Future Location Prediction Using Data Compression Models

Ingrid Burbey
Virginia Polytechnic Institute and State University

Future Location Prediction Using Data Compression Models

by

Ingrid Burbey

Preliminary Examination

submitted to the faculty of

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Computer Engineering

Dr. Thomas L. Martin, Chairman

Dr. Mark Jones

Dr. Scott F. Midkiff

Dr. Manuel A. Pérez-Quiñones

Dr. Joseph Tront

26 February 2008

Blacksburg, VA

Table of Contents

From [1]

1 Introduction

One can easily imagine many useful applications that are supported by the ability to predict a user's next location. For example, a context-aware to-do list could remind someone to buy groceries on the way home from work. A mobile worker could be notified to cache data before they are about to enter areas with no wireless connectivity [2]. Location-awareness has been used in smart homes to adaptively control heat and lighting [3]. IBM [4], Intel [5] and Microsoft [6] are all doing research where location is used to communicate someone's current status. DARPA's Strategic Technology Office has initiated the Location and Connection Aware Content Pushing (LOCO) Program, with the goal to "provide information to commanders before they need it by anticipating their needs" [7].

Applications such as reminders, context-aware to-do lists, connectivity notification and content-pushing are all single-user applications.

Introduction

If location prediction is expanded to predict not only the next location but also future locations and the times when someone was going to be there, and this information is shared with other users, several interesting collaborative, multi-user applications can be supported, as illustrated in this scenario.

Among their other duties, Alice and Bob are working together on a proposal. They work fairly independently, Bob usually from his office on campus and Alice from her home office. Even though she has a desk on campus, Alice prefers her quiet home office and the flexible schedule it supports.

Bob is working on a graphic for the proposal and wants to discuss it with Alice. It is not a high priority discussion, but it would be most efficient to edit the image together, so Bob wonders if Alice will be coming onto campus sometime in the next few days. He does not want to interrogate Alice about her plans, so he signs onto the prediction server to see when Alice will be on campus. The response indicates that there is an 85% probability that Alice will be at her desk between 1:00pm and 4:00pm that day. Bob makes a note to himself to stop in her office after he returns from lunch.

At the same time, Alice's PDA launches her context-aware to-do list, which reminds her that, if she goes on campus, she may want to return the library books that are due soon. She takes a moment to put the books in her briefcase. She notices that Bob has queried the system about her availability. There is a library book-drop between her office and Bob's; if Bob stops in, perhaps he can drop off the books on his way back to his office.

Alice is also a member of a new research group on campus. If her colleagues are available, she would like to meet with them to discuss ideas for collaboration. She does not know these people very well and is reluctant to take on the chore of trying to coordinate *everyone's schedules. They have all subscribed to the prediction server, so within minutes, Alice has found a block of time when all of them are on campus, and not in classrooms teaching. She composes an email to the group suggesting a time and a conveniently located conference room. Now that the small details of her days are taken care of, Alice can focus on her research.*

This scenario illustrates several applications which are possible when location predictions are shared between multiple users. Computer-supported-collaborative-work (CSCW) is enhanced when co-workers have an awareness of when others are available or un-interruptible. Paul Dourish writes, "Awareness support in computer systems' design is crucial, since knowledge of group and individual activity, and co-ordination are central to successful co-operation" [8]. Impromptu meetings make up over half of all office interactions [9]. Predictions can provide opportunities to rendezvous, if they reveal that two people are going to be in the same or nearby location at the same time.

Future location prediction can also support the disabled or aged by providing reminders or monitoring for out-of-the-ordinary activities. An example of this is the "Opportunity Knocks" system [10] which monitors a user's location as he rides the bus. If he seems to be headed in the wrong direction, his cell phone "knocks" to check if this is a problem. If he is on the wrong bus, the application consults the website for the bus schedule and provides instructions on how to get off at the next stop and get to the correct bus.

