewall that proactively manages the incoming calls based on the preconfigured set of rules by keeping the unsolicited calls away while allowing wanted calls to either ring the phone or be forwarded to voice mail. On the hand, amalgamating the outgoing-call prediction functionality of CPL with nuisance detector [2] and event calendar can create a useful application such as Call Reminder that provides an automatic reminder for placing a call based on probability of making a call to a particular person, nuisance level of the user, and associated events.

III. Call Prediction Framework

When that cell phone rings, how often do we make a guess on who the caller might be? More often than not, but if we do make a guess, we are usually right. We often base this estimation on the caller's call history as well as our call history with the caller.

Each caller exhibits a unique calling pattern which can be observed through history of "time of the calls" *i.e.,* we normally expect a call from someone who has a history of making several calls during a particular time period of the day. For example, your spouse likes to call you while you are driving to work in the morning therefore when your phone rings while you are on the way to work you are likely to guess that it is a phone call from your spouse. The pattern can also be observed from "day of the calls," for example, your friend, John, has made several calls to you on every Tuesday because it is his day off, therefore when your phone rings on Tuesday, the first person that comes to mind is John. Likewise, the person who has made the most "number of calls" to you (regardless of time and day) among other callers is also the person whom you most anticipate the calls from. Receiving a call is also influenced by the "reciprocity" or call interaction between the user and the caller. For example, you may anticipate a phone call from a specific person based on your last phone conversation with the person (*e.g.,* "call me when you get home" or "call me same time tomorrow" or "I'm busy right now, call me back in an hour"). This reciprocity may sequentially lead to a later call received from the person caused by your initiative. For example, you decide to make a call to an old friend to whom you have not called for a long time, and later you start to receive calls from this old friend. Another example, you make a call to your mother to get some advice during the night (assume that normally you do not make or receive calls from her during this time), and then you receive calls from your mother later on during that night. These are the examples that actually happen in our everyday lives as a phone user. Understanding the actual human behavior towards phone usage gives the CPL an intelligence to assist its user effectively and in the same time makes the smart phone smarter.

III.A. Datasets

Predicting future calls is a challenging task. It requires a design of model that should incorporate mechanism for capturing and learning the caller/callee's calling patterns. Calling patterns can be extracted from the call logs, which can be obtained from a variety of

sources. For example, they may be collected by a network or service operator for billing purposes or they may be captured directly on device such as a mobile phone or on a software application such as a VoIP softphone. In our current implementation, we use two sets of real-life call logs of 30 combined users with nearly 4,00 callers/calees and over 56,000 call activities. Our first dataset consists of three-month call logs of 20 individual mobile phone users, which were collected at University of North Texas (UNT), Denton, during summer of 2006. These 20 individuals were faculty, staff, and students. These call logs were collected as part of the Nuisance Project, where Kolan *et al.* [2] studied the nuisance level associated with each phone call. The details of the data collecting process are given in [3]. Our second dataset consists of three-month call logs of ten mobile phone users, which were collected during summer of 2008 at UNT. These ten subjects were also faculty, staff, and students.

As part of the data collecting process (for both datasets), each user downloaded three months of detail telephone call records from his/her online accounts on the mobile phone service provider's website. Each call record in the dataset had 5-tuple information as follows where an example call record is shown in Fig. 2.

- Date: date of the call
- Start time: start time of the call
- Type: type of the call *i.e.,* "Incoming" or "Outgoing"
- Call ID: caller/callee identification
- Talk Time: duration of the call (in minutes)

Date	Start time	Type	Call ID	Talk time
3/11/2007	2:28PM	Outgoing	123-4567890	2
3/11/2007	5:31PM	Incoming	888-8888888	11
3/11/2007	8:12PM	Incoming	999-9999999	6
…	…	…	…	…

Figure 2: An example of a call record. Note that Call ID's have been modified for privacy reason.

III.B. System Overview

The call record shown in Fig. 2 is subject to preprocessing to extract features or information about "time of the calls" (day and hour), "total call count," and "reciprocity". The pre-processed call records are eventually fed into the classifier to be ingested. Classifier then outputs a list of phone numbers ordered by the likelihood of the number being the next-hour caller (as an incoming call predictor) or the dialing number (as an outgoing call predictor). The basic system overview is shown in Fig. 3.

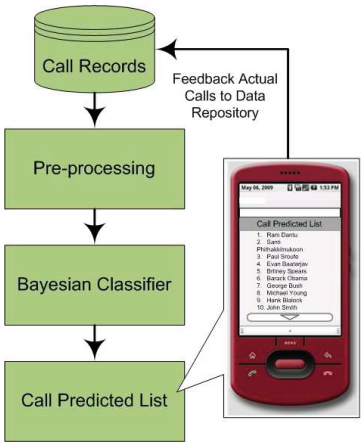

Figure 3: Basic system overview.

III.C. Inference Engine

With the same framework, the CPL can function as an incoming call predictor and an outgoing call predictor with just a simple modification in the direction of the calls (*i.e.,* incoming and outgoing) in the analysis. Therefore, let us consider the CPL first as an incoming call predictor.

Our inference engine is driven by a Bayesian classifier, which has two modes of operation; training and predicting. During the training, classifier ingests the pre-processed call logs and constructs four hash tables that primarily contain call counts of the corresponding features. The first table maps each unique telephone number (or caller identifier) to a count of calls received for each day of the week as shown in Fig. 4.

Caller ID	Day of week						
	1	2	3	4	5	6	7
123-4567890	5	23	6	2	11	0	1
888-8888888	6	0	0	1	0	33	4
999-9999999	0	0	11	8	21	7	8
…	…	…	…	…	…	…	…

Figure 4: An example of a hash table for day of the week.

The second table maps each unique telephone number (or caller identifier) to a count of calls received for each hour of the week as shown in Fig. 5.

