Predicting Delay-Bound Violations for Cellular Transmissions:
Pre-Hospital ECGs Uploaded from Moving Vehicles

A Thesis

Presented to
the faculty of the School of Engineering and Applied Science
University of Virginia

---

in partial fulfillment
of the requirements for the degree

Master of Science

by

Alexander Gessner

May

2012

APPROVAL SHEET

The thesis

is submitted in partial fulfillment of the requirements

for the degree of

Master of Science

AUTHOR

The thesis has been read and approved by the examining committee:

Advisor

Accepted for the School of Engineering and Applied Science:

Dean, School of Engineering and Applied Science

May

2012

Acknowledgements

First, I would like to thank my advisor, Professor Stephen Patek, and the rest of my committee, Professors John Lach and Barry Horowitz, at the UVA School of Engineering and Applied Science for all their help and guidance. Jianping Wang was instrumental in solving many of the early technical challenges. I'd especially like to thank Drs. Mark Sochor and David Burt, physicians at the UVA Medical Center Emergency Department who provided constant guidance and subject-matter expertise and direction. I would also like to thank all of the care providers who allowed me to shadow their work and answered all of my questions: EMTs Mary Fowler and Larkin Hamrick as well as Dr. Juliana Minak. Special thanks as well to the UVA medical student Kristen Gruenther and UVA Systems Engineering undergraduate students Andrew Gould, Erin Myers, Marisa Mutty, Michael Durej, Imran Khan, Bethany Martin, and Peter Bradley for all of their help and parallel efforts. This work was supported by the WICAT Center at UVA and the Lockheed Martin Corporation.

Abstract

Time-critical data uploads from mobile hosts are challenging when there is high variability in connectivity to the base stations. In these situations, it can be difficult to know in advance if a transmission will meet a delay-bound guarantee. One example of this type of transmission is photographs of 12 lead ECGs. The images are captured by EMTs using the printouts from machines in ambulances, and they are sent to physicians for pre-hospital diagnosis of heart attacks. Current systems do not adapt their operation to the quality of service of the local network, and as a result, they do not provide guarantees or feedback to users about the success or failure of the transmission. A data-centric, application-specific approach and validation method are presented to predict the likelihood of failure in real-time for one of these transmissions. The prediction informs both ends of a voice conversation - between EMT and physician - allowing them to adapt to the knowledge of whether or not the image will be available. Our approach is implemented as an algorithm in an iPhone application that manages the capture and transmission of these diagnostic photographs. Field experiments validate the efficacy of the predictor; it is able to distinguish successful transmissions from failed ones 96% of the time. The framework and end-to-end transmission system are designed to allow for generalizability and ease of extensibility to other networks, cities, and future network improvements.

Table of Contents

List of figures

CHAPTER 1

1 Introduction

1.1 Motivation

The end-to-end transmission system discussed in this thesis was designed to aid pre-hospital care specifically in the area of diagnosis for potential patients of ST-elevation myocardial infarctions (STEMI). The system is composed of an iPhone application (“app”) that captures and wirelessly transmits images of 12-lead electrocardiograms (ECGs) and a webserver that receives the images and displays them to doctors in the Emergency Department of the hospital. Using two-way radios on emergency bands, emergency medical technicians (EMTs) in the field have the ability to communicate with hospital physicians with near 100% coverage in the Charlottesville area. Existing systems to transmit a diagnostic ECG from the field are substantially less reliable in areas like this without complete cellular coverage because transmissions are instant, and they can only work at best if there is connectivity at that moment.

Using a validated prediction algorithm, the “app” is able to predict the probability of successfully transmitting an ECG image within two minutes from a moving vehicle en route back to the hospital. Success in this context means that within the two minute time limit, the entire image has been uploaded from the smartphone and received at the server. Additionally, acknowledgements have been received at the phone indicating the success. If none or part of the image has been received in the time limit, this is considered a failure. In the context of predictions, a successful prediction is a true positive or true negative; a failed prediction is a false positive or false negative. Transmission success, however, is different from model success.

The calculated probability in the model is updated in real-time and easily visible to the EMT on the ambulance with a patient. Its primary function is to serve as an input to the conversation between the EMT and the hospital physician. If the ECG is likely to arrive, the physician can know to expect and use this information for diagnosis. If the image is unlikely to arrive, the physician need not stand at the computer screen waiting idly by when a diagnosis immediately will be just as accurate as one made several minutes later.

The 12-lead ECG is a several seconds snapshot of the electrical potential across 12 points around the heart. Each of these 12 waves is displayed in a standard configuration in a three by four rectangle. Since the configuration is standardized, a photograph of the output from any 12-lead ECG machine is recognizable to and readable by any trained physician. Within the “app,” the EMT is led through the process of capturing a photograph of the printout from the ECG machine on the ambulance. The “app” then compresses and crops the image to reduce the size of the transmission down from over a Megabyte to about a 30 Kilobyte jpeg file. This reduction in the file size drastically increases the probability of successful transmission in areas of mediocre cellular data coverage.

A first round of data collection was undertaken to gage the bandwidth and variability of the AT&T cellular data network around the city of Charlottesville, Virginia. The key insight from this data was that the successful upload of a 30 Kilobyte file from a moving vehicle may be determined more by the amount of connectivity rather than the throughput. This is because any location with any data network connectivity at all had sufficient throughput for this application.

Another, more realistic, round of data collection was performed. For this effort, 83 compressed photographs were attempted to be uploaded within 50 miles of the University of Virginia Medical Center. Each transmission was performed from a moving vehicle bound for the hospital. In addition to these transmissions, the data collection also included the recording of simple, binary connectivity measures, hereafter referred to as pings, every 4 seconds. A successful ping occurs when 56 bytes can be uploaded and 16 bytes downloaded within about 2 seconds. For the scope of this document, successful and unsuccessful pings will be referred to as “ONs” and “OFFs” respectively. The 83 transmission attempts and 6833 pings include timestamp and GPS information. A logistic regression model is presented to model the probability of success of a transmission attempt using the pings and other data as predictor variables. The best model includes only one predictor variable: the percentage of pings within one kilometer of the current location that have been recorded as ONs. For the most accurate prediction, success is determined by greater than 65% of the nearby pings being ON.

This model stands up to a number of evaluation techniques including ordinary bootstrapping, random cross-validation, and two newly presented forms of geographic cross-validation to promote generalizability. The model utility test passes with a p-value less than 10^{-14}, and the optimism-adjusted area under the ROC curve using the method outlined in [53] is 0.96.

A parallel effort is currently being undertaken by a group of undergraduate Systems Engineering students to design the application and integrate the algorithm discussed herein for robust testing by real users.

1.2 Contribution

The primary contribution of this work is in the field of application-specific predictions of mobile telemetry characteristics. Bandwidth estimation techniques cannot fully and quickly describe the network conditions of a fast moving vehicle through a cellular data network. It is, however, feasible to narrow the focus and only estimate those bandwidth characteristics that are actually necessary for the context of the specific application. This research presents a real opportunity for fast, limited bandwidth estimation in a real system with the potential to improve patient care. The method presented is a predictive model for a transmission built around simple measures of nearby network connectivity. Use of this type of data in this way for this kind of prediction is novel.

Secondary contributions can also be found in the fields of pre-hospital STEMI care and geospatial modeling evaluation. There is novelty in the minimization of file size of 12-lead ECG photographs while still maintaining readability. Also, the methods of geographic cross-validation leverage the unique way the system will ultimately be used and thus how the data was collected to promote geographic generalizability of the method and the system beyond AT&T in Charlottesville.

There is also a contribution to the field of wireless health. Increasingly, smartphone platforms are being used as aggregators of bio-sensor data for wireless health applications [58] [60], and are even beginning to be used for closed-loop wireless health applications [59] [57]. With 1GHz processors, modern smartphones are capable of intelligently managing the transmissions between sensors and distant care providers. In this context, it is not hard to imagine situations in which it would be beneficial for an aggregator to know in real-time whether or not an upload transmission would succeed within a given amount of time, e.g. in remote monitoring applications where an immediate response may result from near real-time health data.

1.3 Outline

Chapter 2 of this thesis will provide background of all the relevant and intersecting fields of research. It will begin by describing the field of wireless quality of service and bandwidth estimation: all of the technical fields in which this research can provide value. Then the background will present the health care usecase for which the research was performed and evaluated. Finally, chapter 2 will present the current modeling techniques that will be used and a synthesis to help direct the rest of this thesis. Chapter 3 will present the general research problem and the application-specific challenge. Chapter 4 is dedicated to discussing the SPRITE algorithm: what it is, how it was developed, and how it will be implemented. Chapter 5 is the evaluation of the predictive model by standard and custom techniques. Chapter 6 will conclude the thesis with reflections and thoughts for future work.

CHAPTER 2

2 Background

2.1 Bandwidth estimation and mobile data quality of service

Predicting the likelihood of successfully transmitting a file of known size (~28 Kilobytes) within a known time limit (2 minutes) from a known location (captured by the GPS on the smartphone) while driving to another known location (the hospital) is a kind of geographic real-time bandwidth estimation problem. There are two possible ways to go about solving this problem: (1) estimating the bandwidth (connectivity, reliability, throughput, …) and then calculating the likelihood of successfully transmitting given those expected conditions or (2) skipping the estimation of bandwidth and modeling success or failure directly from realistic data points that include factors known to impact bandwidth. Approach (1) requires examining the field of bandwidth estimation, which approaches this problem from three different directions: signal strength maps, Probe Gap Models (PGM), and Probe Rate Models (PRM). Each of these three methods, while powerful, is fundamentally inappropriate for our application. Approach (2) requires leveraging the application-specific features into the field of predictive modeling.

2.1.1 Geospatial mapping

Wireless broadband providers like AT&T typically publish maps of their claimed network coverage [1] [2] [3]. These maps are based on geospatially interpolating signal-to-noise-ratio (SNR) measurements taken in the field and applied to topological maps that include obstructions like buildings and bridges [4]. These methods require a large amount of context information like the size of newly constructed buildings as well as the locations and precise tuning parameters of the network towers. This information is either difficult to obtain and update (e.g. buildings and new construction) or would require significant corporate agreements to begin to collect (e.g. tower locations and tuning parameters) and might still be rendered outdated by routine network improvements. These techniques and models are generally built with full knowledge of the network to solve problems like tower location and network management optimizations [5] [6]. Additionally, in the newest versions of iOS, it is not possible to determine the SNR of the phone programmatically. A bandwidth estimation technique built upon corporate agreements for information-sharing could not be maintained passively and may discourage expansion and generalizability of the work to other cities and networks.

2.1.2 Probe gap models

Bandwidth estimation can be performed accurately without full knowledge of the network by using a method called a Probe Gap Model (PGM). This method is based on the idea that transmitting a large number of small files of known size and at known time-intervals from a client to a server can then be examined on the server. The differences in the time intervals are a strong indicator of available bandwidth in the end-to-end channel. While new methods like Traceband have been able to speed up the measurements of the most seminal methods like Spruce, these techniques may still be too slow for this research [7] [8]. For a single estimate, Traceband uploads as few as 87 Kilobytes and as much as 146 Kilobytes from the end device. This amount of data dwarfs the actual size of the compressed ECG transmission! More importantly though, the available bandwidth calculation is made on the server and would then have to be communicated to the end device. Our application needs to be capable of predicting success and acting on it from regions of no connectivity. A PGM approach would not be able to communicate from the server to the smartphone if the smartphone were in a region of zero bandwidth.

2.1.3 Probe rate models

The alternative approach is performed by inducing traffic on the network from the end-user device by transmitting packets of known file size at deliberate times and receiving the acknowledgements (or lack thereof) from the server. The network's performance (packet drop rate and round trip times) on these transmissions is then used to calculate network characteristics like throughput, reliability, and lag at the end-user device. This method is based on the Probe Rate Model (PRM). These techniques like Pathload, pathChirp, and imTCP take on the order of several minutes to converge, so they tend to be applied to very high speed, wired networks [9] [10] [11] [12] [13] [14]. The amount of time necessary to obtain a measurement is simply unfeasible for a moving vehicle: 40 mph * 1 minute = 1 kilometer. If a measurement were to converge within 1 minute, that value would represent some kind of combination of readings gathered over the span of a kilometer of distance. It is unclear whether this value would have any real meaning at all if the network conditions changed through the distance covered and how to associate that measurement with a location on a map. In general, PRM methods are considered unfeasible for wireless cellular networks [15].

2.1.4 Factors shown to affect wireless bandwidth

There are a number of factors that have been shown to affect wireless transmissions: speed [16], weather [17], time of day and day of the week [18], SNR, and tower locations. Periodic measures of simple connectivity have also been shown to inform future predictions of available bandwidth [19]. As mentioned above, SNR and tower location are not feasible to obtain for use in this system. There is, however, a proxy. AT&T publishes a map (Figure 1) of their claimed service: white represents no claimed coverage at all, light blue is EDGE, and dark blue is 3G service.

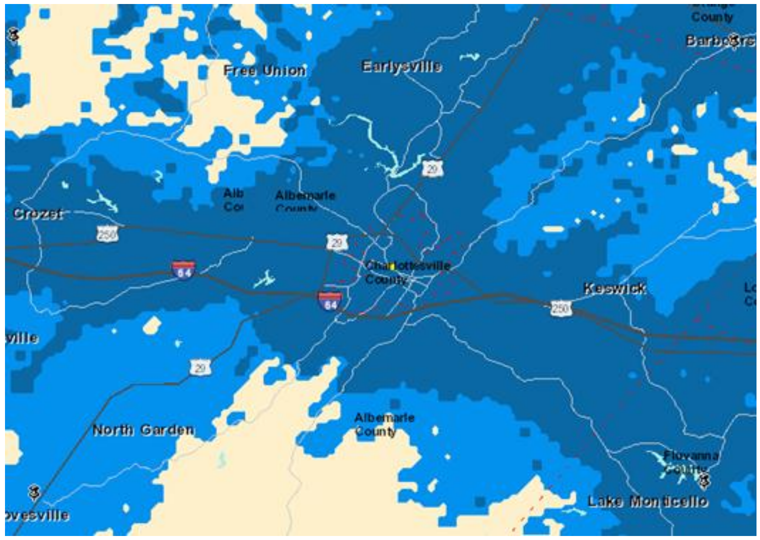


Figure 1: Map of AT&T reported signal strength around Charlottesville

This map is based on a geospatial interpolation of signal strength using tower locations and topography. We do not intend to evaluate its accuracy, but since it is based on non-real-time variables to which we do not have access, it represents a potential source of information for a data-centric approach to a bandwidth estimation problem.

