Proactive Call Drop Avoidance in UMTS Networks

Shan Zhou* Jie Yang† Dahai Xu‡ Guangzhi Li‡ Yu Jin‡ Zihui Ge‡ Mario B Kosseifi§ Robert Doverspike‡
Yingying Chen¶ Lei Ying*

*Arizona State University, Tempe, AZ 85287
†Oakland University, Rochester, MI 48309
‡AT&T Labs - Research, Florham Park, NJ 07932
§AT&T Network Services, Alpharetta, GA 30004
¶Stevens Institute of Technology, Hoboken, NJ 07030

Abstract—**The rapid advancement of smartphones has instigated tremendous data applications for cell phones. Supporting simultaneous voice and data services in a cellular network is not only desirable but also becoming indispensable. However, if the voice and data are serviced through the same antenna (like the 3G UMTS network), a voice call with data sessions requires better radio connection than a voice-only call. In this paper, we systematically study the coordination between the voice and data transmissions in UMTS networks. From analyzing a large carrier's UMTS network recording data, we first identify the most relevant network measurements/features indicating a potential call drop, then propose a drop-call predictor based on AdaBoost. Moreover, we develop an intelligent call management strategy to voluntarily block data sessions when the voice is predicted to be dropped. Our analysis utilizing real service provider's data sets shows that our proposed scheme can not only predict drop calls with a very high accuracy but also achieve the highest user satisfaction compared to the other existing call management strategies.**

I. Introduction

A. Motivation

The wide usage of cell phones has increasing impacts in our social and business lives. Numerous data applications have emerged with the rapid advancement of new-generation smartphones. Cisco suggests that the average monthly data usage of the smartphone nearly tripled in 2011, and will increase 17-fold by 2016 [1]. Thus, supporting simultaneous voice and data services in a cellular network becomes not only desirable, but also indispensable.

On the other hand, most 3G smartphones are equipped with only one radio/antenna to transmit voice and data through cellular network [1]. In the current 3G network using CDMA2000/EVDO [2] where voice and data have to be serviced separately, [2] i.e., the data sessions will be blocked whenever a voice call comes in. In contrast, 3G network with UMTS (Universal Mobile Telecommunication System) [3] provides the appealing feature of supporting simultaneous voice and data by combining their packets together and sending over the single radio connection. For the new deployment of the 4G network such as LTE (Long Term Evolution) Advanced [4], provisioning voice over IP (VOIP) is not as mature as using the 3G techniques. Therefore, once a voice phone call is initiated, many carriers will switch the data sessions on LTE network to the mixed calls (voice + data) in 3G network by utilizing Circuit Switched FallBack (CSFB) [5]. Therefore, supporting reliable simultaneous voice and data services in UMTS networks will remain important in the future. However, it is known that the mixed calls require better radio connection than voice-only calls in UMTS networks [6], which demands the coordination between voice and data transmissions in case of weak signals. It is critical to reduce dropped calls and improve the service reliability when supporting voice and data simultaneously.

To minimize dropped calls, one promising direction is to voluntarily block data sessions when the voice call could be dropped due to weak signals. Thus, the simplest call management strategy is to block data whenever a voice call is made, however, defeating the purpose of serving voice and data simultaneously. Another straightforward strategy [7] involves blocking data only when the receiving signal strength of a cell phone is less than a pre-specified threshold. However, there is a no systematic approach of predicting potential call drops and intelligently performing call management by taking into account of users' satisfaction when supporting voice and data simultaneously. In this paper, through investigating the major reasons of the dropped call in transmitting both voice and data based on a large carrier's UMTS network recording data, we aim at devising a smart call management strategy grounded on machine learning approaches that can predict drop calls by identifying most relevant features and improve the reliability of serving voice and data simultaneously via intelligent call management based on user satisfaction.

B. Contributions

There has been little work on dropped call research related to the coordination of voice and data in 3G cellular networks due to real service data sets are usually not available to the public. To the best of our knowledge, this is the first paper systematically studying the coordination between voice and data transmissions in UMTS networks and providing valuable insights on supporting voice and data services simultaneously. We summarize our contributions as follows:

1) **Dropped Call Reasoning:** In this work, we first analyze a large carrier's data sets that record such low-level measurements and events. From the recorded information from the mixed calls with voice and data, we identify the features most relevant to the dropped calls from multiple measurements/features extracted from each call using machine learning techniques. An interesting discovery is that SIR_{err} rather than the signal strength is one of the most relevant features, where SIR_{err} is the gap of

[1]They may have a separate WiFi antenna. Some 4G phone have separate radios for 3G and 4G.