The third table maps each unique telephone number (or caller identifier) to the total number of calls received as shown in Fig. 6.

Caller ID	Hour of day						
	0	1	2	...	21	22	23
123-4567890	0	0	0	...	9	3	1
888-8888888	2	0	0	...	15	8	2
999-9999999	0	0	0	...	27	9	0
...	...	...	...	...	...	...	...

Figure 5: An example of a hash table for hour of the day.

Caller ID	Call count
123-4567890	118
888-8888888	121
999-9999999	157
...	...

Figure 6: An example of a hash table for cumulative frequency of calls.

Quantify the "reciprocity" is not quite trivial. Having no knowledge about the context of the previous phone calls of the user, it is difficult to identify which outgoing calls would influence the future incoming calls. Nevertheless, the recent received calls can be linked to the user's calling behavior. These recent received calls are typically stored in the "last dialed calls" list (normally a list of last 20 outgoing calls) where the lower order corresponds to more recent dialed number (*e.g.,* "1" is the most recent dialed number, "20" is the least recent dialed number). Thus the same number/contact can occupy in more than one position on the list. Clearly the numbers/contacts on the list are pushed down one position when a new call is received. Based on the position on this list and its corresponding number of times that actual incoming caller was listed on that position, the likelihood of receiving a call can be estimated. For example, suppose the current statistic (hash table) shows that position "3" of the list has the most counts, it implies that the number/contact that is on position "3" of the current "last dialed calls" list has the highest likelihood of being the next caller. Therefore, the fourth hash table maps each position on the "last dialed calls" list to the count of the calls received as shown in Fig. 7.

Once the input call records have been ingested and the hash tables generated, the classifier is considered trained. With the classifier trained on a set of representative call records, it is then ready to be used in predicting mode. The classifier is given a target day of week, hour of day, total call count, and current last-20-dialed-calls list, and uses the calling behavior model to estimate the likelihood of the user receiving each of the telephone numbers (or caller identifiers) seen in the training data. Clearly, the classifier can only make predictions for numbers that it has already seen.

Position on last-20-dialed-calls	Number of times when a call is received and its caller is listed on corresponding position on last-20-dialed-calls list
1	69
2	45
3	71
...	...
...	...
19	3
20	8

Figure 7: An example of a hash table for caller's position on the last-20-dialed-calls list.

A likelihood metric then is calculated for each number seen by the classifier and the numbers are then sorted in descending order of likelihood of being received. If the caller's behavior has a high degree of predictability (*i.e.,* they tend to make calls consistently to user at this certain time of the day, or in this particular day of the week, or after some number of calls from the user), then it is expected that the number is likely to be listed towards the top of the list. If there is a tie *i.e.,* several numbers end up with the same value of likelihood, then the classifier list them in the alphanumerical order.

Our inference engine is based on the Naïve Bayesian Classifier, which is a simple probabilistic classifier based on Bayes' theorem with independence assumptions. In our case, we want to compute the likelihood of each number (T_n) being received given that the day of the week (D_x), hour of the day (H_y), the current last-20-dialed-calls list (L_z), and total call count (F_n). Bayes rule [4] of conditional probability is given by Eq. 1.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}, \tag{1}$$

where $P(A|B)$ is the *posterior* probability, which is the probability of the state of nature being A given that feature value B has been measured. The *likelihood* of A with respect to B is $P(B|A)$, which indicates that other things being equal, the category A for which $P(A|B)$ is large is more "likely" to be the true category. $P(A)$ is called *prior* probability. The *evidence* factor, ***P(B)***, can be viewed as a scale factor to guarantee that the posterior probabilities sum to one.

We use this rule to obtain the probability of a number being received given a specific hour of the day, day of the week, current last-20-dialed-calls list, and total call count, as given by Eq. 2.

$$P(T_n|D_x, H_y, L_z, F_n) = \frac{P(D_x|T_n)P(H_y|T_n)P(L_z|T_n)P(F_n|T_n)P(T_n)}{P(D_x, H_y, L_z, F_n)}, \quad (2)$$

With the Naïve Bayesian classifier, a well known issue occurs when a particular attribute value doesn't occur in conjunction with every class value in the training data. In our case, the attributes are D_x, H_y, and L_z. The class values are the incoming telephone numbers (callers). The computed probability of a number being received at a particular time will be zero if the training data has no instance of that number being received during either the specified hour or the specified day.

A solution to this problem is to start all the call counts in the Hash tables for day-of-week and hour-of-day at one instead of zero and defining some normalizing factors in the resulting computations. This is not an issue for the F_n since there must be at least one call count for any seen incoming call. For L_z, this is sort of an issue since only those numbers/contacts that are on the current last-20-dialed-calls list are considered. A solution for this case is to assign the lowest call count of the position on the last-20-dialed-calls list (hash table) to those phone numbers that are not on the current last-20-dialed-calls list. Therefore, those numbers that are not on the current last-20-dialed-calls list will have the same probability of being received as the lowest probability of the number on the current list being received. There is also a possibility of one telephone number occupies more than one position on the current last-20-dialed-calls list. In this situation, the highest call count among all positions occupied by that telephone number is assigned to it.

Adopting this approach, we compute the likelihood of the caller T_n being received, given D_x, H_y, L_z, and F_n, by Eq. 3.

$$L(T_n|D_x, H_y, L_z, F_n) = \left(\frac{C(T_nD_x)+1}{C(T_n)+7}\right) \cdot \left(\frac{C(T_nH_y)+1}{C(T_n)+24}\right) \cdot \left(\frac{C(T_nL_z)}{C(L)}\right) \cdot \left(\frac{C(T_nF_n)}{C(T_n)}\right), \quad (3)$$

where $C(T_nD_x)$ is the call count from the caller T_n on day D_x ($x = 1, 2, 3, \ldots, 7$), $C(T_nH_y)$ is the call count from the caller T_n during hour H_y ($y = 0, 1, 2, \ldots, 23$), $C(T_nL_z)$ is the call count from the caller T_n when T_n's position on the current last-20-dialed-calls list is L_z ($z = 1, 2, 3, \ldots, 20$), $C(T_nF_n)$ is the total call count from caller T_n ($n = 1, 2, 3, \ldots, N$, where N is the total number of callers that have made at least one call to the user), $C(L)$ is the total call count of all position on the list (sum of the second column of hash table in Fig. 7), and $C(T_n)$ is the total call count from caller T_n over the entire training data.