2.1.5 Cellular access networks

Wireless measurements of quality of service (QoS) include all of the typical wired measurements along with additional factors for the variability of the channel [20]. Many of these service measures are obfuscated for this system by the choice to use HTTP over TCP instead of dealing with the actual signal strength and managing the transmission bit-by-bit in real time. In systems with this kind of obfuscation, the QoS measures at the application level are more relevant. More simply, these are: access to a connection, delay, throughput, and packet loss rate.

Application-level use of QoS metrics are often used for real-time adaptation of downloads like variable bit rate multimedia streaming or clients requesting different versions of the same services (e.g. webpages) based on exceptionally low or high QoS metrics. These metrics can also be used at lower levels in the smartphone operating system to perform optimizations for tasks like network selection [21]. Measuring throughput, as mentioned above, is difficult, slow, and inaccurate. Measuring connectivity, however, is much faster and can be performed at any level of the network stack. Connectivity measurements at the application level are often used to enable or disable features. There is also a large body of research using connectivity to perform real-time or predictive user localization and related features like future network access [18] [19] [22] [23].

2.1.6 Motion through a wireless network

If an end-user device in an access network (client) is moving at high speeds, the QoS metrics can be highly variable. Motion through a wireless network has been researched from many different perspectives: low level wireless sensors, high speed vehicular motion, node localization, …. Some of these will not be discussed because they are far out of scope. Also, because of the platform-level constraints (iPhone 4 on AT&T) and transmission methodology (HTTP over TCP), many of the SNR-based results in the wireless sensor metrics are not useful here because the iOS platform does not allow programmatic readings of SNR.

High speed motion has been shown experimentally to affect a whole range of cellular access network QoS metrics like throughput and packet loss rate because of Doppler effects and tower hand offs [16]. In Jang's research, the motion was completely within the bounds of network coverage. Faster speed yielded worse metrics. It is conceivable, however, that some motion could actually improve certain metrics, especially if the application-specific needs are somewhat delay tolerant. For example, a client device behind a tree might have no connectivity because of the obstruction. Any and all tests performed from that location would imply terrible QoS. Motion through that point, however, might not be noticeable in a real measurement of throughput. If the application specific demands are for a relatively small amount of data over a relatively long period of time (when compared to average throughput and delay of the network), QoS may wind up depending more on some notion of the amount of connectivity over the path traveled rather than precise measurements of signal strength at a single location.

2.1.7 Vehicular communication

Available bandwidth estimation from a moving vehicle is a difficult problem primarily because of the motion of the client [24]. Ott and Kutscher, running experiments with uploads from a moving vehicle passing through a wifi access point, made a number of relevant realizations. First, throughput is highly variable and non-linear, even under the controlled conditions of a wifi access point. Second, geography and speed play significant roles. All of the tests saw a sharp drop in throughput because of a dip in the road. When driving 60 mph as opposed to 25 mph, the throughput rate was cut in half from about 340 KBps to about 170 KBps, and this rate cut is independent of the reduced time spent in the bounds of the network making the transmission. Last, for an application with substantial delay tolerance and minimal data needs, "short periods of connectivity can achieve substantial information exchange that are likely to suffice for many transaction style applications" [24].

Because of the difficulty involved in "highly dynamic network topologies" like high mobility in a wireless network, Lai recommends leveraging unique application specific features for more useful information [25]. In his case, it was the highly repetitive and regularly periodic patterns of public transportation systems. In another example of application specific features being used to solve the problem of high speed motion through a wireless network, Zhao et al. proposes a vehicle-to-vehicle relay system to ensure that each vehicle receives a single transmission from a fixed node. They found that the quickly changing signal strength caused highly variable throughput that resulted in some vehicles receiving the transmission while other vehicles did not. With multiple vehicles moving together, relays from vehicle to vehicle dramatically increases the information diffusion [26]. Similar vehicle-to-vehicle techniques for data dissemination are in the literature [27]. A related example uses vehicle trajectory to optimize a download [28]. This work analytically solves an optimization for packet and vehicle rendezvous at wifi intersections along a suspected path. In our research, the application specific features to leverage are the fixed, small sizes of the files relative to the network throughput and the typically improving bandwidth that is seen while the vehicle is en route and the transmission is being attempted.

There is also a large amount of less relevant but related research in the field of wireless vehicular communication systems, mostly over 802.11 wifi. IEEE has defined a set of standards for different paradigms of vehicular communication (V2V: vehicle to vehicle, V2I: vehicle to infrastructure, VANET: vehicular ad hoc networks, …) under 802.11p [29]. Much of this work is focused on two primary goals: individual vehicle localization and multi-vehicle traffic patterns. Pathirana uses mobile base stations to perform vehicle localization and trajectory prediction [30]. Relatively unique to this work, however, is that the end-users as well as the base-stations are all mobile. This makes for a highly complex and variable system. With active monitoring data from many vehicles and clever passive monitoring techniques, real-time traffic conditions can be estimated [31].

Additionally, research has been aimed at contributing to smarter vehicles that can drive with significantly less human interaction, though this research is more about sensors and situation awareness than transmissions and connectivity [32]. In a similar forward-looking approach, research has also been directed at network management of the vehicle communication systems that are expected in the near future [33]. This research approaches the interesting challenges of multiple end-users with different mobility characteristics and throughput needs, and how best a management system can balance fairness.

Cottingham et al. were able to build coverage maps specifically for vehicular wireless communication, but only with very large amounts of SNR data. This research represents a significant step in understanding vehicular wireless transmissions, but their focus was on very high speed networks in an urban location [18].

While relevant to describe the space of the research in this thesis, most of the related work in wireless communication from vehicular systems is too low-level to apply to this work, too localized to use a broadband cellular access network, too forward looking to consider real world constraints in a region with only modest cellular data coverage, or requiring too much data to be adaptable as per the requirements of the system in this thesis.

2.1.8 Connectivity autocorrelation

Connectivity of wireless links demonstrates temporal autocorrelation which is dependent on the quality of the link and the amount of time between measurements. The point is relevant to this research because pings are captured and reused later for prediction. The available research is inconclusive to determine how long these pings remain useful in a model.

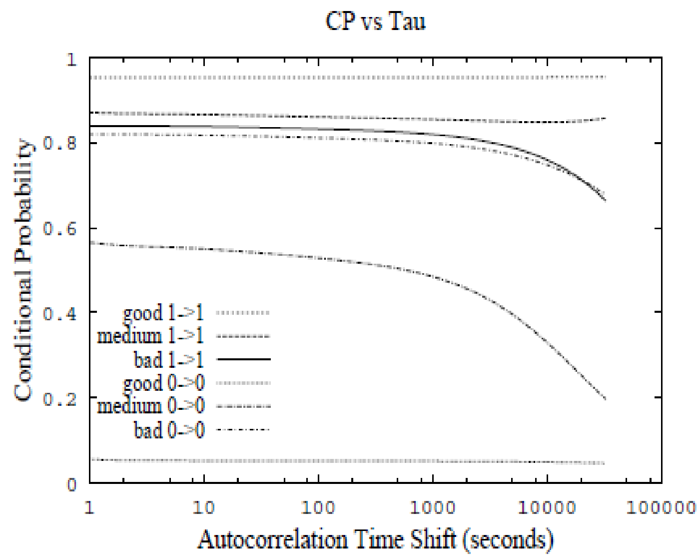


Figure 2: Conditional probability of connectivity as a function of time shift [34]

In Figure 2, good, medium and bad describe the overall signal strength between the pair of wireless nodes, 1->1 is an ON ping following another ON ping [34]. Horizontal lines would indicate no change in the autocorrelation of connectivity with respect to time elapsed. What is visible, however, is that very strong and very weak networks show stable autocorrelation for ON and OFF pings respectively. Otherwise, the autocorrelation tends to fall off. This indicates that as time progresses, the effects of randomness in this study's network overwhelmed the predictive power of previous pings. Also, ON and OFF pings exhibit different temporal autocorrelation properties from each other and may need to be treated differently by SPRITE in future work.

In research for an application called BreadCrumbs, ping data was successfully used for evaluating a predictive model "several weeks" after it was first collected [19].

2.2 Pre-hospital STEMI care

Each year in America, about 250,000 people have a STEMI (ST-elevation myocardial infarction), the deadliest kind of heart attack, and nearly one-third of them do not receive treatment [35]. STEMIs are particularly dangerous because every 30 minutes following the onset of symptoms results in a 7.5% increased chance of morbidity within a year [36].

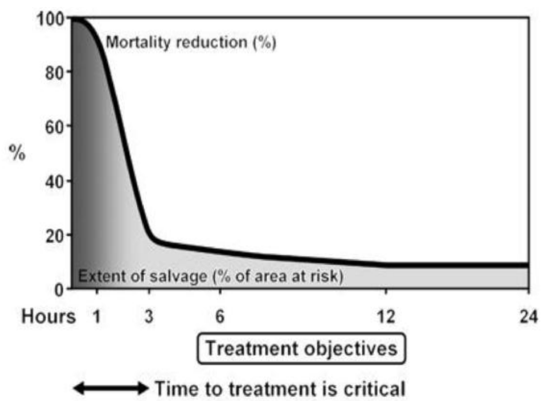


Figure 33: Mortality reduction from a STEMI as a function of time to treatment [55]

See Figure 33 for the plotted mortality reduction as a function of time till treatment.

2.2.1 STEMI diagnosis and treatment

Diagnosis of a STEMI is typically made using a 12-lead electrocardiogram (ECG). The ECG is a several seconds snapshot of the electrical potential of the heart muscle measured on the skin at 12 points around the heart. Each of these 12 leads is displayed in a standard configuration: a three by four rectangle. Each beat of each wave is composed of several components: the P wave, the QRS complex, the T wave, the ST segment, the PR interval, and the QT interval. STEMIs are often recognized by slight changes in the ST segment of certain combinations of leads. Some STEMIs present with extremely noticeable patterns in the ECG. Others can be so difficult to recognize that they require the trained eye of a cardiologist.

Treatment at the UVA Medical Center requires a small, specialized lab called the catheterization lab (“cath lab” for short) to be opened and staffed. The hospital does not have sufficient resources to keep it running all the time. When a patient with a STEMI is en route or present at the hospital’s Emergency Department, a physician can activate an alert. This decision initiates a chain of events that impact over a dozen care providers. If this happens when the lab is not open, a cardiologist and two nurses who are off-duty but on-call are summoned to the hospital, usually from their homes. This process can take up to 45 minutes [37]. If a STEMI patient is en route within 45 minutes of the hospital and the “cath lab” is not open, then faster E.D. decision making directly yields earlier treatment which, as mentioned above, directly improves patient outcomes [36]. Additionally, about a third of hospitals in America are rural [56]. Opportunities for time savings are even greater in these locations because of the distances involved.

2.2.2 Existing transmission systems

Aside from speed of recognition, Emergency Department physicians also have an interest in making this initial decision as accurate as possible. False negatives can result in patient death, treatment delay, or both. False positives, on the other hand, can result in provider fatigue and financial costs to the hospital. For these reasons, diagnosis has begun to move forward to pre-hospital care. Ambulances can now be equipped with 12-lead ECG capturing machines and devices to transmit the diagnostic ECG wirelessly to the hospital.

In this way, a physician can read the ECG and decide to activate the alert before ever laying eyes on the patient. Giovas et al. has independently validated that the transmission of ECGs from moving vehicles is feasible and improves patient care [38].

Because of other activity in the Emergency Department and the risks associated with systemic indecision, physicians are willing to wait for a transmission for two minutes. This means that the best possible scenario is a successful transmission within two minutes. If success is not possible within this timeframe, patient outcomes are better served by the physician making a decision without the ECG in hand [37].

There are two types of systems that transmit 12-lead ECGs wirelessly from ambulances to hospitals: direct and indirect. Direct systems include the machine which actually captures the data. These are sold by large-scale medical device suppliers like Medtronic, Philips, and Zoll. Because they have access to the actual waveform points, a relatively small amount of data, they can transmit it over the commercially-available, wireless voice networks. These systems tend to have initial setup costs in the tens of thousands of dollars as well as monthly licensing and maintenance fees to be borne by hospitals as well as the Emergency Medical Service (EMS) rescue squads that run the ambulances. Because hospitals buy many other pieces of equipment from these companies, the relationships and the sales force are already in place at an institutional level. EMS squads typically have much smaller budgets making the costs potentially more of an issue for them. This type of solution dominates the market [37], however many rescue squads and hospitals around the United States do not have a system in place because the costs are prohibitive. Additionally, many of the drawbacks with the direct systems stem from their nature as “black-box,” inflexible tools with minimal user-interactivity. These systems provide no feedback to the user regarding the likelihood of the transmission being successful or even whether or not it actually was received. If it failed, the EMT must be told by the Emergency Department, at which point another attempt can be made.

Indirect systems use an existing 12-lead capture device and handle the transmission as a separate function. One example of an indirect system is called FastECG [39]. It is a smartphone application designed to take a picture of the printout from any 12-lead capture device and then wirelessly transmit it as an image to a centralized server. Then a physician with an internet connection is able to look-up the image through the FastECG website. Indirect systems are much less expensive and can capture ECGs from any brand of 12-lead machine. This capture, however, dramatically increases the amount of data because a series of waves must be converted to a photograph. The additional data and smartphone-platform-limitations necessitate that the transmissions go over commercial data networks instead of voice networks. Data networks have higher throughput but often much less coverage. This can mean potentially lower availability than systems like the direct ones which use voice networks.

A number of similar research-based systems have also been developed. As part of the Code Blue research effort at the University of Harvard, a wireless sensor infrastructure was prototyped for Emergency Medicine workflows. The system, called miTag, is designed to capture and transmit a range of patient vitals over reliable wireless mesh networks. The technology is focused on rapid triage for large-scale disasters capable of overwhelming all local hospitals. A test was performed with emergency responders reacting to a simulated car wreck at a busy intersection in Washington, DC [40]. A similar, operational and more expansive effort in Tucson, Arizona was recently shut down because of a funding deficit [41].