[2]Some CDMA2000 carriers, like Verizon, are to provide simultaneous voice and data services for the smartphones equipped with dual antenna.

978-1-4673-5946-7/13/$31.00 ©2013 IEEE

the measured SIR from the target SIR for the required transmission speed.

2) **AdaBoost-based Drop Call Predictor:** We propose a drop-call predictor based on AdaBoost [8], which shows high prediction accuracy under either the full set of available features or the most relevant features. The predictor has low demand on computation and memory, and thus is flexible to be deployed at either phone side or network side.
3) **User-satisfaction-based Call Management:** We propose a simple yet effective scoring system to quantify the user satisfaction. This proposed metric enables a fair comparison among various call management strategies which voluntarily block data transmissions. we find that the strategy of blocking data whenever voice comes performs the worst in terms of user satisfaction. Our AdaBoost-based drop call predictor can achieve the highest user satisfaction.

II. Background

A. RAB and mRAB

In addition to voice services, UMTS networks also support various data services, like web browsing, video streaming, etc.. Different services require different Quality of Service (QoS) and the required QoS is fulfilled by certain Bearer. In particular, the Radio Access Bearer (RAB) service is characterized by various attributes such as traffic class, maximum bit rate, guaranteed bit rate, delay, etc. [9]. UMTS networks also allow users to use multiple simultaneous services requiring different QoS by establishing multiple RABs (mRAB).

B. Data Sets

In this paper, we mainly use two data sets. The first data set contains Ericsson's GPEH (General Performance Event Handling) events collected at one Radio Resource Controller (RNC) in western U.S. on June 27, 2011. The GPEH data set comprises both RNC-internal events and Inter-node events. These events are triggered by signaling messages between RNC and user equipment (UE) and between RNC and base station (NodeB), respectively. Some important events include RAB establishment/release, soft handover evaluation/execution, radio quality measurement, etc. The GPEH events provide a rich source of features, such as signal strength, the type of RAB being used when the call is dropped, etc., to help assess network performance and to diagnose various problems like dropped calls.

The second data set covers all voice call records collected from MSCs (Mobile Switching Center) which are served by that particular RNC during the same time period. Each voice call record contains detailed information regarding each phone call, such as the starting time, ending time, originating numbers, terminating numbers and call terminating reasons, which we use as the ground truth to separate dropped calls and normally ended calls (or normal calls in short). We match these two data sets to identify all GPEH events generated during each phone call.

We note that no customer private information is used in our analysis. We anonymize all customer identities by hashing phone numbers and International Mobile Subscriber Identities (IMSI) prior to joining two data sets in order to protect customers' privacy. Further, at places we present normalized views of our results while retaining the scientifically relevant bits.

III. Reasoning and Predicting dropped calls

In this section, we employ an advanced feature selection technique to identify from a rich set of features provided by GPEH data set the most important ones that best predict dropped calls. On top of the selected features, we propose a machine learning based method for real-time prediction of potential dropped calls before they occur. Especially, when a mixed call is predicted to be dropped, we can take proactive actions based on the user satisfaction management framework, such as blocking the data session (see Section IV), to protect its voice session thus to improve customer experience. Note that all the calls discussed in this section are *mixed (voice and data) calls* unless otherwise specified.

A. Feature Extraction

From the GPEH data set, we enumerate a number of features that characterize the status of voice calls. Intuitively, the most recent measurements of an ongoing call are more important for drop call prediction than using a long history. Suppose the current time is t and we want to predict whether a call will be dropped in the next second, i.e., $t \rightarrow t+1$, we utilize the events during the latest Δt seconds, i.e., from $t - \Delta t$ to t, which we refer to as the *observing window*. The observing window should have a moderate size so that it can both provide sufficient historical information for an accurate prediction and avoid the out-of-dated information to enable predicting short-live calls. According to our experiments, a fixed time window of 20 seconds ($\Delta t = 20$) yields the best result. If a call is predicted to be dropped, the data connection could be disabled and the radio bearer to be reconfigured. Since the radio bearer reconfiguration procedure takes 250ms to 500ms [10], and the computational complexity of the prediction algorithm is low, 1 second is enough for us the take action. We next describe the features used in our study.