IV. Performance Analysis

In this section, the CPL is evaluated with the actual call logs of 30 mobile phone users as described in Section 3. The first two months (approximately 60 days) of call logs are used to train the CPL and the rest of the call logs are assumed to be the future observed call activities to test the performance of the CPL by observing for each call received what position that actual caller has in the predicted list. If the CPL performed perfectly, one clearly would expect the actual caller to be at the top of the predicted list. Generally, such performance is not achievable, but one might expect that the actual caller would tend to appear earlier rather than later in the list.

IV.A. Improvement over Conventional Last-Received-Calls List

The overall performance of the CPL based on these 30 mobile phone users is shown in Fig. 8 where the its accuracy is measured by the average percentage of the actual callers listed within the predicted list as the length of the list varies from 1 to 20. One may be curious to find out that if the conventional last-20-received-calls list, which already exists in today's mobile phone, is used as a call predicted list. How well can it perform? Will it performs better than our CPL? *The comparison is illustrated in Fig. 8 where it can be seen clearly that our CPL outperforms the last-20-received-calls list (if used as predictor) with nearly 30% better accuracy.*

IV.B. Impact of Caller Population

The CPL would always predict the caller correctly, if there was only one caller. In general, the population of the callers increases *e.g.,* meeting new friends, signing up with a new group, being on telemarketers' list, etc. This increasing number of caller population may affect the accuracy of the CPL *i.e.,* it becomes harder to guess the correct number from a larger callers pool.

To illustrate the impact of the increase of the caller population on the CPL, we randomly select one user in our datasets as an example shown in Fig. 9 where the vertical axis represents the accuracy of the CPL, and horizontal axis represents the cumulative caller

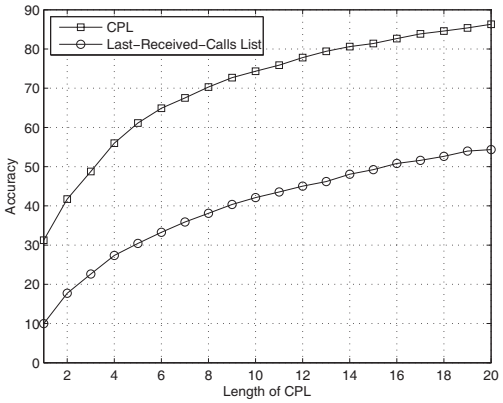

Figure 8: Overall performance of the CPL comparing to the conventional Lat-20-Received-Calls list.

population that continues to increase from 41 callers to 70 callers. It shows that the accuracy decreases dramatically as the caller population becomes larger for different length of the list (L = 1, 5, 10, 15, 20). The accuracy drops with relatively higher rate for the shorter length of the list as one may expect.

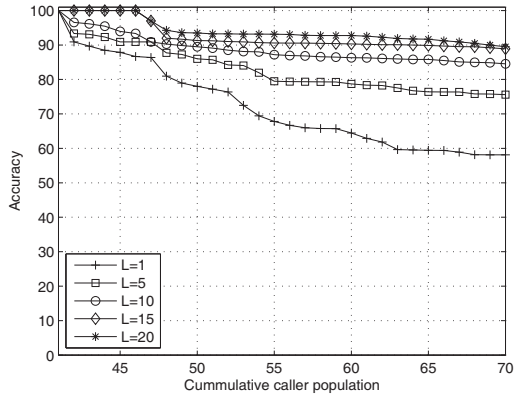

Figure 9: A demonstration of the impact of the increasing cumulative caller population on the accuracy of the CPL.

IV.C. Impact of New Callers

In the meanwhile, the new callers or the first-time callers (whose call received for the first time) also have a negative impact on the performance of the CPL. This may be a bigger issue for those users who are more social and those who are unfortunately on numerous telemarketers' lists. This is a voice spam problem, which is expected to increase severely, especially in the VoIP networks where the cost of communication is extremely low with the absurdly large IPv6 address (can supports 2128 addresses). To demonstrate the impact of the new callers, we examine the accuracy of the CPL without considering the new callers *i.e.,* if the caller is the first-time caller then it is not taken into account for the accuracy computation. After the first call however the caller will be recognized and taken into account for accuracy computation as normal. *It can be seen from Fig. 10 that the accuracy of the CPL is indeed improved about 10% as the new callers are not considered.*

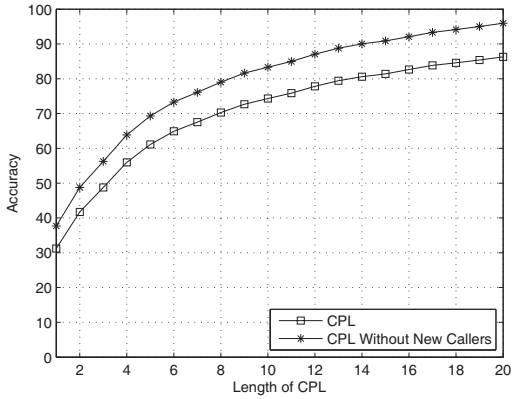

Figure 10: Overall performance of the CPL with and without considering the new callers.

If we modify our definition or criterion for the new callers by redefining the new caller to be the caller who has called C times in the past, then we observe that as variable C increases the accuracy of CPL also increases accordingly (shown in Fig. 11). This unsurprising result implies that the CPL can predict more accurately for the callers whose behaviors have been learned for a longer period of time.