For a project called TeleTrauma at the University of Massachusetts, a system was prototyped to transmit ECG and images from an ambulance to the hospital. The researchers conducted a test where they transmitted data from a laptop while driving around Amherst in a car. The transmission updated compression levels on the image for recognized changing bandwidth but did not consider delay tolerance [42].

Some of the cities around the world with the best commercially available wireless coverage are in East Asia, so it is reasonable to expect there to be advanced research efforts attempting to leverage wireless networks in those cities. A system for ambulances was prototyped and field tested in Taiwan over a 3.5G mobile broadband network using sensors connected via Zigbee to a smartphone. Sensors measured ECG, blood pressure, pulse, body temperature, and created a real-time video feed. All of this data was collected for a small number of real patients in the city and transmitted over the network to the hospital through a smartphone. The researchers claimed that network connectivity and available bandwidth were never an issue. In fact, their planned future work is more sensors and more data over 4G networks. This study presents an interesting counterpoint to Charlottesville because of the sharp contrast in wireless bandwidth availability [43]. Similar systems also exist in Europe on high speed wireless links without the need to consider constraints like HIPAA [44].

The Cincinnati Children's Hospital Medical Center in concert with the GlobalMedia Corporation developed an operational system called TransportAV. This system consists of a camera attached to a gurney which facilitates doctor-to-patient interaction while patients are in transit. The system attempts to be real-time, so frame rates fall when network bandwidth drops [45]. This means that any video captured of patients in transit through low network coverage is going to be of extremely low quality. TransportAV values data immediacy over data resolution, whereas the system discussed in this research is operating in a near real-time regime.

2.2.3 Opportunities

ECG transmission systems have the potential to use context to perform much more intelligently. For example, the EMT could be notified whether or not the transmission was successful, and a failed transmission could be re-initiated automatically until the two minutes time limit. Furthering this point, the lower availability of data networks with respect to voice networks could be mitigated by transmitting from a moving vehicle supported by real-time knowledge of nearby bandwidth conditions. For example, when the patient's ECG is captured, there may not be any wireless coverage. If the ambulance is driving towards the hospital, however, coverage may return in a matter of seconds. If this is the case, an intelligent system could predict this and still attempt the transmission if it is likely to succeed within two minutes. This example and the previously described constraints associated with pre-hospital STEMI care imply an interesting point; on an ambulance driving towards the hospital with uncertain cellular coverage, an EMT and the

Emergency Department could benefit from knowing how likely a transmission is to succeed within two minutes. This research provides a novel, accurate algorithm to deliver this feature. FastECG, the nearest existing system, provides the desired post-transmission feedback, but it does not re-initiate after failure or provide any pre-transmission indications of the likelihood of success. Also, the transmitted photographs in FastECG are around 180 Kilobytes. For reasons discussed below, these images can be further compressed to make the transmissions more likely to succeed.

2.2.4 ECG as image

Without complex corporate agreements to obtain the proprietary waveform data directly from the ECG machines, the obvious way to digitize the ECG onto a smartphone is by simply taking a photograph of the printout from the machine. This solution, however, dramatically increases the amount of data necessary to represent the raw information, because waveform points require substantially fewer bits to represent than a JPEG image of those points. Images, however, can be compressed very efficiently. This raises an additional question: how fine does the resolution of the image have to be for the ECG to remain readable by a physician? A more compressed image would have a smaller file size and be easier and faster to transmit in a variable network.

2.2.5 File size minimization

Since the best possible outcome for this transmission system is successfully sending the ECG image, the file size of the photograph should be as small as possible to increase this likelihood. Several methods were considered, but the final decisions were significantly impacted by platform and application limitations.

The first step was cropping. ECG strips are longer and thinner than the standard resolution of a landscape photograph. Through experimentation, if the ECG is lined up with the top of the photograph, the bottom two-fifths of the image can be cropped away reliably without removing any parts of the ECG.

Physicians in the Emergency Department at UVA requested that the color remain in the image because they are accustomed to using the red gridded background. Also, for a transmission that is slow to arrive, they would rather get parts of the image in a readable resolution rather than the whole image progressively. For these two reasons, the compression methodology chosen was to compress the ECG down to the lowest readable value, then break it into a grid of smaller images sent separately and recombined on the web server as they arrive.

Several compression methods for the whole ECG photo were attempted. The iOS platform has convenient image compression in PNG and JPEG formats. PNG compression unpredictably changed the colors of the image and did not reduce the file size any better than JPEG. JPEG, which is generally recommended for photographs, was visibly much clearer. Also, no simple vector tracing APIs could be found to vectorize the ECG image or extract the waveform data points.

After attempting numerous JPEG compression methods and informally examining dozens of compressed ECG photographs with two medical professionals (a highly trained emergency medicine cardiologist and a relatively novice medical student), a compression methodology was chosen to minimize file size and maximize readability. This method is a two-step process within the iOS platform: (1) compression using the native iPhone function UIJPEGRepresentation() with the parameter 0.5, and (2) resolution reduction by a factor of five. The original captured photo is at a resolution of 2592 by 1936 pixels and is about 1.5 Megabytes in file size. After the cropping and compression discussed above, the resulting image is 519 by 233 pixels and averages 27.7 Kilobytes in file size. In Figure 3, the left is the original, uncompressed picture. The right image is compressed using the method described above.

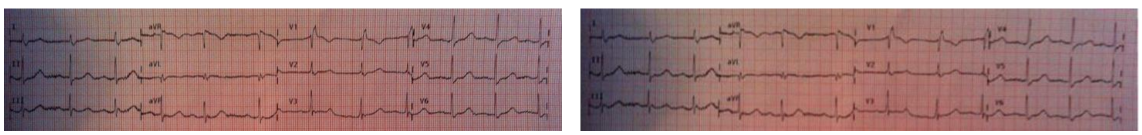


Figure 3: Example of photo compression on 12-lead ECG

The preliminary data collection effort verified that JPEGs of about 1.8 Kilobytes could be transmitted reliably around the Charlottesville area. For this reason, the 28 Kilobytes compressed ECG was divided into a grid of 16 smaller images of about 1.8 Kilobytes each. These files were the actual transmissions.

2.2.6 Survey validation

A formal, randomized survey was built to validate that this specific method of file size reduction does not affect the readability of an ECG. Twenty different ECGs were chosen and photographed. These were built into an online survey using SurveyGizmo where they appeared in a random order along with a short story like what might accompany a transmitted ECG from an ambulance [46]. In any single survey, whether or not each image was compressed or uncompressed was also a random event with probability 0.5. As of April 2012, this survey is still awaiting deployment by the physicians in the Emergency Department at the UVA Medical Center.

2.2.7 Near real-time mobile health telemetry

Near real-time (NRT) in the context of mobile health telemetry indicates a reference to the amount of delay expected and tolerable in the system. Real-time telemedicine tends to focus on capabilities like doctor-patient video streaming. Totally asynchronous telemedicine is something more akin to a patient e-mailing a doctor the day's diagnostic data. Both of these extremes in the time domain present an opportunity to be ignorant of highly variable network connectivity like the edge of a cellular network. In this kind of environment, video streaming is impossible and a time-insensitive e-mail can wait several hours or more. NRT transmissions operate in-between these two other paradigms. These transmissions are time-sensitive but tolerant of some delay depending on the application. For this research, the delay tolerance is less than two minutes, because a transmission that cannot complete within this amount of time is better off not attempted.

2.2.8 Preliminary data collection

Plan

A preliminary round of data collection was undertaken to get a basic understanding of the network conditions: throughput and reliability. In each of fifty-two locations within twenty five kilometers of the Emergency Department (Figure 4) a car was parked and a set of transmissions was attempted. Five groups of twenty JPEG images were uploaded from a custom-made iPhone application to our server. The groups of images were similarly compressed jpegs within group but of progressively larger sizes.

The average file size from each of the five groups is below:

Group Number	Average Size of Files
1	1.86 KB
2	2.51 KB
3	3.71 KB
4	9.06 KB
5	39.50 KB

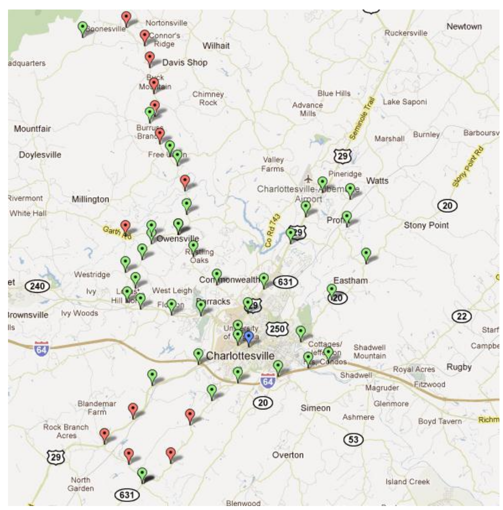


Figure 4: Preliminary data collection locations

Results and interpretation

A simple throughput calculation was performed and attributed to each location based on the size of the files received and the time elapsed. Many of the locations tested had no connectivity at all and therefore zero throughput. The others ranged from 30 Kilobytes/minute up to well over 1 Megabyte/minute. The data is plotted in Figure 5.

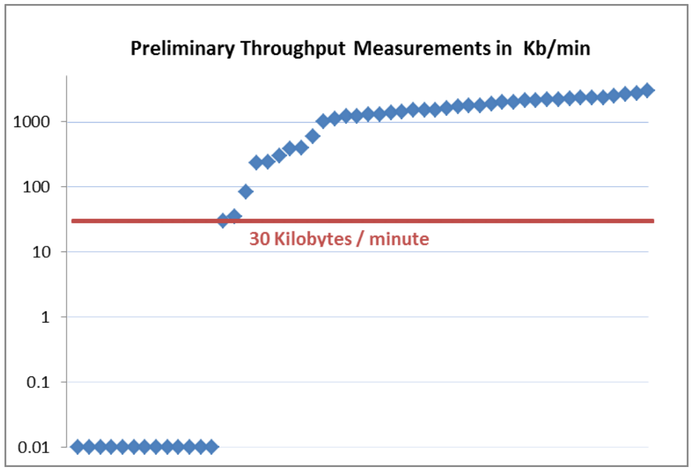


Figure 5: Throughput measurements from preliminary data collection

In the preliminary data collection effort, each data point in the above figure is the reduction of a stream of data uploaded from a single, precise location and time around Charlottesville down to a single measurement of throughput in Kilobytes per minute. While many of these values are 0, implying no connectivity, all of the rest were at 30 Kilobytes per minute or greater. The aim of this project is to upload a file that is typically less than 30 Kilobytes within only two minutes. The logical conclusion from these two statements is that, when not moving, connectivity is a perfect predictor of success and while driving in a moving vehicle, nearby measures of connectivity might be a predictor of success. This strongly indicates that real-time measures of bandwidth need not be exhaustive; a small amount of information about the network at a point (connectivity vs. no connectivity) may be enough for an application like this one.

2.3 Predictive modeling

The predictive application feature described in this research requires a data-supported model. There are three decisions to be made: the type of model, the number of data points required for significance, and how to evaluate it.

2.3.1 Modeling methods

There are countless modeling techniques in the field of machine learning. For this application, we are using predictor variables to model a binary categorical response: success or failure of a transmission as defined above. Since basic linear models cannot model a categorical response, logistic regression is the standard adaptation. This technique, instead of fitting to 0 or 1, models the probability of the outcome being a 1. Since the response variable is transformed to a continuous number between 0 and 1, a linear model can be fitted to a logistic function, which also takes the values from 0 to 1. Linear models tend to be among the simplest and most interpretable, so logistic regression is a natural choice. Logistic regression works well on relatively small data sets [47] and is computationally tractable for repeated, real-time calculations on a smartphone platform.

Decision trees and neural networks are also simple, fast, and interpretable. Additionally, because of the ways they split the data, it is possible for these methods to discover multiple regions of the data behaving differently. For example, since the speed of the vehicle can be expected to increase bandwidth in some ways and decrease it in others, a decision tree or a neural network might be able to split the data set in ways that identify the points for which speed helps and the ones for which it does not. Like logisitic regression, decision trees and neural networks are computationally tractable for a STEMI transmission system.

There are numerous other techniques (e.g. support vector machines, random forests, and generalized additive models), some of which have been implemented successfully on smartphone platforms [48] [49]. These methods all require a lot more computation for each measurement which can typically be augmented by cloud computing. If the calculations must all be performed on the client device like, for example, predicting network service from regions with no connectivity, then using a server for help with computations is not possible. Also, the more computational methods are typically much less interpretable than logistic regression models and decision trees.

2.3.2 Power analysis and EPV

Considerable research has been performed regarding the number of data points necessary to achieve statistical significance for a logistic regression model. Peduzzi discusses the concept of Events Per Variable (EPV) to categorize the number of data points used to train a model with respect to the number of relevant predictor variables and likelihood of the categorical events. EPV for a binary classifier is the number of less common outcomes (e.g. for a medical experiment of 1000 patients of which only 900 survive, the less common number of outcomes is 100) divided by the number of predictor variables:

EPV = # of outcomes / # of predictors

Using Monte Carlo simulation and sampling from a large, modeled data set, he demonstrates that 10 EPV is sufficient to avoid the common problems of models built on too little data [50]. Vittenghoff using a broader range of initial data sets demonstrates that 10 EPV is actually quite conservative, and significance can be achieved with as few as 5 EPV [47].

2.3.3 Evaluation methods

The evaluation of logistic regression models also represents an interesting body of research. Receiver Operating Characteristic curves have been used since first invented during World War Two for analyzing radar signals. The curve shows the rate of true positives vs. false positives for a binary classifier as the threshold probability is varied from 0 to 1. This graph shows an intuitive, graphic way to choose how to use a classifier like a logistic regression, especially when the risks associated with false negatives and false positives are not the same.

The area under an ROC curve is equivalent to the probability of correctly discriminating between a randomly chosen success and failure. This value, AUC, is a commonly used metric to compare the performance of different predictive models partially because as a metric, it is independent of the ratio of successes to failures in the data set [51]. AUC is also used by the company, Kaggle, to evaluate the submissions to its predictive model challenges [52].

AUC is primarily a measure of discrimination because it evaluates how well a model separates between the binary states. The complimentary set of metrics is based on calibration, a measure of how accurate the probabilities actually are. Depending on the application, occasionally calibration is more relevant. In this research, the end result of the model is a simple determination of “probably will send” vs. “probably will not send.” Accuracy in this regime points directly to discrimination rather than calibration which is why, for this research, our focus is on measures of discrimination rather than calibration. Similarly, we used one-third-sized test sets and standard cross-validation methods.