From all the events recorded in the GPEH data set within the observing window, we extract the following features which are more likely to be related to dropped calls and derive additional features on top of existing features using expert knowledge:

- Time till the latest RAB reconfiguration.
- Number of soft handover report events, evaluation events and execution events
- Number of Radio Resource Control (RRC) measurement reports.
- Mean, variance, and minimum value of RSCP, EcNo, SIR_{err} and RSSI.
- Derivative with respect to time of RSCP and EcNo.
- Type of the latest RAB.
- Last RAB reconfiguration event

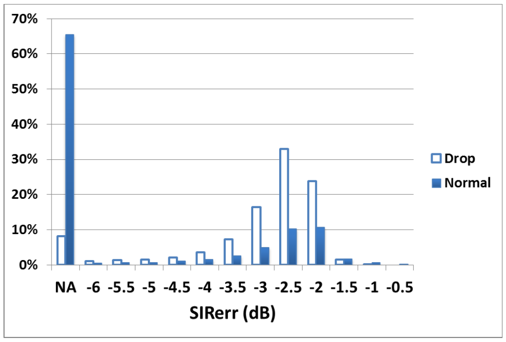

Fig. 1. Distribution of mean of SIR_{err} of dropped calls and normal calls

With the collected features, drop call prediction can be formulated as a classification problem, i.e., classifying on-going calls into potentially dropped calls (positive instances) and normal calls (negative instances). we adopt advanced Fast Correlation-Based Filter (FCBF) to identify the features that are most relevant to dropped calls (Section III-B). We then build a classifier to automatically predict drop calls in real-time using the machine learning algorithm *Adaboost* (Section III-C). At the end of this section, we discuss implementation details of the proposed method.

B. Feature Ranking

All features are ranked by applying the Fast Correlation-Based Filter (FCBF) feature selection algorithm [11]. Comparing to the other feature ranking and feature selection algorithms, FCBF is among the most scalable ones. This is important given the large data set we have. Based on the FCBF results, we can remove redundant features to reduce the complexity of our drop call prediction algorithm.

After applying FCBF on our data set, we find the most relevant features are 1) Mean of SIR_{err}, 2) Time since the last RAB reconfiguration 3) Last RAB reconfiguration event. We call these features the *selected features* in the rest part of the paper. We further investigate the significance of the mean of SIR_{err}. Its distribution is shown in Fig. 1. For example, in 32% of the cases, the SIR_{err} is around -2.5db when a call drops. There could be no SIR_{err} measurement (i.e. "NA" or Not Available) within the observing window. From Fig. 1, 65% normal calls do not have SIR_{err} measurements while only 8% dropped calls do not have such measurements.

To understand this phenomenon, we need to know under what condition the measurement of SIR_{err} is triggered. In fact, SIR_{err} is measured at Terrestrial Radio Access Network (UTRAN), specifically at the base stations (NodeBs), representing the quality of uplink. The SIR_{err} is maintained to be 0. However, when the SIR of some user cannot reach its target value due to power control problems, a SIR_{err} measurement will be triggered [12]. A typical scenario is that when a user moves fast, the signal received by the based station fluctuates continuously. In this case, an SIR_{err} measurement is usually triggered to inform that the quality of the physical channel cannot be maintained due to inner loop power control problem, and a call is also likely to be dropped under such a circumstance.

TABLE I
PREDICTION RESULTS WITH FULL AND SELECTED FEATURES

Algorithm	Full Features			Selected Features		
	TPR	FPR	AUC	TPR	FPR	AUC
AdaBoost	88%	19%	0.91	88%	22%	0.88
Max Entropy	87%	20%	0.89	89%	35%	0.77

C. Predicting dropped calls

We select the machine learning algorithm – AdaBoost – to train the classifier. Adaboost [8] is one of the most widely used machine learning algorithms in many areas. The basic idea of Adaboost is to combine multiple weak classifiers into a much stronger classifier, where each weak classifier could have substantial error rate and is only required to provide better than random guessing. We adopt decision stumps, or one-level decision trees, as the weak classifiers.