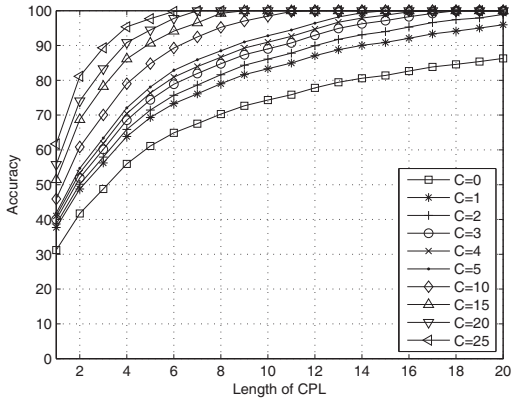

Figure 11: The impact of the new callers to the accuracy as the criterion of new caller (C) varies from 0 to 25.

IV.D. Impact of Mobile Social Closeness

We can further extend the concept of the new callers to infer the "social closeness". The number of incoming calls alone can only be used to quantify the social closeness to some extent. In social science, the social closeness of people has been discussed and found that it can be based on the amount of time and the intensity (frequency) of communication [5][6]. Granovetter [5] suggests that the time spent in a relationship and the intensity along with the intimacy and reciprocal services form a set of indicators for social tie. Marsden and Cambell [6] evaluate the indicators and predicators of strength (tie) described by Granovetter [5] and conclude that "social closeness" or "intensity" provides the best indicator of strength or tie.

In mobile social network, the amount of time and the intensity of communication can be measured by the call duration (talk time) and the call frequency (number of phone calls).

In our daily life, we communicate with people in the mobile network at different instances. These people constitute our mobile social network. Based on amount of time and intensity of communication with these people, our mobile social network can be divided into three broad groups:

Group 1: Socially Closest Members – These are the people with whom we maintain the highest socially connectivity. Most of the calls we receive, come from individuals within this category. We receive more calls from them and we tend to talk with them for longer periods. Typically, the face-to-face social tie of these people is family member, friend, and colleagues.

Group 2: Socially Near Members – People in this group are not as highly connected as family members and friends, but when we connect to them, we talk to them for considerably longer periods. Mostly, we observe intermittent frequency of calls from these people. These people are typically neighbors and distant relatives.

Group 3: Socially Distant Members – These individuals have less connection with our social life. These people call us with less frequency. We acknowledge them rarely. Among these would be, for example, a newsletter group or a private organization with whom we have previously subscribed. This group also includes individuals who have no previous interaction or communication with us. We have the least tolerance for calls from them *e.g.,* strangers, telemarketers, fund raisers.

We quantitatively define the social closeness between user i and user j from the user i's perception ($S(i,j)$) by Eq. 4.

$$S(i,j) = \sqrt{(1 - F(i,j))^2 + (1 - T(i,j))^2}, \quad (4)$$

where $F(i,j)$ is the normalized call frequency (normalized to the maximum call frequency among all users with whom user i communicate) between user i and user j, which is given by Eq. 5, and $T(i,j)$ is the normalized call duration or talk time (normalized to the maximum talk time among all users with whom user i communicate) between user i and user j, which is given by Eq. 6.

$$F(i,j) = \frac{f(i,j)}{\max\limits_{k \in U_i}\{f(i,k)\}}, \quad (5)$$

$$T(i,j) = \frac{t(i,j)}{\max\limits_{k \in U_i}\{t(i,k)\}}, \quad (6)$$

where $f(i,j)$ is the total number of calls or call frequency between user i and user j, $t(i,j)$ is the total call duration or talk time between user i and user j, and $U_i = \{1, 2, ..., N\}$ is the set of all users associated with user i (*i.e.,* all users who have made/received calls to/from user i with total of N users).

Therefore, $S(i,j)$ has values in the range $[0, \sqrt{2}]$, which indicates the mobile social closeness between user i and user j from user i's perspective where 0 implies the closest and $\sqrt{2}$ implies the farthest relation. Based on this quantity, we can categorize all users associated with user i into three social groups using a simple grouping algorithm as follows.

Let R denote the Euclidean distance from coordinate (μ_F, μ_T) to (1,1) where μ_F and μ_T are the means of $F(i,j)$ and $T(i,j)$, respectively and $j \in U_i$. If $S(i,j) \leq R/2$, then user j belongs to Group 1, if $R \geq S(i,j) > R/2$, then user j belongs to Group 2, and if $S(i,j) > R$, then user j belongs to Group 3.

To validate the accuracy of our social closeness/grouping computation, we use the second set of our data described in Sect. III.A. During our second dataset collecting process, we interviewed the subjects about the social closeness for all of his/her associated users by having the subjects identified for each associated user (caller/callee ID) the perceived social group. Each participant received $20 as compensation. As the result, our second dataset includes additional information of social group corresponding to each associated user.

After comparing our calculation against the user feedback, we are able identify social groups correctly with the overall accuracy rate of 93.8%. The detailed result is shown in Table 1, which presents number of correct classification (Hit), number of incorrect

classification (Miss), and the accuracy rate (Hit/(Hit + Miss)) for each user. Based on the follow-up interviews with these ten subjects, most of "Miss" are caused by confusion between the face-to-face social closeness and mobile social closeness. For example, one of the subjects indentifies his roommate as a group 1 member since the subject sees and talks with his roommate on daily basis, the subject however does not make/receive many phone calls to/from him. As the result, his roommate is classified to group 2 based on our calculation (Eq. 4) but identified as group 1 member by the subject. To avoid biased feedbacks from the subjects, we did not provide any information about our social closeness computation or much more details about the three social groups than the description provided earlier in this section. Nevertheless, we believe that we have a decent result in accuracy rate and, in addition, we do not have any incorrect classification that misses more than one level of social group.