Bootstrapping is also a well-researched technique for model building and evaluation [53]. In this case, bootstrapping to obtain distributions for the model coefficients and AUC will demonstrate how stable the model is. If the distributions on the coefficients have very low standard deviations and 95% confidence intervals not including 0, then there is confidence that the value is stable and significant. AUC ranges from 0 to 1, with 1 representing perfect classification. For a bootstrapped distribution of AUC, the higher the values are, the better and more stable the model.

2.4 Synthesis

The pre-hospital STEMI care usecase is an opportunity to leverage unexplored opportunities in application-specific, limited mobile bandwidth estimation. Because of the specific demands associated with transmitting ECGs, exhaustive bandwidth characteristics are not needed. This presents an opening to use predictive modeling and modest data collection to create a classifier that may be of use to actual care providers.

CHAPTER 3

3 Problem statement

3.1 General problem

The desired output of this research is a simple classifier that provides a real-time prediction of the likelihood of success of a transmission given the size of the file, the allowable delay, and bandwidth characteristics from a moving vehicle based on periodic probing. The size of the file is about 28 Kilobytes. The allowable delay is two minutes. The inputs are any real-time measures of bandwidth or factors expected to contribute to bandwidth as well as any non-real-time indicators of cellular coverage like the published service maps. The vehicle is within a 45 minute drive of and bound for the UVA Medical Center Emergency Department. The prediction must be performed and updated in real-time from a lean, smartphone platform that does not necessarily have a data connection at all times.

The model will be evaluated using standard methods in the field of predictive modeling: test sets, bootstrap confidence intervals, ROC curves, and AIC. Because of the generalizability constraints, the model will also be subjected to less conventional methods of geographic verification.

3.2 Specific instance

Pre-hospital STEMI care is an interesting usecase for near real-time mobile health telemetry. Unlike many other usecases, knowledge of the likelihood of success before the transmission is attempted could be very useful to care providers. The fundamental effort of this research is how to obtain this estimate given the real world constraints of an actual emergency medicine system. Project-level constraints have dictated that the work be done on an AT&T iPhone 4 servicing the Emergency Department at the University of Virginia Medical Center in Charlottesville, Virginia. The algorithm designed and validated in this research is for the STEMI transmission system, and it must be adaptable to real-world network updates as well as generalizable to other networks, hospitals, and cities. This research is an enabling component of a pre-hospital STEMI transmission system that is intended to provide the capability to EMS agencies around the country, particularly in more rural areas that may not have already purchased another system. These real-world system constraints have driven many of the choices made in developing the algorithm.

This predictive model is based on a snapshot of network conditions to maintain consistency, so all data were collected over the period of only a few weeks. Adaptability of the method going forward is dealt with separately. There were two types of data points collected: attempted transmissions and pings. Each of these data points includes a timestamp, GPS location and corresponding accuracy, and speed. The problem of this research is using real-world, accessible data to provide feedback to emergency responders that is fast enough to be useful and accurate enough not to be ignored.

The scope of this research and contribution are limited to an empirical framework for measurement based quality-of-service. The knowledge gained is substantial, and it will be invaluable for the greater project-level effort of deploying a prediction algorithm in an operational system.

CHAPTER 4

4 SPRITE: Smartphone Predictions in Real-time for Informed Transmissions of ECGs

4.1 Algorithm

For the purpose of this thesis, we will consider the predictive model to be a part of the transmission algorithm which includes all of the methodology behind managing the transmission and making decisions independent of the model's results by using context information to identify unexpected deviations. For example, if the application loads and every ping to the custom webserver is OFF, but connectivity exists to other websites, then regardless of the model's prediction, the application should warn the user that the server may be disconnected.

The best model created for the probability of a successful transmission in this application given all of the possible real-time and prior predictor variables was based on only a single variable: the percent of all pings collected within a 1 kilometer radius that were ON, and it is significant well below the 0.001 level:

$$P[success \mid PERCENT\_ON\_1KM] = \frac{\exp(-4.666 + 7.614 * PERCENT\_ON\_1KM)}{1 + \exp(-4.666 + 7.614 * PERCENT\_ON\_1KM)}$$

For the best accuracy, success is predicted when PERCENT_ON_1KM, the percentage of all of the pings within 1 kilometer of the current location that are ON, is greater than 0.648. This yields an interesting result: the success of a small upload from a mobile device in a near real-time state is largely driven by nearby measures of connectivity. An additional intuition is that an increase in 10% of the nearby pings being ON results in an approximate doubling of the probability of successful transmission.

4.2 Data collection

4.2.1 Methodology and organization

To build the most accurate model, the data collection process was designed to be as similar as possible to the real users' workflow. Each data point represents a location and time and speed from which a transmission of a JPEG averaging 27.7 Kilobytes was attempted to a central server from a moving vehicle bound for the hospital. If all 16 component images are received and acknowledged by the server within two minutes, this data point is a success; otherwise it's a failure.

Data points were carefully chosen to maximize the variability of the potential predictor variables discussed above as well as the results. We anticipated about three of the potentially dozens of predictor variables would be significant. If half of the data collected are successes, then 60 points and three variables yield an EPV of ten, which is a conservative goal considering the literature.

Since real-users will transmit ECGs en route to the hospital, this data collection was organized into routes as in Figure 6.

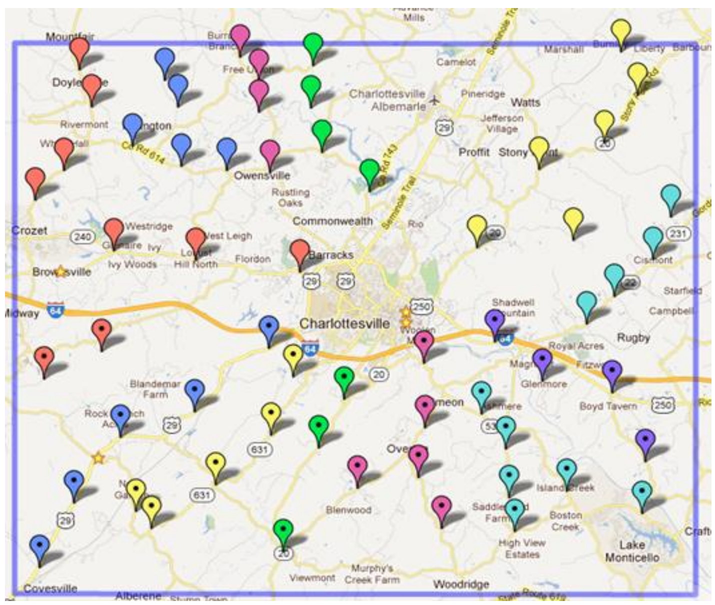


Figure 6: Data collection points organized into routes

4.2.2 Ping method

In addition to success and failure of the transmissions, there was a clear opportunity to collect some lesser measurements continuously before and between the transmission attempts. It was anticipated that these real-time measurements could be significant predictor variables. For predominantly two reasons, we chose to simplify these real-time measures of bandwidth to simple connectivity rather than any of the myriad choices from the bandwidth estimation literature. First, the preliminary data collection effort strongly implies that more information than simple connectivity is simply unnecessary. Second, this application will be deployed in a moving ambulance, and to maintain accuracy in the geospatial regime, measurements need to occur so fast that there is simply not sufficient time for anything more than a simple measure of connectivity.

The motion of the vehicle provides the most unique constraint of this work. GPS measurements on an iPhone 4 while driving around Charlottesville are consistently accurate to within 50 meters or less. If a measure of real-time connectivity is accurate in the space domain to within 50 meters, the time elapsed between the start and finish of that measurement should be less than time taken to drive out of the measurement's accuracy: 50 meters / 45 mph = 2.8 seconds. Any measure of real-time bandwidth for use as a predictor variable must complete within 2.8 seconds to maintain the existing geospatial uncertainty. Experimentation showed that this amount of time is barely enough to send and receive an acknowledgement for a single packet of negligible size. This time constraint completely eliminates the possibility of using any of the bandwidth estimation techniques based on transmitting multiple packets of different sizes because there simply is not sufficient time. The method that was chosen for real-time connectivity (ping) is NSURLConnection() in the Apple CocoaTouch platform. This function was used to upload a 56 byte file and download a 16 byte acknowledgement. There were a number of reasons for this choice. First, as a higher level networking function, it keeps the cellular radio powered for faster recognition of network coverage when moving out of a region without coverage. Second, as a method that actually transmits a file and receives an acknowledgement, this method is more informative than simpler methods like Reachability() in areas with weak and/or variable signal strength. A ping with NSURLConnection() typically takes between 0.7 and 1.3 seconds, but was observed to be as slow as 2.5 seconds and as fast as 0.4 seconds. The decision to use this method represents a calculated tradeoff between location accuracy and the amount of bandwidth information that can be obtained. We have chosen to sacrifice precise, real-time measures of throughput that cannot be associated with a location for simple pings that can be linked to a location and time.

4.2.3 Results and interpretation

Transmissions were attempted from 83 different locations around Charlottesville as shown geographically in Figure 7. These data points (green is a successfully acknowledged transmission, red is failure for some reason, blue is the location of the Emergency Department) are overlain on the AT&T service map.

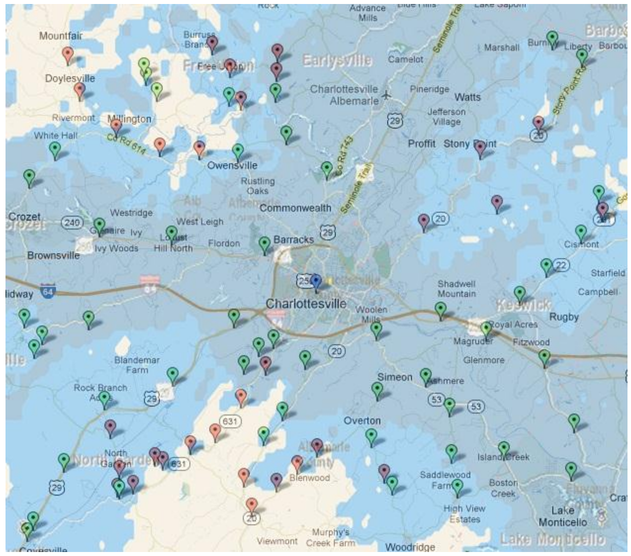


Figure 7: Data points overlain on the AT&T coverage map

It is worth noticing on this map (Figure 7) that the failures are all clustered into three general regions of the geographic space that generally corresponds to the locations AT&T claims to have weaker service.

Figure 8: Ping data from a quadrant of the map

Additionally 6388 pings were captured during the data collection effort. The image in Figure 8, shows all of the captured pings from the north west corner of the considered area. The blue locations indicate ONs, the red locations indicate OFFs. Using this map, it is possible to get a general understanding of the variability present in this factor.

4.3 Forward-looking implementation

Data collection for this model was intentionally performed quickly enough to ensure that no long term trends in the network could be noticed, which is to say, data was collected between AT&T network updates. In practice, AT&T updates the network around Charlottesville every few months. It remains to be seen how long pings are informative for the prediction of future transmissions. In the BreadCrumbs system, ping data was used for future bandwidth prediction for as long as several weeks [19]. There is not enough literature to support any kind of long term plan to deal with the changing temporal autocorrelation of broadband network connectivity data. For this reason, we only use the newest ping data replacing an older point captured at the same location. All of the data, however, are stored for later examination and model improvement. Additional application-level safeguards are present to mitigate the biggest risks, false positives: predicting success when a transmission will actually fail. Since large changes in broadband commercial networks tend to be improvements, while small changes tend to be local outages, it is possible that positive and negative pings may have completely different autocorrelation functions. This is supported, at least in a shorter time frame, by

Cerpa's work [34]. This is a particularly important area for future work, and it could directly build upon and significantly improve this research.

In an effort to provide an estimate to care providers regarding the utility of the implementation of this application in the emergency pre-hospital STEMI care system, a method was devised to estimate a prior probability of successful transmission. Applying the model to the location of every ping in the dataset provides a very rough assessment of this utility. For the Charlottesville area covered in the test effort, 81% of locations measured for connectivity would be predicted as having successful transmissions. This estimate would likely be very different for the catchment areas of other hospitals, but the method provides a rough sense of the potential value for the care providers.

To implement this model in a transmission algorithm for an actual application in Charlottesville, we would recommend a number of considerations. First, a plan needs to be devised for scalability especially with regard to multiple devices. All of the ping and transmission data should be captured and stored centrally for routine analysis. There will be room for improvement of the transmission algorithm over the long run. Maintenance would likely be quite modest. The server-side software is light and could reside on existing hospital machines.

To implement this model in another city, we would recommend verifying the principle assumptions of the model: (1) the network throughput anywhere there is service is very high relative to the 28kB size image file, and (2) that connectivity is generally improving en route to the hospital. If these two are met, there is very little reason to believe that results would be different. Similarity could be validated as quickly as a couple of days of data collection and analysis. This process would also serve the dual purpose of seeding a ping database before any operational runs and providing data for the calculation of a rough prior probability estimate.

Maintenance of the application is likely to be a hospital consideration. The server implementation is light and simple. We expect a well-trained hospital IT staff would have little trouble keeping this system functioning for the foreseeable future.

If significantly greater accuracy is needed for a prediction, the circles used for ping collection could be changed to ellipses. Instead of a radius of all encompassed pings, an ellipse could be stretched along the direction of motion covering the roads the EMT is likely to drive toward the hospital. This solution represents substantially more work than the current implementation for a number of reasons. First, the database query on the phone would become substantially more complicated and memory-intensive. Second, this method would require a road network to be kept on the phone in memory as well as a learning algorithm for likely paths taken by the driver. EMTs are a bit unpredictable, so dictating a route or using data from non-EMT drivers might not be as accurate. It is our belief, however, that this kind of solution could be even more accurate than the model presented above.

CHAPTER 5

5 Evaluation

5.1 Standard methods

The ROC curve for this model on the data set (Figure 9) has an area under the curve of 0.96. This means that the conditions of a randomly chosen pair of success and failure can be distinguished 96% of the time.