We choose AdaBoost because it has low computational complexity, and thus is more scalable than other state-of-the-art machine learning algorithms for both training and testing. In addition, the accuracy of Adaboost classifier outperforms many popular machine learning algorithms such as Naive Bayes and is comparable to more computational intensive classifiers like boosting decision trees and random forest, etc. [13].

To evaluate the performance of Adaboost algorithm on predicting dropped calls, we use standard 10-fold cross-validation[3] [14]. The prediction performance is measured using true positive rate (TPR) (i.e., the portion of positive samples that are classified correctly), false positive rate (FPR) and the Area Under the receiver operating characteristic Curve (AUC). The AUC is widely used for comparing classification algorithms, and a larger AUC indicates better accuracy. In addition, we also compare Adaboost algorithm with the maximum entropy algorithm [15].

D. Prediction Results

The prediction result using the full set of features is shown in the second column of Table I. The TPR of AdaBoost is as high as 88%.

The classification result using only the selected features is shown in the third column of Table I. For AdaBoost, the TPR remains the same while the FPR increases from 19% to 22%. The overall accuracy indicator, AUC, decreases from 0.91 to 0.88, which suggests that we can still obtain very good accuracy using only the selected features. In addition, the AUC of maximum entropy algorithm decreases much more significantly (from 0.89 to 0.77) with the selected features, which further suggests using AdaBoost for the real deployment.

E. Implementation Constraints

Our machine-learning based prediction is lightweight algorithm that can be deployed both at UE and RNC sides. If the

[3]The whole data set is randomly divided into 10 subsets. The whole procedure consists of 10 rounds. In each round, one subset is selected as the testing set, and the union of the other subsets is the training set. In the end, the results of the 10 rounds are averaged to get the final result. The cross-validation method can avoid the over-fitting problem when the number of positive samples (like dropped calls) are relatively small.

algorithm is deployed on the UE side, the user will have the freedom to enable or disable this function. But the software development will be challenging for the hundreds of different types of UEs. On the other hand, if the algorithm is deployed at RNC side, the deployment is much easier and all the users in the network can benefit from the algorithm at the cost of additional computation overhead at RNC.

IV. Quantitatively Strategy Evaluation with User Satisfaction

In this section, we seek to define a user-satisfaction-based metric that can quantitatively reflect the effectiveness of a call management strategy. We observe that our AdaBoost-based prediction scheme can achieve the best user satisfaction.

A. New Metric Based on User Satisfaction

Consider a mixed call with a voice call and simultaneous data transmission. After applying a specific call management strategy, the call could end up with one of the following four situations: (1) both voice and data kept (E_1), (2) only voice kept (E_2), (3) only data kept (E_3), and (4) neither voice or data kept (E_4). We can assign a score for each case to represent user satisfaction upon the situation. By summing up the scores of all the mixed calls within a cellular network, we can derive the overall user satisfaction as the metric to be optimized.

How to define a meaningful score for each situation? In this paper, we employ a simple credit/penalty scoring system. For simplicity, we grant one point for a kept voice call, and $m \leq 1$ points for a kept data session where m is the relative importance of data over voice. Moreover, we deduct k points as the penalty for a dropped voice call, and deduct km points for a dropped data session. Therefore, the scores (w_i) for the four situations are:

- $E_1 : w_1 = 1 + m$
- $E_2 : w_2 = 1 - km$
- $E_3 : w_3 = -k + m$
- $E_4 : w_4 = -k - km$

Denote $p_i, i = 1, 2, 3, 4$ as the probability of a mixed call ended in situation E_i within a cellular network. The overall user satisfaction metric can be defined as

$$S = \sum_{i=1}^{4} p_i w_i. \tag{1}$$

With the above metric, we can compare various strategies of voluntarily blocking data transmission. These strategies fall into the following three categories:

- L_0: Block data whenever a voice call comes
- L_1: Block data when necessary
- L_2: Always enable data usage during a voice call

Our prediction scheme based on AdaBoost (presented in Sec. III) falls into category L_1, and is named as $L_1(AdaBoost)$. In the following subsections, we introduce another threshold-based strategy called $L_1(Threshold)$ and study the selection of parameters (m, k), and computing p_i, with the proposed scoring system. The user satisfaction metric of these two strategies under L_1 category together with strategies from categories L_0 and L_2 will be compared.