User	Hit	Miss	Accuracy Rate (%)
1	60	5	92.31
2	57	6	90.48
3	48	5	90.57
4	141	13	91.56
5	127	8	94.07
6	188	11	94.47
7	88	3	96.70
8	80	6	93.02
9	62	1	98.41
10	87	4	95.60
Overall	938	62	93.80
Mean	93.80	6.20	93.72
Std. Dev.	44.82	3.61	2.64

Table 1: The result of validation of social group calculation, which includes the number of correct/incorrect classification (Hit/Miss) based on our social closeness calculation and group classification, and the accuracy rate for each user.

To see the impact of the social closeness on the CPL, Fig. 12 shows the overall accuracy rate versus the length of the CPL for different social ties; group 1, 2, and 3. The CPL performs better in accuracy for the callers with closer social tie.

IV.E. Impact of Change of Life's Schedule

Since call logs represent human behavior associated with trends and changes of behavior over time, thus the accuracy of the CPL can also be impacted by

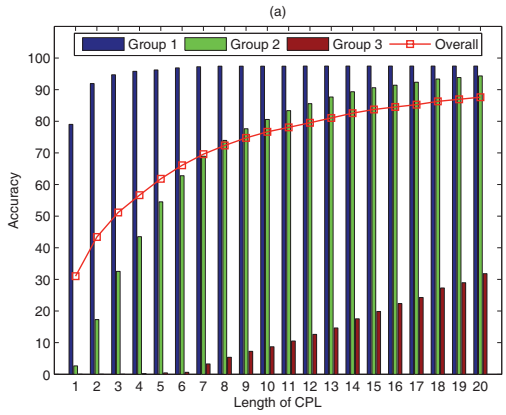

Figure 12: The overall accuracy of the CPL as an incoming call predictor for different lengths of the list as well as for different social groups.

the change of the caller's life schedule because it changes the calling pattern towards the user. For example, your friend changes job from working Monday through Thursday from 8AM to 5PM to working Friday through Sunday from 6PM to 3AM. This major change of your friend's life schedule may result in totally different calling pattern towards you, from receiving several calls at night and on weekends to several calls during the day and on weekdays, for instance. The change of calling pattern of several callers could degrade the performance of the CPL even more.

IV.F. How fast can CPL become reliable?

How fast can the CPL learn to become a reliable predictor for its user? This is an important question to answer. In attempt to answer this question, we monitor the accuracy of the CPL as the learning time or usage time (*i.e.,* number of days since that user starts using CPL) increases. We find that the accuracy normally starts with a low value, fluctuates, then gradually increases, and eventually becomes more stable at some level. The answer to the question of when the CPL will become a reliable predictor or when the accuracy will become stable, is not trivial. Of course, for CPL being reliable predictor does not necessarily mean that it has perfect accuracy (100%) but rather it has a stable accuracy (*i.e.,* small variation). The accuracy level when it becomes stable as well as the time that takes for the accuracy to become stable may depend on various factors such as number of incoming calls per day, structure of caller's calling pattern, and aforementioned factors that impact the accuracy (*i.e.,* increase of caller population, new callers, change of caller's calling pattern).

Nonetheless, we demonstrate the relationship be-

tween the learning time and the accuracy of CPL by plotting accuracy as learning time (number of days) increases for three different sample users (randomly selected from our dataset) with different incoming call rates in Fig. 13, Fig. 14, and Fig. 15 where their number of incoming calls per day are 15.65, 5.61, and 2.05 respectively for different length of the predicted list ($L = 1, 5, 10$, and 20).

As we previously speculated that one of many possible factors that may determine how fast the accuracy to become stable was the rate of incoming calls or number of calls received per day. Since other factors such as structure of caller's calling pattern and change of caller's calling pattern are harder to identify and are difficult to quantify for comparison among users, therefore we can restrict our attention to just incoming call rate and assume for a moment that other factors are approximate the same for all users.

In fact, it is evident in Fig. 13, 14, and 15 that the accuracy of CPL becomes stable faster for higher incoming call rate. *It is reasonable because the more calling information (higher incoming rate) that CPL learns, the quicker CPL recognizes caller's calling pattern.*

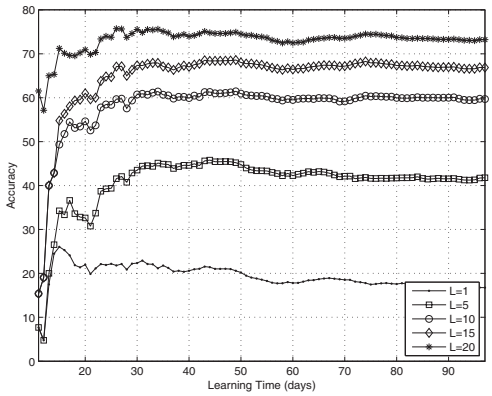

Figure 13: The accuracy of CPL as learning time increases for sample user who receives averagely 15.65 calls per day.

IV.G. Unpredictability of Calls

The accuracy rate of the CPL can be impacted by many different factors as mentioned previously. One of the factors that has a high impact on the accuracy of the CPL is the "randomness" of the calling pattern of each caller.

Randomness or uncertainty associated with a random variable has been studied and defined as the information entropy by Claude E. Shannon [7] as follows.

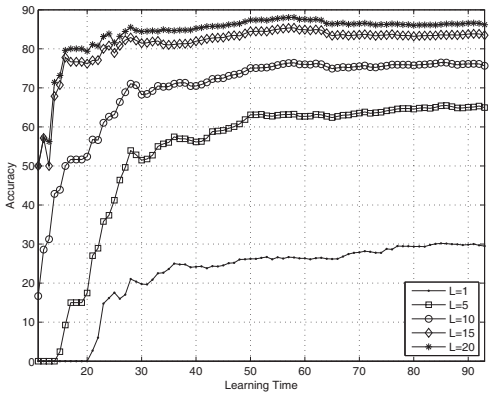

Figure 14: The accuracy of CPL as learning time increases for sample user who receives averagely 5.61 calls per day.