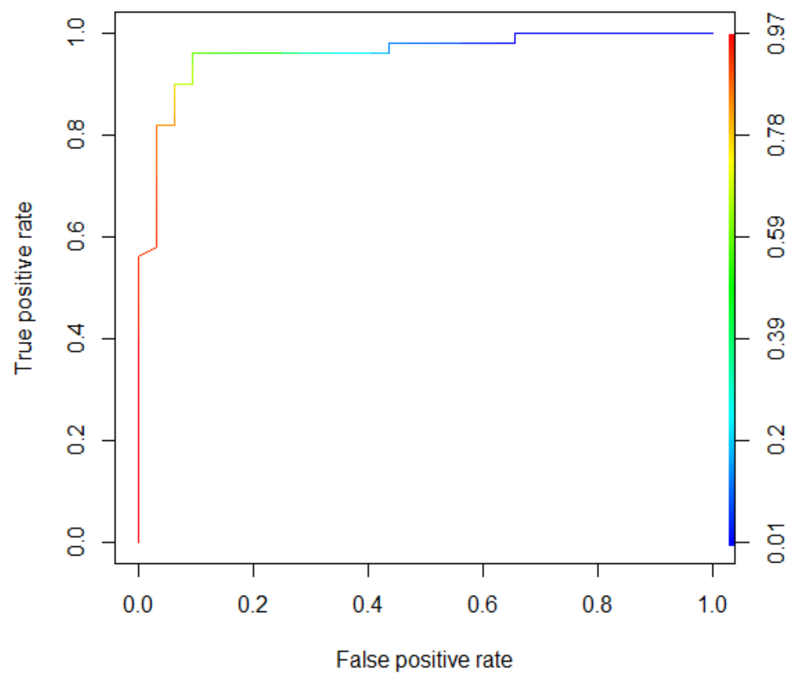


Figure 9: Receiver Operator Characteristic (ROC) curve of the model

As a metric, AUC can be overly confident. Steyerberg et al. outlines a technique to estimate this optimism: the difference between the bootstrapped calculations for AUC estimated on the same and different data. Each bootstrapped sample includes about 63% of the data points in the whole set. Each calculation of AUC can be performed on the same 63% of the data or the other 37%. The difference between these two calculations is an accurate estimate for the final optimism calculation on the full data set [53]. For the logistic model in this thesis, the optimism on AUC is about 0.0004. This is miniscule, and therefore very encouraging for the generalizability of the work.

Figure 10 is another visual representation of the same data in the ROC curve, but from a different perspective. It shows fitted values against the actual results. The perfect model would show all of the points clustered in the top right and bottom left corners. Like the

ROC curve in Figure 9, the actual vs. fitted graph helps show the tradeoffs inherent in choosing a threshold probability.

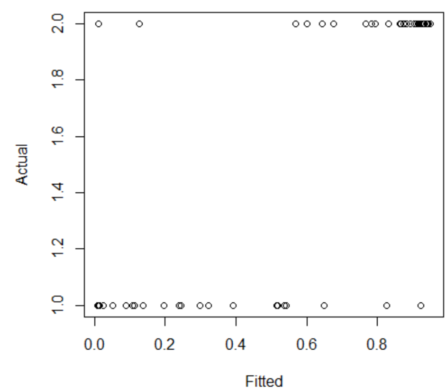


Figure 10: Actual results vs. Fitted values

Bootstrapping was performed to obtain confidence intervals on the model coefficients and the AUC metric. For the model coefficients, it is clear that they are very stable with tight probability distributions. Also, neither has a confidence interval including 0, which indicates significance.

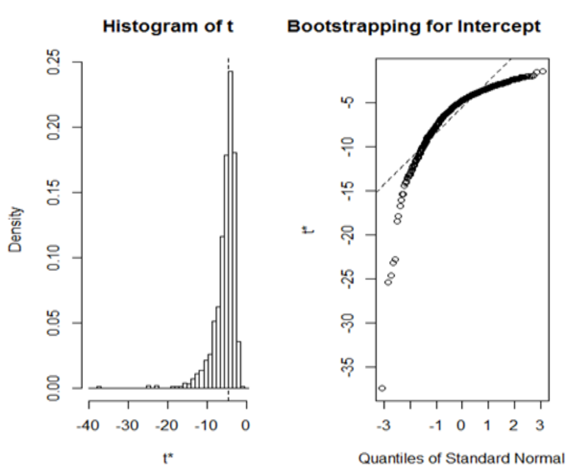


Figure 11: Bootstrapping results for 1000 iterations on the model intercept

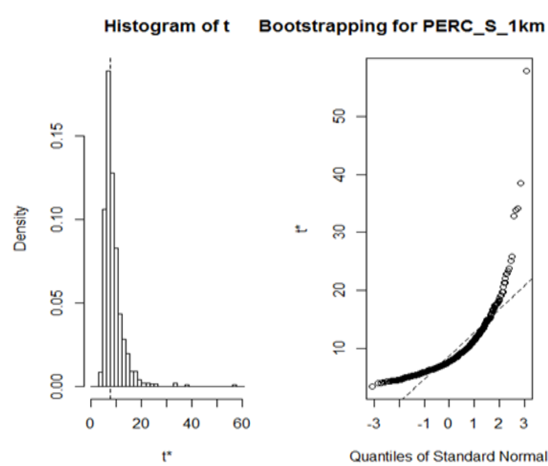


Figure 12: Bootstrapping results for 1000 iterations on the model coefficient

The following is a bootstrapped graph of AUC, area under the ROC curve.

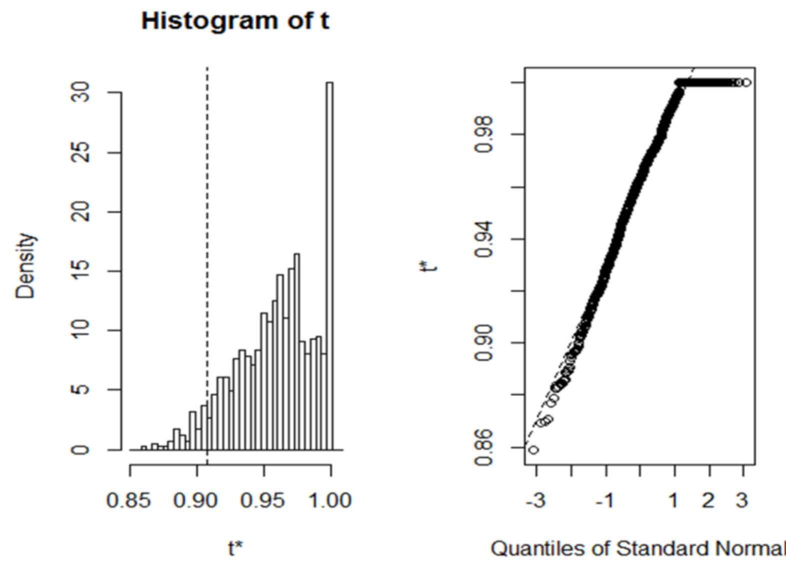


Figure 13: Bootstrapping results for 1000 iterations on the area under the ROC curve

The 95% confidence interval for this metric is 0.895 to 1.0 with a median of 0.963. For the application described in this research, these indicate a very reliable and accurate model.

As mentioned above, evaluation of the model focused on measures of discrimination rather than calibration. That being said, a standard measure of calibration was calculated: Brier Score. This value ranges from 0 to 0.25 representing a perfect model and a completely useless one respectively. For the model discussed herein, the Brier Score was 0.077. This represents decent calibration but not exceptional. The fact that the discrimination metrics are better than the calibration ones is acceptable for this model because of the way it will be used. As implemented, the result is a simple binary indicator. In the future, calibration may be more useful. For now, this Brier Score value is acceptable but not exceptional.

5.2 Geographic cross-validation

Since data were collected in routes and the routes were organized spatially almost like spokes, a cross validation technique was designed to work specifically for this data to evaluate its geographic generalizability. From the 13 routes, 9 contiguous ones were chosen for a training set and the remaining 4 were the test set. This model was then compared with each of the other 12 models created by rotating this setup all the way around. In Figure 14, the points under the blue section are training points, while those under orange are testing points.

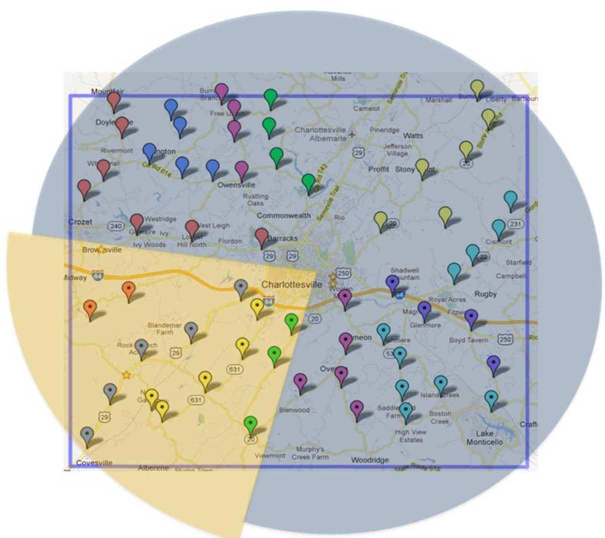


Figure 14: Diagram of Geographic Cross-Validation

The following table shows the results from the Geographic Cross Validation (GCV) rotating technique described above:

Model Number	AUC	Intercept	Ping Coefficient	% ON Cutoff

1	0.962	-4.134	6.829	0.691
2	0.908	-4.448	7.611	0.691
3	0.917	-4.580	7.899	0.824
4	0.938	-4.041	7.209	0.823
5	0.922	-4.316	7.201	0.65
6	1.000	-3.934	6.358	0.65
7	1.000	-4.201	6.616	0.65
8	0.949	-5.808	9.113	0.65
9	0.956	-5.476	8.798	0.666
10	0.956	-5.610	8.998	0.709
11	0.936	-5.738	9.219	0.709
12	1.000	-4.331	6.835	0.709
13	1.000	-4.106	6.634	0.691
ALL DATA*	0.960	-4.666	7.614	0.648
AVERAGE	0.957	-4.671	7.640	0.701
STD DEV	0.034	0.709	1.053	0.059

The important points to note in this table are the consistency across all of the models. While they are not identical, the standard deviations within column are all small relative to the magnitude of the values. Also, the average of all of the models is very similar to the model made with all of the data. This model has an AUC of 0.96 and coefficients as in the equation above.

An additional round of cross-validation was undertaken to ensure that no single route was affecting the model so much as to suspect the claims of geographic generalizability. To prove that no route had too much of an impact, 13 models were built and evaluated using random training and test sets with each of the 13 routes removed from the data set. In Figure 15, the points under the orange section were removed from the data set. A model was built and evaluated using random data points from the remaining points in the blue section.

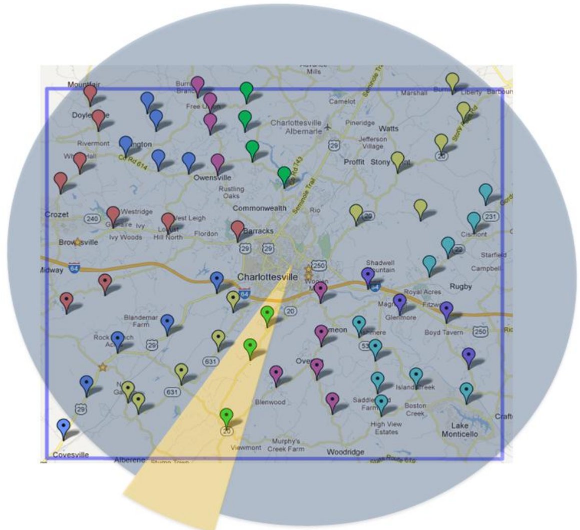


Figure 15: Diagram of route-removed Geographic Cross Validation

The table below shows this data:

Deleted Route	AUC	Intercept	Ping Coefficient	% ON Cutoff
1	0.957	-4.714	7.574	0.697
2	0.964	-4.431	7.395	0.71
3	0.965	-5.694	8.808	0.701
4	0.961	-4.425	7.518	0.69
5	0.972	-5.193	8.542	0.696
6	0.963	-4.495	7.477	0.69
7	0.961	-4.802	7.736	0.692
8	0.962	-4.892	7.811	0.69
9	0.961	-4.620	7.463	0.721
10	0.957	-4.851	7.915	0.696
11	0.986	-6.395	10.353	0.677
12	0.959	-4.761	7.743	0.692
13	0.960	-4.894	7.781	0.715
AVERAGE	0.964	-4.936	8.009	0.697

The obvious outlier in this table is for route 11. This route had more failed transmission attempts than any other. The intercept and coefficient for this model appear very different from the others. We can be reassured though that its AUC is better than the average, and the actual significant variable's cutoff value was still close to the other rows. This table makes a strong claim for the stability of the data and similarity across all 13 of the routes.

5.3 Additional evaluation

This section will provide analysis and insight into the variables that were not used in the model as well as data points for which the model prediction was incorrect.

5.3.1 Insignificant variables

Many of the variables that were expected to be predictive were actually moderately useful, but were not capable of performing better than the percent of ON pings in one kilometer, or when added to a model that also included this variable, did not add any extra performance. These variables dropped out of the feasible models for a number of possible reasons. Some of the variables were dropped out by step-wise regression using AIC. Others were added to models and showed no significance, a confidence interval that included zero, failed a partial F test, or simply did not improve the performance on test sets.

We suspect many of the variables that affect real-time cellular signal strength are actually baked into the ping variables. This is why adding those variables into models was not able to improve them.

For the sake of simplicity, below is a graph of the bootstrapped coefficient values for several of the variables considered. There were many others examined, and variables were discarded for many reasons, but these graphs are straightforward and accessible.

5.3.1.1 Speed

The speed of the vehicle was calculated as an average from recorded GPS readings, and verified against the value recorded on the data collection sheet for that point. We expected this variable to be significant and potentially even non-monotonic. We tried many potential interaction and second order terms as well as simply separating the data set into faster points and slower points. Nothing seemed to make a difference. The graph below is the bootstrapping results from 1000 samples of the coefficient for speed in a logistic regression model that also includes the significant variable, percent of pings that were ON within one kilometer.