B. $L_1(Threshold)$ Strategy

Within the L_1 category, one straightforward scheme is called $L_1(Threshold)$ where the data is blocked when the cell phone receiving signal (RSSI) is smaller than a predefined threshold x. The scheme is motivated by the observation that the dropped call rate increases when the RSSI decreases.

C. Selection of m and k

In practice, there are two kinds of data session: user initiated and background running. Usually, user is not aware of background data session when making phone calls. Whether the background data session is kept or not has less impact on the overall user satisfaction. We thus mainly focus on the data session initiated by user when making phone calls. Previous study [7] showed that the user-initiated data constitutes 15% of the data usage during voice calls. Thus, when we see a mixed call with data usage, the probability that the data usage is initiated by the user is 15%. Assume that the user-initiated data is of the same importance as a voice call. Then the importance of data usage can be set to $m = 0.15$.

We also employ a penalty parameter k in the user satisfaction metric to quantify the negative effect of dropped calls. A too small k indicates the dropped call has little impact on the user satisfaction, while a too large k causes one dropped call dominating the user satisfaction regardless of the contributions from numerous normally-ended calls. Empirically, we set the range of k between 1 and 4 to reflect the appropriate penalty of a dropped call, where the value 4 indicates the largest penalty point such that one dropped call will not dominate the user satisfaction.

D. Strategy-based Derivation of Probability p_i

To obtain the user satisfaction of the network defined in (1), we need to know p_i, the probability of a mixed call ended in situation E_i. The probability of each situation under different call management strategies is derived as follows.

L_0 **Strategy:** Under this call management strategy, data usage is blocked whenever a voice call is made. Hence, the probability of situations with both data and voice kept situation (i.e., p_1) and data kept (i.e., p_3) is equal to 0. Knowing the probability of the dropped call p_4, which can be obtained directly from the GPEH data set, the probability of only voice kept p_2 can be calculated as $p_2 = 1 - p_4$.

L_2 **Strategy:** Under this strategy, a mixed call is always allowed. For the given mixed calls, the dropped call probability of each situation (i.e., $p_i, i = 1, 2, 3, 4,$) can be obtained directly from the GPEH data set. We note that the situation with only data kept E_3 is very rare. In fact, in all the dropped calls we have in GPEH data set, we did not observe such situation. Thus, the probability of only data kept p_3 can be set as approximately zero.

L_1 **Strategy:** The L_1 strategy is the most intelligent one among the three call management categories. The probability

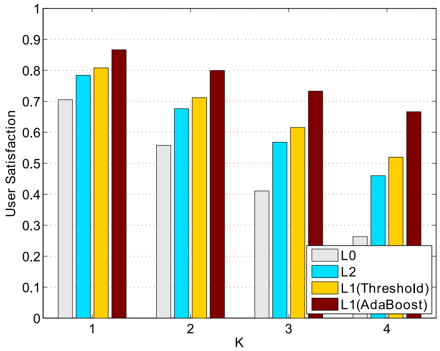

Fig. 2. User satisfaction for different strategies with relative data importance $m = 0.15$ and different call-drop penalties k.

under each situation is derived by combining the probability of each situation under L_0 and L_2 strategies, and concurrently considering the prediction results using L_1 (either AdaBoost or threshold-based methods):

- p_1: Under L_1 strategy, in order to maintain both data and voice sessions, a normally-ended mixed call should be predicted not to drop. Hence, p_1 under L_1 is the probability of the both voice and data kept under strategy L_2 multiplied with the probability of a correct prediction of a normal call.
- p_2: The probability of the situation with only voice kept under L_1 strategy includes two parts. First, situation with voice only kept happens if a mixed call will not drop, however, it is predicted to be dropped, and the data is blocked by L_1 strategy. Second, a mixed call is correctly predicted as a dropped call, and the voice call is saved by blocking the data.
- p_4: Similarly, the probability of neither voice or data kept includes two parts as well. The first part is the dropped mixed calls couldn't be predicted by prediction algorithms. Hence, these calls will be dropped eventually because L_1 will not block the data connection due to the mis-prediction. The second part is the dropped mixed calls that are correctly predicted, however, the call is still dropped after blocking the data connection.
- p_3: Finally, since the probability of only data kept p_3 under strategy L_1 is smaller than that of under strategy L_2, we set p_3 under strategy L_1 as zero due to p_3 under strategy L_2 is approximated to zero.