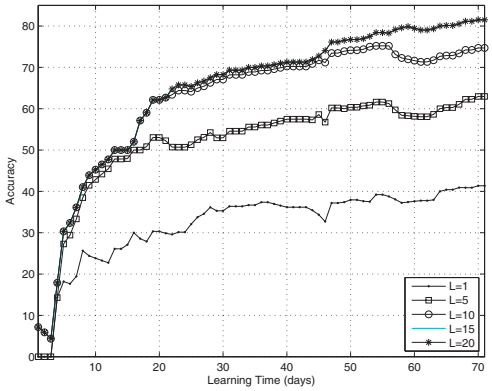

Figure 15: The accuracy of CPL as learning time increases for sample user who receives averagely 2.05 calls per day. Note that accuracy curve for $L = 15$ is equal to $L = 20$.

$$E(X) = -\sum_i p(x_i) \log_2 p(x_i), \qquad (7)$$

where $E(X)$ is an entropy of random variable X where $x_i \in X$ and $p(x_i) = Pr(X = x_i)$.

By adopting this information entropy, we define the *Unpredictability of Incoming calls* (UI) as the sum of the entropy of each caller such that UI increases with randomness of each caller as well as the number of possible callers. The unpredictability of incoming calls for the user k (UI_k) is given by Eq. 8.

$$UI_k = \sum_k^N \left(-\sum_h^{24} p_k(h) \log_2 p_k(h) \right), \qquad (8)$$

where N is the total number of callers and

$$p_k(h) = \frac{C(T_k H_h)}{\sum_{h=1}^{24} C(T_k H_h)}. \quad (9)$$

We compute the UI_k for each user in our dataset ($k = 1, 2, 3, \ldots, 30$). Fig. 16 shows that the accuracy rate of the CPL at $L = 5$ decreases unsurprisingly with the unpredictability.

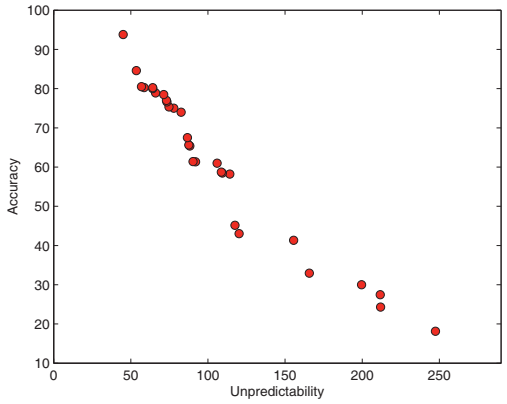

Figure 16: The overall accuracy rate of CPL as an incoming call predictor decreases with the unpredictability of incoming calling patterns.

IV.H. CPL as an Outgoing Call Predictor

With the same framework, the CPL can function as an outgoing call predictor. We find that the analyses that have been done so far for the incoming call predictor is also valid for the outgoing call predictor. Figure 17 shows the overall accuracy rate of the CPL as an outgoing call predictor with and without considering the "new callees". About 10% improvement in accuracy is also evident. Figure 18 shows the accuracy rate of CPL as an outgoing call predictor for different social groups. A similar result to the incoming call predicted list's is also obtained here where the CPL predicts much more accurately for the callees who are within a closer social tie. Figure 19 shows the accuracy of CPL as an outgoing call predictor decreases with the unpredictability of the user's outgoing calling pattern. As expected, the accuracy rate decreases with the unpredictability of the outgoing calling pattern.

V. Applications of CPL

To demonstrate the usefulness of CPL besides its own features, we describe here two applications of CPL including Call Firewall and Call Reminder.

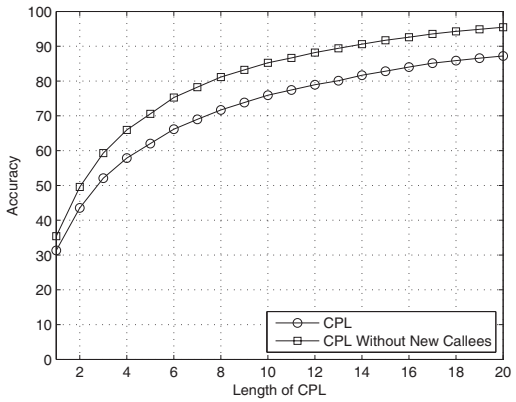

Figure 17: Overall performance of the CPL as an outgoing call predictor with and without considering the new callees.

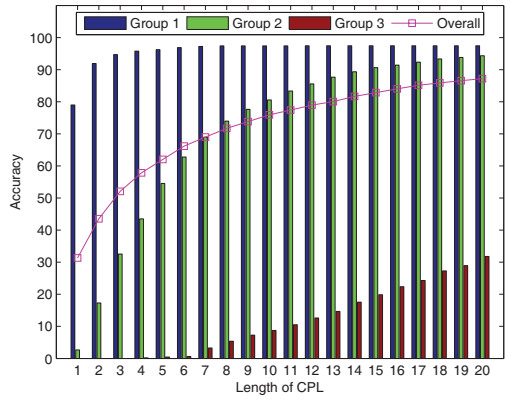

Figure 18: The overall accuracy of the CPL as an outgoing call predictor for different lengths of the list as well as for different social groups.