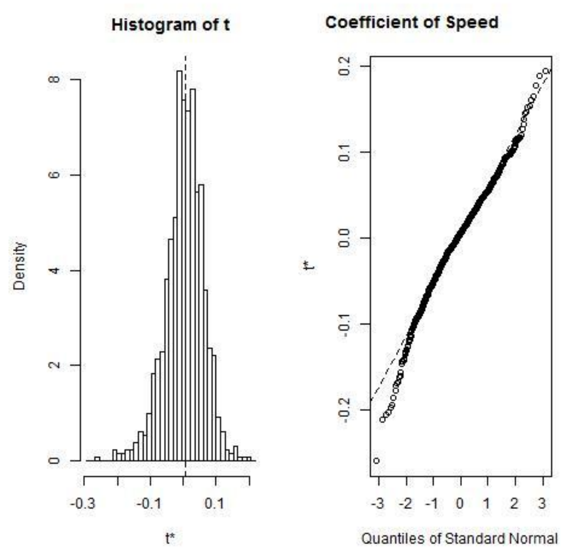


Figure 16: Bootstrapping results for 1000 iterations on a model coefficient for speed

It is clear that the values of the coefficient are very close to normally distributed around 0. This is a strong indication that the variable is not significant.

5.3.1.2 AT&T map

We anticipated this variable to operate like a reasonably accurate prior that could then be updated with real-time information like speed and ping results. While moderately informative by itself, all of the ways that information from this map were captured and added to models with real-time information wound up not improving those models. We tried all sorts of models based on the spot color from which the transmission was initiated with diverse weights for the difference between light blue (2G coverage) and dark blue (3G coverage). We also tried weightings based on the ratio of white to light blue to dark blue in the area around the point of transmission as well as the spot color of points at defined distances towards the hospital. In fact, the color of the point 1 kilometer closer to the hospital from the location of transmission actually showed significance in the wrong direction!

The best results were seen with a simple split based on a binary variable for claimed 3G coverage (dark blue). The graph below is the bootstrapping results from 1000 samples of the coefficient for that binary variable in a logistic regression model that also includes the significant variable, percent of pings that were ON within one kilometer.

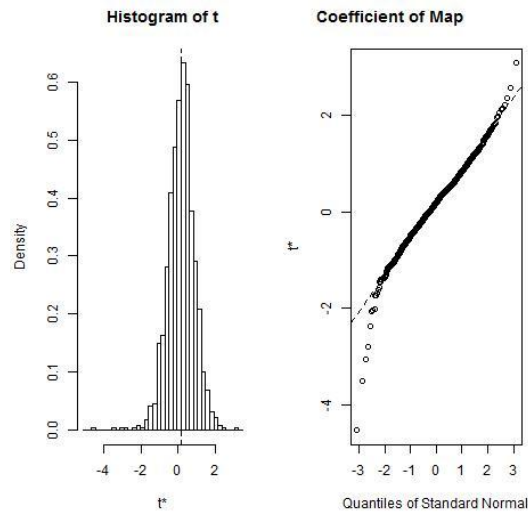


Figure 17: Bootstrapping results for 1000 iterations on a model coefficient for the map

It is clear that the values of the coefficient are very close to normally distributed around 0. This is a strong indication that the variable is not significant.

5.3.1.3 Weather

This variable is a binary categorical for rain. Humidity and wet weather have been shown to have an impact on signal strength, and it was thought that this might affect the probability of successful transmission. This variable was not significant at all. We suspect that if it does play a role, it is captured by the real-time pings. The graph below is the bootstrapping results from 1000 samples of the coefficient for that binary variable in a logistic regression model that also includes the significant variable, percent of pings that were ON within one kilometer.

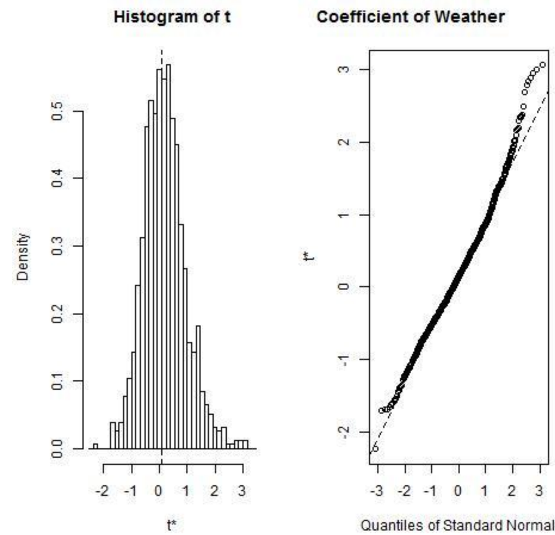


Figure 18: Bootstrapping results for 1000 iterations on a model coefficient for the weather

It is clear that the values of the coefficient are very close to normally distributed around 0. This is a strong indication that the variable is not significant.

5.3.1.4 Time of the day

It is fairly common anecdotally that the time of day can have an impact on bandwidth in general. Peak times tend to be reported as late afternoon to early evening. We kept track of the time of day that all of the transmissions occurred, and attempted to determine whether or not this played a role. We attempted categorical variables for different chunks of time: morning, afternoon, evening, and what we presumed to be the peak time (5pm to 9pm). None of these were significant at all. The graph below is the bootstrapping results from 1000 samples of the coefficient for the binary variable associated with peak time in a logistic regression model that also includes the significant variable, percent of pings that were ON within one kilometer.

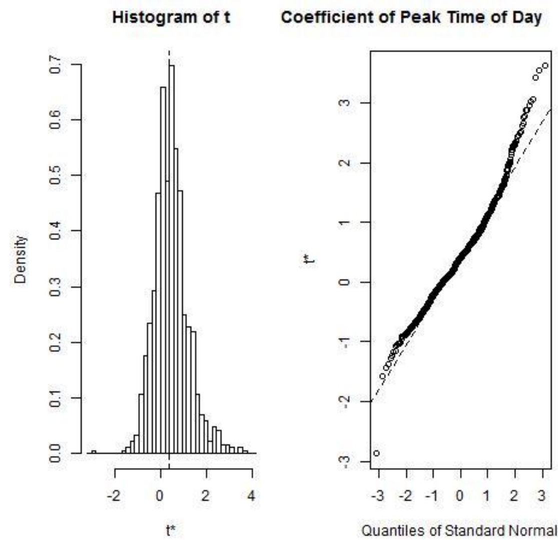


Figure 19: Bootstrapping results for 1000 iterations on a model coefficient for the time of day

It is clear that the values of the coefficient are very close to normally distributed around 0. This is a strong indication that the variable is not significant.

5.3.1.5 Day of the week

We also considered the possibility that the day of the week would make a difference. After trying several combinations of categorical variables, nothing was significant. The graph below is the bootstrapping results from 1000 samples of the coefficient for a binary variable associated with whether or not the day was over the weekend. The variable was added to a logistic regression model that also includes the significant variable, percent of pings that were ON within one kilometer.

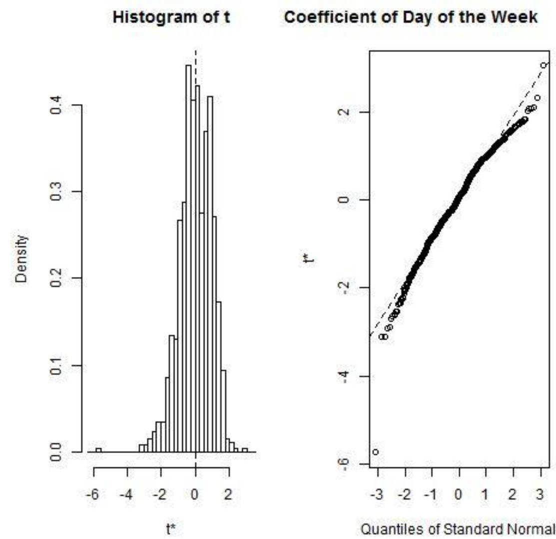


Figure 20: Bootstrapping results for 1000 iterations on a model coefficient for the day of the week

It is clear that the values of the coefficient are very close to normally distributed around 0. This is a strong indication that the variable is not significant.

5.3.1.6 Spatial Autocorrelation

In modeling anything with a spatial component, there must be a presumption that autocorrelation could exist. We attempted to find it in our response variable. We attempted to add factors like "distance to closest success" and "distance to closest failure." The graph below is the bootstrapping results from 1000 samples of the coefficient for the quantitative variable associated with the ratio of the distance from a success to the distance from a failure. The variable was added to a logistic regression model that also includes the significant variable, percent of pings that were ON within one kilometer.

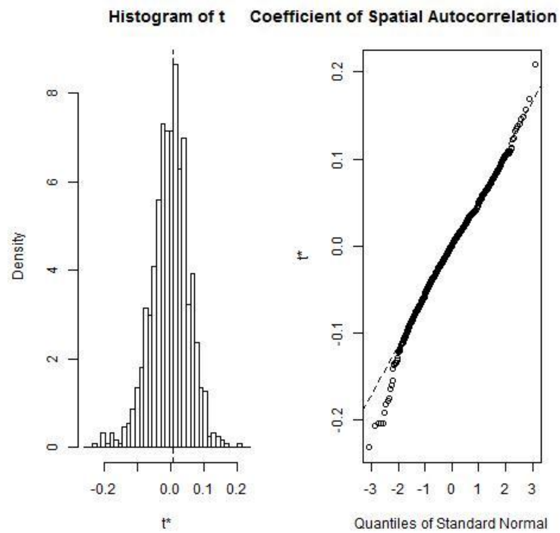


Figure 21: Bootstrapping results for 1000 iterations on a model coefficient for spatial autocorrelation

It is clear that the values of the coefficient are very close to normally distributed around 0. This is a strong indication that the variable is not significant.

5.3.1.7 Elevation

We have heard anecdotally and experienced ourselves that elevation has an impact on cellular service. This variable was also not significant at all. It is likely this variable would have more of an impact if the vehicle were not in motion. The graph below is the bootstrapping results from 1000 samples of the coefficient for elevation as it was added to a logistic regression model that also includes the significant variable, percent of pings that were ON within one kilometer.

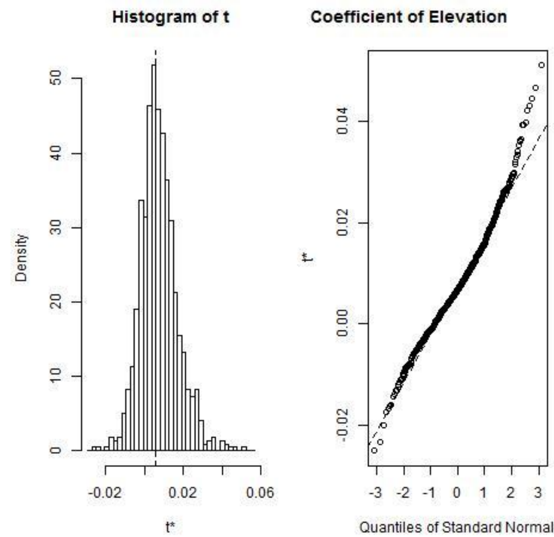


Figure 22: Bootstrapping results for 1000 iterations on a model coefficient for elevation

It is clear that the values of the coefficient are very close to normally distributed around 0. This is a strong indication that the variable is not significant.

5.3.2 Algorithmic failures

While somewhat aside from formal evaluation metrics, an analysis of every point in the data set for which the algorithm miscalculated, there are mitigating circumstances. All of the false negatives (predicted failure but observed success) appear simply to have switched from fail to success too late. In reality, if an EMT were to attempt a transmission when the algorithm actually does switch over to predicting success (a matter of seconds down the road), the transmission would have arrived at exactly the same time because it will not actually begin to transmit until the same region of service is entered. All of the false positives (predicted success but observed failure) were only failures because not all of the ECG was received in time. In these points, more than 80% of the image was successfully uploaded. In many cases, this amount of the area of the ECG transmission will be enough for a physician to make a diagnosis.

5.3.2.1 False negatives: predicted failed transmission, observed successful

Point 28

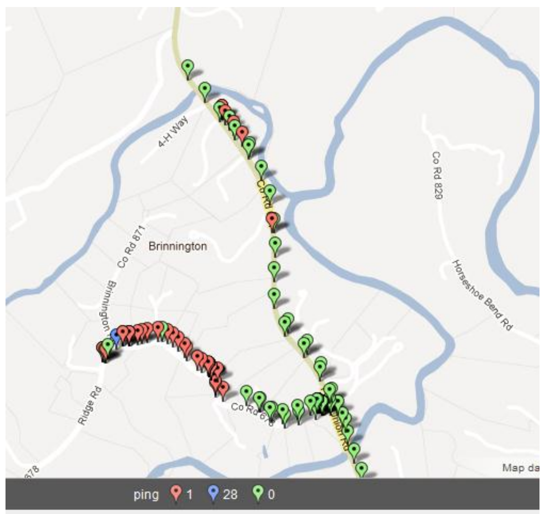


Figure 23: Map of pings around point 28

From the blue point, the car drove east and then south. This represents an interesting point in that there is very little data on one side of the point. In fact, almost all of the data is coming from the direction to which the vehicle will soon be heading. Another issue here is that there are a lot more points in the region where the car is driving slowly. In implementation, the model would update to success pretty shortly along in the drive. If an EMT waited 1 minute to hit send (around the point when it would turn green), the application actually wouldn't have lost any time in delivering the file because it would have been sent once the phone returned to service anyway. This kind of false negative is actually more like a surprise success than an observed failure.

Point 69

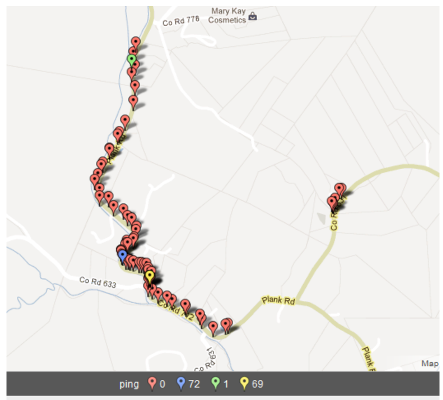


Figure 24: Map of pings around point 69

From this map, the hospital is to the north-north east. There was a failure here going east directly towards the hospital (this transmission was begun from the yellow flag) and a success driving west towards Route 29 (this transmission was begun from the blue flag). OFF pings are all around but near a location of decent service visible in the map below. The vehicle barely made it into the region in the two minutes with enough time for the transmission to succeed.

The map below shows the pings for a larger area in which the pocket of service is visible.