E. User Satisfaction

We evaluate the overall user satisfaction under different strategies based on Formula (1). Fig. 2 presents the user satisfaction under different strategies when varying the penalty parameter k from 1 to 4. We found that the overall user satisfaction achieves the highest value when the drop-call penalty point k is set to the smallest value 1 for all strategies, and user satisfaction decreases when the penalty point k increases. More importantly, we observed that the user satisfaction of our proposed $L_1(AdaBoost)$ is consistently the best among all the strategies when varying the penalty points k value. The user satisfaction of L_0 is always the worst across different values of penalty due to it blocks data connections whenever there is a voice call. Moreover, the strategies with drop call prediction (i.e., L_1) can consistently achieve better user satisfaction than that without drop call prediction (i.e., L_2).

V. Conclusion

In this paper, we take a first step to systematically study the coordination between voice and data transmissions in UMTS networks. Based on the study from a large carrier's data sets, we identify the features most relevant to the dropped calls from multiple measurements/features extracted from each call using machine learning techniques. We propose a drop-call predictor based on AdaBoost, and a new metric based on user satisfaction to perform intelligent call management for supporting voice and data simultaneously. We show that our AdaBoost-based drop call predictor can achieve the highest user satisfaction under numerous scenarios.

References

[1] "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2011-2016," Feb. 2012.

[2] L. Korowajczuk, *Designing CDMA2000 systems*. Wiley, 2004.

[3] C. Cox, *Essentials of UMTS*. Cambridge University Press, 2008.

[4] H. Holma and A. Toskala, *LTE for UMTS: Evolution to LTE-Advanced*. John Wiley & Sons, 2011.

[5] I. Tanaka, T. Koshimizu, and K. Nishida, "CS fallback function for combined LTE and 3G circuit switched services," *NTT DOCOMO Technical Journal*, vol. 11, no. 3, pp. 13–19, 2009.

[6] R. Kreher and T. Ruedebusch, *UMTS Signaling: UMTS Interfaces, Protocols, Message Flows and Procedures Analyzed and Explained*. New York, NY, USA: John Wiley & Sons, Inc., 2007.

[7] L. S. Myneni and M. B. Kosseifi, "Effects of background data on call quality in simultaneous voice and data calls in a UMTS network," AT&T, Tech. Rep., 2011.

[8] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, 1997.

[9] 3GPP, "Typical examples of Radio Access Bearers (RABs) and Radio Bearers (RBs) supported by Universal Terrestrial Radio Access (UTRA)," 3rd Generation Partnership Project (3GPP), TR 25.993, Sep. 2008.

[10] C. Johnson, H. Holma, and I. Sharp, "Connection setup delay for packet switched services," in *IET International Conference on 3G and Beyond*, 2005, pp. 1–5.

[11] L. Yu and H. Liu, "Feature selection for high-dimensional data: A fast correlation-based filter solution," in *Proceedings of International Conference on Machine Learning*, vol. 20, no. 2, 2003, pp. 856–863.

[12] R. Kreher, *UMTS performance measurement*. Wiley Online Library, 2006.

[13] Y. Jin, N. Duffield, J. Erman, P. Haffner, S. Sen, and Z.-L. Zhang, "A modular machine learning system for flow-level traffic classification in large networks," *ACM Trans. Knowl. Discov. Data*, vol. 6, no. 1, pp. 4:1–4:34, Mar. 2012.

[14] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *International Joint Conference on Artificial Intelligence*, vol. 14. Lawrence Erlbaum Associates Ltd., 1995, pp. 1137–1145.

[15] S. J. Phillips, M. Dudík, and R. E. Schapire, "A maximum entropy approach to species distribution modeling," in *Proceedings of International Conference on Machine learning (ICML)*, 2004, pp. 83–90.