Location prediction could also be used as a substitute for invasively tracking employees[11]. Current employee tracking systems constantly watch employees, potentially destroying employee trust and privacy. With a prediction system, employee location information could be stored privately on the employee's personal device. The device could then be queried for a prediction of when an employee might arrive at a given destination. For example, instead of tracking computer maintenance technicians in order to find the closest technician when a problem occurs, the system could simply poll the technicians' devices to find which technician is likely to come to the problem location and when. Current tracking systems could tell where the technician is located, but not when he will be available to work on the next problem. A prediction system could predict which technician is likely to be in the area along with an estimated time of arrival.

Of course, destination prediction is a marketer's dream. If a system can predict that a group is leaving their office to go to lunch, it can target advertising [12] and coupons to the group.

Another benefit of prediction is to support opportunistic communications, where, during intermittent connectivity or in the absence of an end-to-end network connection, a device uses context aware message forwarding to provide services [13, 14].

The ability to support context-awareness by predicting future locations is a small step towards fulfilling Mark Weiser's [15] vision of computers that are invisible, so embedded into daily life that they contribute to 'calm technology.' This vision of ubiquitous computing predicts that our devices will become our 'invisible assistants' [16], making our lives less stressful. Instead of *reactive* devices, on which the user has to stop and input a query for information, the goal is *proactive* systems that are integrated into our surroundings so that they can supply our needs in a timely fashion. In order to be proactive, the system needs to be able to predict our context in the near future, i.e. identifying what we will be doing and what we may need.

Our current location can be a good indicator of our current context and activities. Likewise, the ability to predict our future locations can be an initial step towards proactive context-aware systems. This goal of this research is to explore this initial step of predicting future locations, by using someone's location history to train a a predictive model.

The objectives of this research project are the following:

1. Use an inexpensive, universally available location system to
2. evaluate several prediction algorithms using real-world data,
3. while predicting both location and time.

If successful, the contributions of this research will advance the state-of-the-art in location prediction by adding temporal information. The addition of temporal information to location prediction should improve the predictions and allow us to predict users' future locations and the times that the users will arrive at their destinations.

(From [17])

"Prediction is very difficult, especially if it's about the future." – Niels Bohr

2 Related Work

There are several components necessary for predicting future locations. The first of these is a method for determining location. Section 2.1 describes the issues involved with location determination. Because positioning via IEEE 802.11 wireless access points is the only broadly available indoor location system, this chapter begins with a discussion of the benefits and pitfalls of using access points for positioning. Section 2.2 introduces prediction and describes current research in predicting mobile users next locations. The second component of this project is the selection of prediction algorithms. Section 2.3 concludes the chapter with a description of the prediction algorithms based on data compression that will be used in Chapters 3 and 4 to predict both future locations and time.

2.1 Location Determination

Previous research projects which required information about location have relied on various methods to determine location, including manual records [18], environments with built-in location-monitoring infrastructure [3], GPS [2], and association with 802.11 wireless access points. Each of these methods have their advantages and disadvantages. Manual records may be missing locations, but do emphasize key locations. Environments with built-in infrastructure are likely to have precise measurements of location, but are expensive to implement and are not universally available. GPS does not work indoors or in urban canyons and requires extra processing to translate a range of GPS readings into a single significant place. In addition, GPS data are not currently readily available.

This project will use 802.11 wireless access points as beacons for determining location [19, 20]. The advantages of this method are that, currently, 802.11 access points are ubiquitous on college campuses, providing a global location system. Many mobile devices, especially PDAs, include 802.11 radios, which makes the location system inexpensive. If the location information is kept on the user's device and not broadcast, privacy is supported. Currently, location by wireless access point is the only globally available solution that works indoors, where most of us work and live.

There are a few disadvantages with using 802.11 access points to determine location. To use an 802.11 access point as a reference point, its exact location must be known. Currently, most access points are installed without records of their exact locations. Corporate users, such as universities, record access points locations for maintenance purposes. If the use of Voice-over-Internet-Protocol (VoIP) on mobile devices continues to grow, E-911 regulations will require that the device can be located in case of an emergency. It seems a logical conclusion that future access points may be programmed to broadcast their locations.