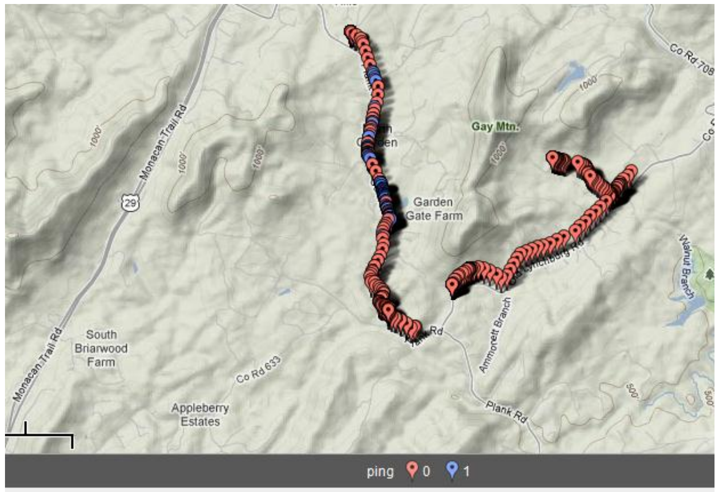


Figure 25: Map 2 of pings around point 69

This pocket, strangely enough, is in between two small mountains. From this map, it is clear that this is actually a small development.

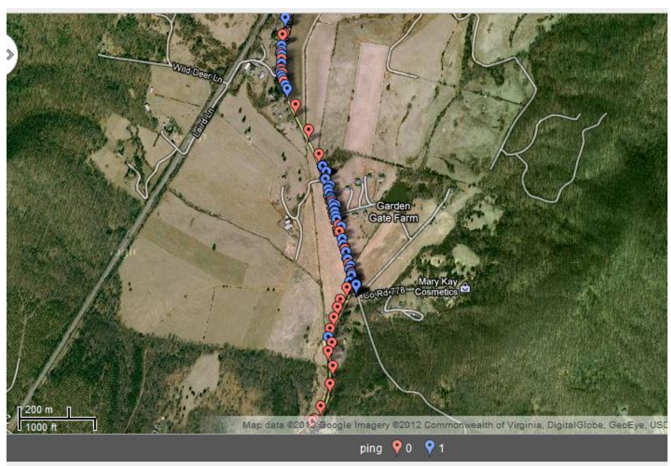


Figure 26: Map 3 of pings around point 69

We believe this is an understandable miss for the algorithm, and would be mitigated by EMT context awareness. On the bright side, this is a perfect opportunity for algorithm improvement. Whenever successes are recorded in areas like this, EMTs and the algorithm alike can know to take advantage of it. Also, inside the small pocket of coverage, the indicator would likely change to green. EMTs who have used the system a few times before and know their districts might know to stop the ambulance for 30 seconds at this point in the road to wait for the transmission to succeed. In this kind of situation, the algorithm is still able to provide information to an EMT even while failing to perform the prediction correctly.

5.3.2.2 False positives: predicted successful transmission, observed failure

Point 79

At this point, the value of Percent_ON_1km is 0.694, which is only slightly above the threshold. Of the 16 file transmission, 13 were received and 12 were acknowledged back to the iPhone as having been received.

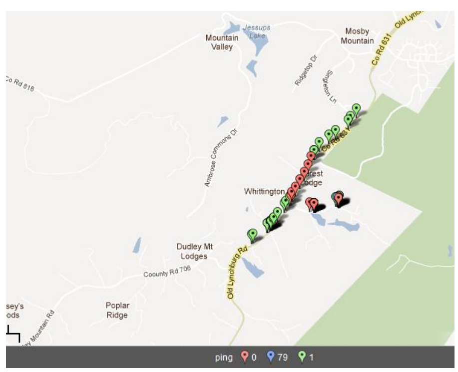


Figure 27: Map of pings around point 79

The blue point from which the transmission was made is hidden behind the right-most OFF ping, and the trajectory was west to Old Lynchburg Road and then north. This map is pretty close into Charlottesville, just a little south of the city. The map below is all of the pings within 2km.

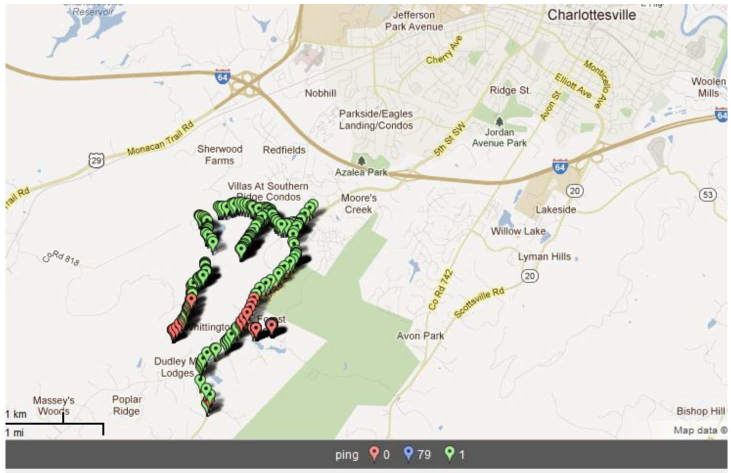


Figure 28: Map 2 of pings around point 79

This is a point without any points behind it. This could have been done better with more careful testing. Pings to the east of the blue point would likely have been OFFs and would have dropped the Percent_ON_1km down below the 69% it returned. This was almost a predicted failure. Also, though we predicted success and it was a failure, 13 of the 16 files were received. In an operational setting, depending on the ECG, this may have actually been enough of the image for a doctor to read it.

Point 78

At this point, the value of Percent_ON_1km is 0.818. Of the 16 file transmission, 14 were received and all 14 were acknowledged back to the iPhone as having been received.

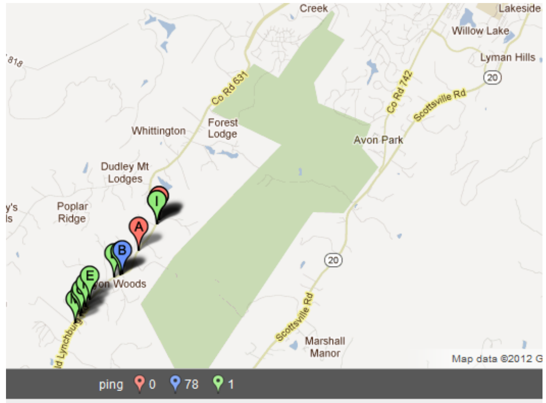


Figure 29: Map of pings around point 78

This transmission has an exceptionally small number of nearby pings. Here it is for 2km:

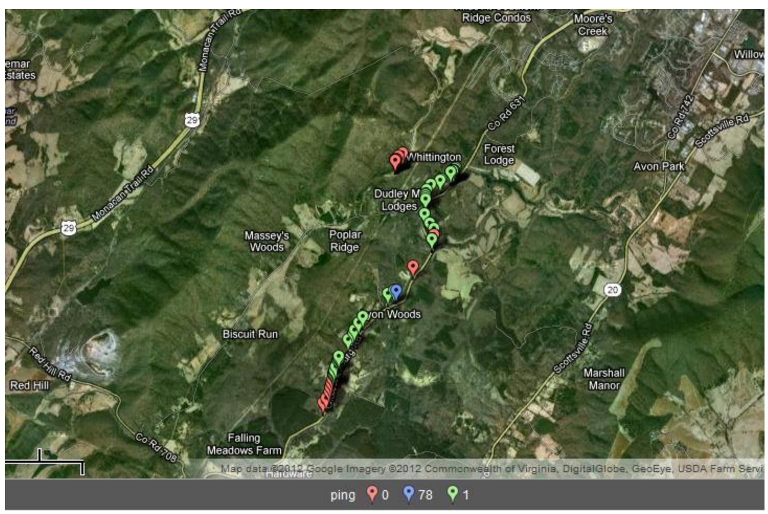


Figure 30: Map 2 of pings around point 78

The four previous pings for this point were successful. The prediction on this road is likely to improve with time. It looks like something on the phone got hung up here while driving past the Rivanna Rifle and Pistol Club, an area known to be heavily wooded. Maybe this was just an outlier in the amount of time it took to capture pings in this area. On the map, it is clear that there is a big space with no pings. Transmission was attempted immediately after a string of positives while heading into a dead zone. The phone basically just did not recover service in time to complete the transmission. It probably would have worked if the vehicle had driven faster or the transmission was attempted 30 seconds later. Again, the vast majority of the image was received and may have been readable at this point.

Point 13

At this point, the value of Percent_ON_1km is 0.9375. Of the 16 file transmission, 14 were received and 12 were acknowledged back to the iPhone as having been received. The point is right behind the OFF ping in the middle.

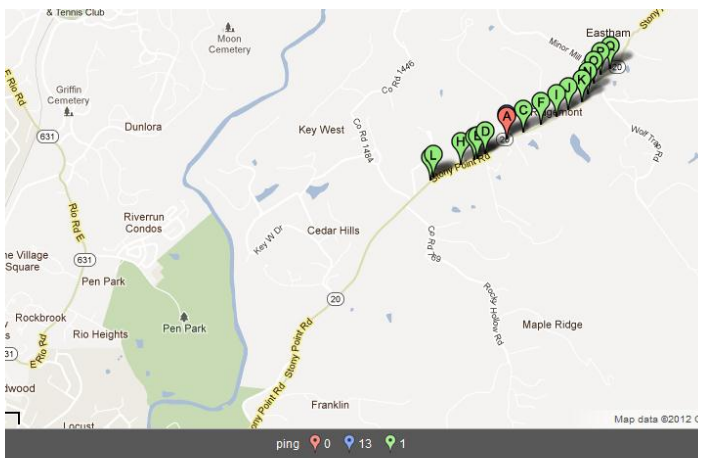


Figure 31: Map of pings around point 13

Here it is at 2km in the direction of motion:

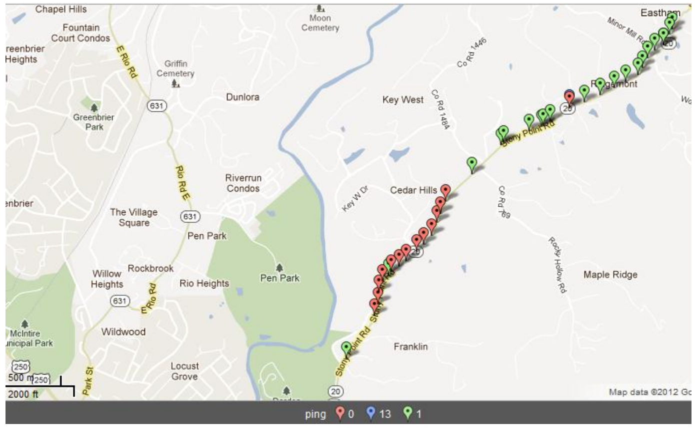


Figure 32: Map 2 of pings around point 13

This is an exceptionally difficult point to predict. The transmission was sent from the middle of a pocket of good service and most of the packets were received before the vehicle slipped into bad service. This road goes from bad to good and then to bad and then to good again in surprisingly sharp changes. While the algorithm missed the prediction, most of the image was received. It is our belief that simply driving slower (the point was collected at 43mph) would have made that little bit of difference. Also, this transmission was sent from just about the only point on the route where the algorithm would have missed. A minute further along and the percent of ON pings would have been low enough to catch the failure.

CHAPTER 6

6 Conclusion

This research developed and validated a data-centric algorithm for predicting the likelihood of success of a near real-time mobile health transmission from a moving ambulance. The unique nature of pre-hospital STEMI care makes this likelihood especially useful to EMTs in the field. The algorithm was implemented for and deployed in an iPhone 4 application that is part of a pre-hospital transmission system. The key insight developed is that simple, nearby measures of connectivity are predictive of the time-critical success of small, mobile transmissions. This insight and its developed method were validated in unique, geographic-based cross validation techniques to promote generalizability. This generalizability, however, is bounded by the nature of the city around which the work was performed. The size of the transmission file is small relative to the maximum throughput of the potential networks, and bandwidth tends to increase from any geographic location bound for the hospital because of its central location. Expansion of this method to other cities should consider whether these two factors are similar.

As a simple, smart transmission method, this system could be a valuable part of emergency medical response particularly in rural America. Additionally, as part of a flexible, expandable framework, these results are potentially quite valuable to the field of emergency pre-hospital care and wireless telemetry at large. It is our intention to offer this system gratis along with an implementation guide and user manual.

As a framework, this method of data collection, model building, and robust evaluation can be applied more broadly to other wireless health systems. Any time immediate response is possible from near real-time data, there could be a value to forecasting the end-to-end quality of service.

It is our hope that future work on this project will focus on several key areas around getting and exploiting significantly more data: fuller understanding of the temporal autocorrelation features of the pings, model improvement specifically geospatial autocorrelation with many more successful and failed transmissions, and platform expansion to Android and other provider networks. Vehicle trajectory predictions based on previous route data is a fruitful field of research. After some time in operational deployment, leveraging this data to improve the algorithm may prove advantageous.

With the infrastructure in place to transmit data from ambulances to hospitals, it is also our hope that HIPAA requirements can be addressed to allow for patient identifying transmissions like name and date of birth. In the longer term, integration with a platform like Hermes [54] could dramatically increase the number of places where an application like this could be useful. In the near term, the application and algorithm will be tested extensively by students and care providers associated with the University of Virginia before operational deployment to real hospital care providers.

References

1. AT&T Coverage Viewer. Available online: http://www.wireless.att.com/coverageviewer.
2. Verizon Wireless Coverage Locator. Available online: http://www.verizonwireless.com/wireless-coverage-area-map.shtml.
3. Sprint Coverage Check. Available online: http://coverage.sprint.com/.
4. D. L. Huff, "AT&T cellular technology review" Military Communications Conference, vol. 2, pp 490-494, October 1985.
5. Abdullah Konak, “A kriging approach to predicting coverage in wireless networks” International Journal of Mobile Network Design and Innovation, vol. 3, no. 2, pp 65-71, January 2009.
6. Heping Liu, Soroor Al-Khafaji, and Alice Smith, “Prediction of wireless network connectivity using a taylor kriging approach” International Journal of Advanced Intelligence Paradigms, vol. 3, no. 2, pp 112-121, April 2011.
7. Cesar D. Guerrero, Miguel A. Labrador, “Traceband: a fast, low overhead and accurate tool for available bandwidth estimation and monitoring” Computer Networks, vol. 54, issue 6, pp 977-990, April 2010.
8. Jacob Strauss, Dina Katabi, and Frans Kaashoek, “A measurement study of available bandwidth estimation tools” In Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement, pp 39-44, 2003.
9. M. Jain and C. Dovrolis, “Pathload: a measurement tool for end-to-end available bandwidth” In Passive and Active Measurements, Fort Collins, CO, March 2002.