V.A. Call Firewall

By adopting the concept of firewall – the wall that keeps destructive forces away from our computer systems, Call Firewall basically monitors and handles incoming calls by keeping unsolicited and unwanted calls away while allowing desired calls to pass through. The problem of unwanted telemarketing calls or spam calls is expected to be a serious problem especially in VoIP networks due to its much lower communication cost than the circuit-switched telephone network system (it also becomes an attractive target for spammers). In fact, SPIT (Spam over Internet Telephony) is roughly three orders of magnitude cheaper to generate than traditional circuit-based telemarketing calls [8]. Unlike e-mail spam, call spam is a real-time problem, which requires a real-time defense mechanism. The real challenge is thus to block the spam call before the phone rings. Not only these

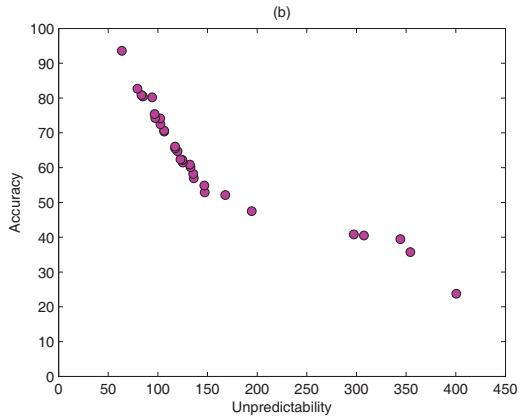

Figure 19: The overall accuracy rate of CPL as an outgoing call predictor decreases with the unpredictability of outgoing calling patterns.

spam calls create nuisance for the user, Kolan et al. [2] showed that each incoming phone call created different level of nuisance depending on the users presence (mood or state of mind) based on situational, spatial, and temporal contexts. Therefore, to address this problem of unwanted calls, the system for detecting voice spam and estimating spamminess level (known as VoIP Spam Detector or VSD) described by Kolan and Dantu [1] and the nuisance computation model (known as Nuisance Detector or ND) proposed by Kolan et al. [2] can be integrated with the call prediction model proposed in this paper (CPL) to proactively handle incoming calls before the phone rings. VSD, as described in [1] is a multi-stage adaptive spam filer based on presence (location, mood, time), trust, and reputation to spam in voice calls. It uses a close-loop feedback control between different stages to detect a spam call. As described in [2], ND is a model for computing nuisance level of incoming calls based on the social closeness and other behavioral patterns such as periodicity of the caller and reciprocity.

Call Firewall functions as follows. CPL generates a periodic 24-hour call prediction to be fed into VSD to learn behavior of callers (among which are spammers) and analyze the trustworthiness (VSD indicates the untrusted calls to be "dropped") and ND computes nuisance level associated with each predicted call (ND determines each call to be either sent directly to "voicemail" or "ringer" to ring the phone), then a set of firewall rules is generated e.g., IF John calls between 10am-11am, THEN forward it to voicemail, IF Pizza House calls between 4pm-5pm, THEN drop the call. The firewall rules are updated periodically (can be as often as every hour depending on the user). The user can also provide feedbacks about the actual nuisance level or reporting spam calls in order to improve the performance of the firewall.

V.B. Call Reminder

One of the common problems of everyday life is forgetting to make a phone call that could either be an event-based call such as birthday call, meeting planning call, etc. or a nonevent-based call such as calling parents on weekends, calling girlfriend/boyfriend during a lunch break, etc. Therefore, besides the proposed outgoing-call prediction functionality of the CPL that generates a list of potential callees to help speed up searching for a number to dial through a typical lengthy address/contact book, we describe here a Call Reminder that makes use of CPL's outgoing call predictor by integrating it with ND and event calendar to generate a "reminder" for the user to place a call to a particular person based on the user's past history, nuisance level, and events.

CPL periodically makes outgoing call prediction (e.g., hourly), which will be mapped onto the nuisance level computed by ND. The result is then evaluated by the decision-making mechanism to generate the call reminder e.g., high likelihood and low nuisance level would imply prompting a call reminder. The event calendar (a function that normally comes with today's mobile phone) is used to provide details about the call reminder e.g., birthday call, meeting plan, project discussion, etc. The user would be prompted with a reminding message such as "Would like to call John about the ACM conference?", "Would like to call Alice about the birthday?", "Would you like to call Mom regarding about dinner?". The user can enter new events into the event calendar for future reminders. A feedback sensor forwards the actual outgoing calls to CPL to be analyzed for prediction as well as provides the user's feedback to ND to calibrate nuisance computation.

VI. Related Work

There have been some works on predictive modeling for telephone call demands. In [9], the authors apply the queuing theory to characterize queuing primitives such as the arrival time process, the service-time distribution, and the distribution of customer impatience. In [10], the author develops two variations of Poisson process models for describing count data of call center arrivals which utilized the proposed mixed models technique. There is also a work describing a predictive model for the emergency 9-1-1 call volumes in [11], where the authors used a multiple lin-

ear regression model technique to construct the multi-dimensional linear predictor based on the call history. The work that is fairly close to our work is [12] where the authors develop a system for predicting a future communication activity based on the past communication event information. The system analyzes the past communication event information (including phone calls and emails) to determine whether a correlation existed in the past communication and predicted the future communication event based on the current communication event and the correlation. The correlation is computed based on the pattern of incoming and outgoing calls *e.g.,* if a call received from "person A" resulted in a later origination of a call to "person B," the correlation value between the "person A" and the "person B" is increased proportionately and the correlation values corresponding to other persons not dialed is decreased accordingly. The work that is closest to our work is [13] where the authors proposed a Call Predictor (CP), which computes receiving call probability and makes the next-24-hour incoming call prediction based on caller's behavior and reciprocity. The caller's behavior is measured by the caller's call arrival time and inter-arrival time. The reciprocity is measured by the number of outgoing calls per incoming call and the pattern of inter-arrival/departure time. The CP only makes prediction for a pre-specified caller of when the caller will be calling in the next 24-hour time frame. In contrast, our CPL predicts the next-hour callers by generating a list of the potential callers. With CPL, user needs not to request prediction for each caller but with one request (*i.e.,* press "predict" button), a list of potential callers will be generated. The main contrast between the CP and CPL is that CP predicts "when" the caller will make a call to the user but CPL predicts "who" will be the caller/callee.

The CP predicts the time slot that the given caller will be calling, based on the average number of calls per day (M). Then, M hour slots with the highest probability will be chosen for the prediction. Based on its original model, comparison to our CLP is not possible. However, with a small modification by using CP's probability computed for not only a given caller but for all potential callers as a measure for ranking among multiple potential callers, a performance comparison can then be made. Figure 20 shows the performance comparison between CP and CPL with this slightly change in the CP's original model. We can see that the proposed CPL performs relatively better than CP – especially with the shorter lengths of predicted list (which are more critical length size than longer ones). The difference is about 10-15% in accuracy. We believe that the lower accuracy of CP is caused by its probability computation that is the average of the four probability measures based on different parameters. By taking the average, the total probability is dominated by a much higher probability component while impacted very little by other smaller probability components. On the other hand, the proposed CPL is based on the likelihood function, which is the product of probability components where each component is equally contributed to the likelihood value.

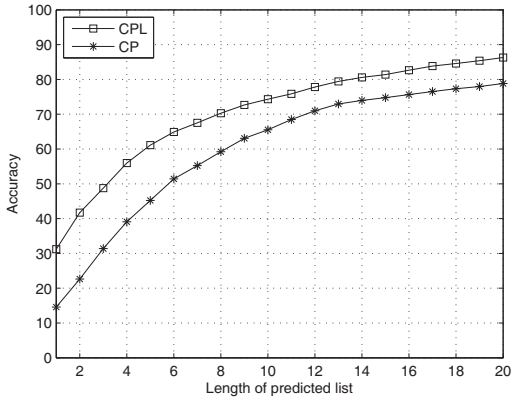

Figure 20: Performance comparison between CPL and CP[13].

VII. Conclusion

With the advancement of technologies embedded in today's mobile phones, people begin to engage the mobile phones more and more into many parts of their lives. Today's technology suggests that the mobile phone will eventually become a personal assistant that intelligently provides useful information to help its user making good decisions or even make decisions based on the user's context with the goal of to enhance quality of life. As a step towards this direction, we present here a model for predicting future callers and callees envisaged as a Call Predicted List (CPL). CPL makes use of the user's call history to build a probabilistic model of calling behavior based on the calling patterns and reciprocity. As an incoming call predictor, CPL is a list of numbers/contacts that are the most likely to be the callers within the next hour. As an outgoing call predictor, CPL is generated as a list of numbers/contacts that are the most likely to be dialed when the user attempts to make an outgoing call (by flipping open or unlocking the phone). This helps save time from having to search through a lengthy phone book. The CPL has been evaluated with the real-life

call logs from 30 mobile users and it shows a promising result in accuracy.

In this study, we have learned that the phone calls that seem random and unpredictable, it actually can be predicted accurately to some extent. We have also learned that there are however numerous factors that can impact the accuracy of the predictor such as the increase number of callers/callees, the new callers/callees, the mobile social closeness, the change of life schedule, the activeness of callers/callees, and the randomness of the calling pattern.

We are also aware of some limitations of this study such as the size of our datasets and the length of the call logs. We find it very difficult to collect these call logs from the subjects due to the privacy issues and the amount of time taken by each interview (about the social closeness) during our second dataset collection. Each interview lasted about one hour, which included downloading call logs from the phone service website and collecting feedback about the social closeness. There were only three months of call logs available for downloading from the service provider web page thus we are limited to three months of data for our analysis.

As our future direction, we will continue to investigate other parameters to improve our model as well as continue to collect more data for our future studies.

References

[1] P. Kolan and R. Dantu, "Socio-technical defense against voice spamming," *ACM Trans. Auton. Adapt. Syst.*, vol. 2, no. 1, p. 2, 2007.

[2] P. Kolan, R. Dantu, and J. W. Cangussu, "Nuisance level of a voice call," *TOMCCAP*, vol. 5, no. 1, 2008.

[3] S. Phithakkitnukoon and R. Dantu, "UNT mobile phone communication dataset," *http://nsl.unt.edu/santi/data_desc.pdf*, 2008.

[4] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. John Wiley & Sons Inc, 1973.

[5] M. Granovetter, "The strength of weak ties," *The American Journal of Sociology*, vol. 78, no. 6, pp. 1360–1380, 1973.

[6] P. V. Marsden and K. E. Campbell, "Measuring tie strength," *Social Forces*, vol. 63, no. 2, pp. 482–501, December 1984.

[7] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423 and 623–656, 1948.

[8] J. Rosenberg, C. Jennings, and J. Peterson, 2006, the Session Initiation Protocol (SIP) and spam. Spam Draft: draft-ietfsipping-spam-02.txt. [Online]. Available: http://tools.ietf.org/html/draft-ietf-sipping-spam-02

[9] L. Brown, N. Gans, A. Mandelbaum, A. Sakov, H. Shen, S. Zeltyn, and L. Zhao, ""Statistical analysis of a telephone call center: a queueing science perspective"," Wharton Financial Institutions Center, Tech. Rep. 03-12, November 2002.

[10] S. Aldor-Noiman, "Forecasting demand for a telephone call center: Analysis of desired versus attainable precision," *Master Thesis*, 2006.

[11] H. Jasso, T. Fountain, C. Baru, W. Hodgkiss, D. Reich, and K. Warner, "Prediction of 9-1-1 call volumes for emergency event detection," in *dg.o '07: Proceedings of the 8th annual international conference on Digital government research*. Digital Government Society of North America, 2007, pp. 148–154.

[12] C. E. Harless and T. J. Kowalski, "System and method for correlating incoming and outgoing telephone calls using predictive logic," *U.S. Patent 6084954*, 2000.

[13] S. Phithakkitnukoon and R. Dantu, "Predicting calls — new service for an intelligent phone," in *MMNS '07: Proceedings of the 10th IFIP/IEEE International Conference on Management of Multimedia and Mobile Networks and Services*. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 26–37.