10. V. J. Ribeiro, R. H. Riedi, R. G. Baraniuk, J. Navratil, and L. Cottrell, “pathChirp: efficient available bandwidth estimation for network paths” In Passive and Active Measurements, 2003.

11. Cao Le Thanh Man, Go Hasegawa, Masayuki Murata, “ImTCP: TCP with an inline measurement mechanism for available bandwidth” Computer Communications, vol. 29, issue 10, pp 1614-1626, June 2006.

12. E. Bergfeldt, S. Ekelin, and J. M. Karlsson, “Real-time bandwidth measurements over mobile connections” European Transactions on Telecommunications, vol. 22, pp 255–267, 2011.

13. C.D. Guerrero, M.A. Labrador, "A hidden markov model approach to available bandwidth estimation and monitoring" IEEE Internet Network Management Workshop, pp.1-6, October 2008.

14. A. Capone, L. Fratta, F. Martignon, "Bandwidth estimation schemes for TCP over wireless networks" IEEE Transactions on Mobile Computing, vol.3, no.2, pp 129-143, June 2004.

15. Dimitrios Koutsonikolas, Y. Charlie Hu, "On the feasibility of bandwidth estimation in 1x EVDO networks" Proceedings of the 1st ACM workshop on mobile internet through cellular networks, pp 31-36, September 2009.

16. Keon Jang, Mongnam Han, Soohyun Cho, Hyung-Keun Ryu, Jaehwa Lee, Yeongseok Lee, and Sue B. Moon, “3G and 3.5G wireless network performance measured from moving cars and high-speed trains” In Proceedings of the 1st ACM workshop on mobile internet through cellular networks, pp 19-24, 2009.

17. S. Helhel, Ş. Özen, and H. Göksu, "Investigation of GSM signal variation depending weather conditions" Progress In Electromagnetics Research B, vol. 1, pp 147-157, 2008.

18. David Cottingham, "Vehicular wireless communication," PhD thesis, University of Cambridge, September 2008.

19. Anthony J. Nicholson and Brian D. Noble, "BreadCrumbs: forecasting mobile connectivity" In Proceedings of the 14th ACM international conference on mobile computing and networking, pp 46-57, 2008.

20. Petcher, "QoS in wireless data networks" 2006. Available online: http://www.cs.wustl.edu/~jain/cse574-06/ftp/wireless_qos/index.html.

21. K. Ahuja, R. Khanna, B. Singh, "Real time available-bandwidth estimation (ABE) algorithm based selection in heterogeneous network for WiMAX and 3G" 5th International Conference on Next Generation Mobile Applications Services and Technologies, pp 169-174, September 2011.

22. Y. Shang, W. Rumi,Y. Zhang, M. Fromherz, "Localization from connectivity in sensor networks," IEEE Transactions on Parallel and Distributed Systems, vol. 15, no.11, pp 961- 974, November 2004.

23. Kezhong Liu, Shu Wang, Yongqing Ji, Xing Yang, Fuping Hu, "On connectivity for wireless sensor networks localization" International Conference on Wireless Communications Networking and Mobile Computing, vol. 2, pp 879-882, September 2005.

24. J. Ott, D. Kutscher, "Drive-thru internet: IEEE 802.11b for 'automobile' users" INFOCOM Joint Conference of the IEEE Computer and Communications Societies, vol. 1, March 2004.

25. Wei Kuang Lai, Kai-Ting Yang, Jia-Liang Guo "A novel bandwidth prediction and queue management scheme for public transportation systems" 2011 Global Mobile Congress, pp 1-6, October 2011.

26. Jing Zhao, Todd Arnold, Yang Zhang, and Guohong Cao, "Extending drive-thru data access by vehicle-to-vehicle relay" In Proceedings of the fifth ACM international workshop on Vehicular Inter-networking, pp 66-75, 2008.

27. Jing Zhao; Yang Zhang; Guohong Cao, "Data pouring and buffering on the road: a new data dissemination paradigm for vehicular ad hoc networks," IEEE Transactions on Vehicular Technology, vol. 56, no. 6, pp 3266-3277, November 2007.

28. Jaehoon Jeong, Shuo Guo, Yu Gu, Tian He, D.H.C. Du, "TSF: trajectory-based statistical forwarding for infrastructure-to-vehicle data delivery in vehicular networks" IEEE International Conference on Distributed Computing Systems, pp 557-566, June 2010.

29. Draft amendment to standard for information technology telecommunications and information exchange between systems local and metropolitan area networks specific requirements Part 11: wireless LAN medium access control(MAC) and physical layer specifications Amendment 7: wireless access in vehicular environment, 2007.

30. P.N. Pathirana, A.V. Savkin, S. Jha, "Location estimation and trajectory prediction for cellular networks with mobile base stations" IEEE Transactions on Vehicular Technology, vol. 53, no. 6, pp 1903-1913, November 2004.

31. D. Valerio, A. D'Alconzo, F. Ricciato, W. Wiedermann, “Exploiting cellular networks for road traffic estimation: a Survey and a research roadmap," IEEE Vehicular Technology Conference, pp 1-5, April 2009.
32. Jie Sun, Zhao-hui Wu, Gang Pan, “Context-aware smart car: from model to prototype” Journal of Zhejiang University – Science A, vol. 10, no. 7, pp 1049-1059, July 2009.
33. R. Bruno, M. Conti, "Throughput and fairness analysis of 802.11-based vehicle-to-infrastructure data transfers” IEEE International Conference on Mobile Adhoc and Sensor Systems, pp 232-241, October 2011.
34. Alberto Cerpa, Jennifer L. Wong, Miodrag Potkonjak, and Deborah Estrin, “Temporal properties of low power wireless links: modeling and implications on multi-hop routing” In Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing, pp 414-425, 2005.
35. American Heart Association: Mission Lifeline. Available online: http://www.heart.org/missionlifeline.
36. G. De Luca, H. Suryapranata, JP Ottervanger, et al., “Time delay to treatment and mortality in primary angioplasty for acute myocardial infarction: every minute of delay counts” Circulation, vol. 109, pp 1223-1225, 2004.
37. Dr. David Burt. Personal conversation. 23 June 2011.
38. P Giovas, D Papadoyannis, D Thomakos, et al. “Transmission of electrocardiograms from a moving ambulance” Journal of Telemedicine and Telecare, vol. 4, 1998.
39. FastECG: Wireless ECG Transmission. Available online: http://www.fastecg.com/.

40. T. Gao, et al., “Wireless medical sensor networks in emergency response: implementation and pilot results” IEEE International Conference on Technologies for Homeland Security, May 2008.

41. N. Versel, “Tucson shuts down ambulance-based telemedicine network” MobiHealthNews 27 January 2011. Available online: http://mobihealthnews.com/10070/tucson-shuts-down-ambulance-based-telemedicine-network/.

42. Yuechun Chu and Aura Ganz, “A mobile teletrauma system using 3G networks” IEEE Transactions on IT in Biomedicine, vol. 8, no. 4, December 2004.

43. C. Chang, et al., “Development of a ubiquitous emergency medical service system based on zigbee and 3.5G wireless communication technologies” Medical Biometrics Lecture Notes in Computer Science, vol. 6165, pp 201-208, 2010.

44. Konstantinos Banitsas, Sapal Tachakra, Euclid Stefanidis, Konstantinos Boletis, "Using HSPA to improve the telemedical links on a moving ambulance" International Conference of the IEEE Engineering in Medicine and Biology Society, pp 739-742, August 2008.

45. Neil Versel, “Mobile telemedicine system beams live video from ambulance to Cincinnati Children's” Fierce Mobile Healthcare, August 2010.

46. Kristen Gruenther, Alex Gessner. “ECG Interpretability Survey” SurveyGizmo. Available online: http://students.sgizmo.com/testsurvey/survey?id=606051&crc=45d2fc2f74f455399f0df2096200323a.

47. E Vittinghoff, CE McCulloch, “Relaxing the rule of ten events per variable in logistic and Cox regression” American Journal of Epidemiology, pp 710-718, 2007.
48. M. Zhao, F. Ge, T. Zhang, and Z. Yuan, “AntiMalDroid: an efficient SVM-based malware detection framework for android” In Proceedings of ICICA, pp 158-166, 2011.
49. I. J. D. Gonzales, “Fall detection using a smartphone” Master’s thesis, Gjovik University College, 2011.
50. P. Peduzzi, J. Concato, E. Kemper, TR Holford, AR Feinstein, “A simulation study of the number of events per variable in logistic regression analysis” Journal of Clinical Epidemiology, pp 1373–1379, 1996.
51. JA Hanley, BJ McNeil, “A method of comparing the areas under receiver operating characteristic curves derived from the same cases” Radiology, vol. 148, pp 839–843, 1983.
52. Kaggle Data Prediction Competitions. Available online: http://www.kaggle.com.
53. Ewout W Steyerberg, Frank E Harrell Jr, Gerard J.J.M Borsboom, M.J.C Eijkemans, Yvonne Vergouwe, J.Dik F Habbema, “Internal validation of predictive models: efficiency of some procedures for logistic regression analysis” Journal of Clinical Epidemiology, vol. 54, issue 8, pp 774-781, August 2001.
54. Aditya Dhananjay, Ashlesh Sharma, Michael Paik, Jay Chen, Trishank Karthik Kuppusamy, Jinyang Li, and Lakshminarayanan Subramanian, “Hermes: data transmission over unknown voice channels” International conference on Mobile computing and networking, pp 113-124, 2010.

55. Bernard J. Gersh, Elliot M. Antman, "Selection of the optimal reperfusion strategy for STEMI: Does time matter?" European Heart Journal, vol. 27, issue 7, pp 761-763, April 2006.
56. American Hospital Association: Fast Facts on US Hospitals. Available online: http://www.aha.org/research/rc/stat-studies/fast-facts.shtml.
57. W. Clarke, S. Anderson, M. Breton, S. Patek, L. Kashmer, and B. Kovatchev "Closed-loop artificial pancreas using subcutaneous glucose sensing and insulin delivery and a model predictive control algorithm: the Virginia experience" Journal of Diabetes Science and Technology, vol. 3, no. 5, pp 1031-1038, 2009.
58. M. Colunas, J. Fernandes, I. Oliveira, and J. Cunha, "Droid jacket: Using an android based smartphone for team monitoring" Wireless Communications and Mobile Computing Conference (IWCMC), pp 2157-2161, 2011.
59. W. Eberle, J. Penders, and R. Yazicioglu, "Closing the loop for deep brain stimulation implants enables personalized healthcare for Parkinson's disease patients" IEEE Engineering in Medicine and Biology Society, pp 1556-1558, 2011.
60. G. Power, E. Dillon, and F. Cleary, "Seamless mobile communications for mhealth" Journal of eHealth Technology and Application, vol. 8, no. 1, pp 32-35, 2010.

Appendices

Data collection sheet

The following is a table that was used to collect data from one of the 13 routes that were driven. Many of these variables were captured by the iPhone automatically, but they exist on the data collection sheet for verification. Points that were added to the collection effort were scribbled into the sides or bottom of the sheet.

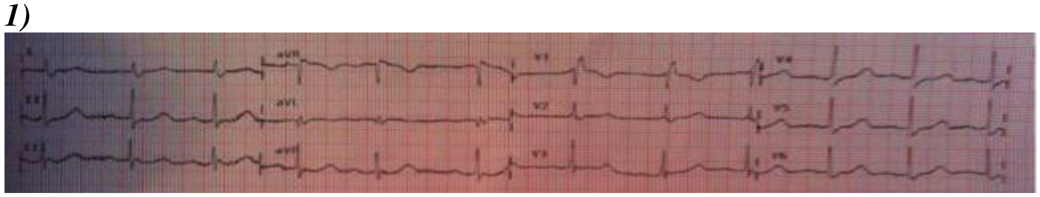
Route: magenta

Name, time, date, weather:						
#	**Intersection**	**Latitude, longitude**	**Speed**	**Amount of time it took**	**# of files received (<=16)**	**Your interpretation of sending from this point**
37	Free Union Road and Homestead Farm Lane					
40	Free Union Rd and Old Free Union Road					
46	Ridge Road and Brinnington Road					
44	Garth Road and Foxfield Track					

Compression survey

The ECG interpretability survey described above has been deployed online via SurveyGizmo [47] but no data have yet been collected. For each of 20 ECGs, the responder is randomly served either a compressed or uncompressed photograph along with a simple story that is similar to what they would know in a real situation. There are two questions from the survey below. They have the same story and ECG except that one of them has been compressed. The responders will only have the opportunity to answer one of these two. Additionally, of the 20 questions the responders will answer, they will be presented in a random order. Since some ECGs are harder than others, we do not expect every responder to be correct each time. We intend simply to compare whether or not the compression (answering question 1 instead of question 2) had an impact on the decision that was actually made. The statistical testing for this effort will include blocking on the responder to control for individual variability in training / experience / skill.

1)

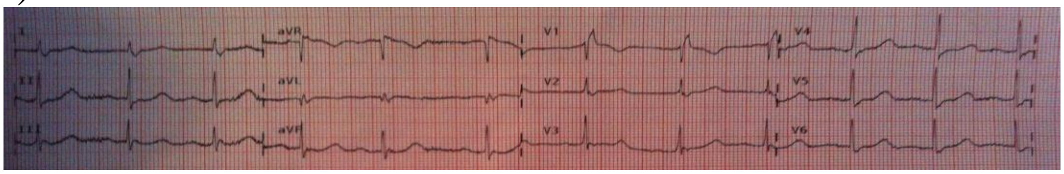


An 83 year old male with a history of diabetes and hypertension is complaining of intermittent sternal chest pain radiating to his back.
Based on the above ECG and scenario, would you activate the cath lab to treat for a presumed STEMI?*

( ) Yes

( ) No

2)



An 83 year old male with a history of diabetes and hypertension is complaining of intermittent sternal chest pain radiating to his back.
Based on the above ECG and scenario, would you activate the cath lab to treat for a presumed STEMI?*

( ) Yes

( ) No