Environmental factors can affect how strongly the access point signal is received by the mobile device. Because of these variations in received signal strength, location estimates using 802.11 wireless access points are not very precise, determining location within approximately 32 meters [21].

With the advent of location systems based on 802.11 wireless access points such as Place Lab [19], and collections of wireless usage data such as CRAWDAD [22] and UCSD[23], it is now feasible to apply prediction algorithms to larger groups of people over indoor and outdoor locations.

2.1.1 Symbolic Location

For location-based applications to be useful , geographical coordinates, such as 37.2302323948768,-80.42062044809619 (Torgersen Hall) need to be translated into *symbolic locations* [24] or *places* [25, 26], such as “Torgersen Hall”, “work”, “office” or “lab.”

For more precise systems such as GPS, this translation requires clustering groups of GPS readings to find significant locations and then asking the users to label these significant locations [2]. (This research is described in more detail in section 2.2.2.) Sensors that are placed on walls or ceilings, such as Cricket [27] or Active Badge [28] include this translation as part of the location system.

Some 802.11 wireless access points translate their “place” because they have been programmed with location information as part of their Service Set Identifier (SSID). For example, the IBM access point data available as part of the CRAWDAD research database [29] includes a Building number as part of the access point ID. My neighbor’s access point is programmed with his surname and his physical (postal) address is publicly available in the phone book.

The granularity of the location measurements returned by an 802.11 based positioning system is actually an advantage when translating positions into places because it removes the need to first cluster the lat/long readings and then translate them into a place.

2.2 Prediction

There are two questions to be answered when attempting to predict: *what* we are trying to predict and *how* we are going to try to predict it. In this project, I am attempting to predict both future locations and the times that someone may be at that locations. The following sections describe existing projects that predict either location or time but not both. Section 2.3 describes the *how*: the probabilistic prediction algorithms which will be considered.

2.2.1 MavHome

The MavHome ("Managing an Adaptive Versatile Home") at The University of Texas at Arlington is a smart home which seeks to 'maximize inhabitant comfort and minimize operation cost.' [3] The inside and surrounding area of the home is divided into zones (as shown in Figure 2-1.) in order to track the inhabitants' locations. The MavHome proposes to use location prediction in order to know which motion sensors to poll to find the inhabitant. The prediction serves two purposes, reducing the number of sensors which need to be polled and allowing longer time periods between location polls. In addition, the predictions can be used to allocate resources, such as adjusting the lights and temperature in rooms that are soon to be occupied [30].

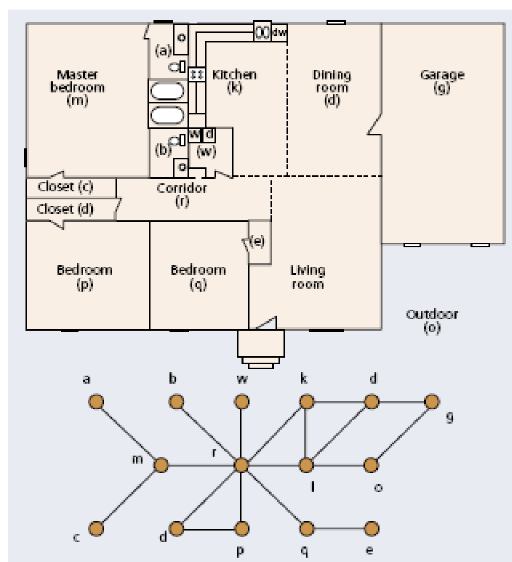


Figure 2-1 A graph model of a smart home floor plan (from [3])

The zones are modeled as a graph where the nodes are places (e.g. rooms) and the edges show connections between places, like hallways or doorways.

Prediction is done using the LeZi-update algorithm [31], an update scheme based on the dictionary-based LZ78 compression algorithm [32]. Movement history is stored as a string of zones, for example, *mamcmrkdkdgoog*. The LZ78 compression algorithm encodes variable length string segments using fixed length dictionary indices, updating the dictionary as new ‘phrases’ are seen. For example, the string of zones above would be parsed into the unique phrases *m, a, mc, mr, k, d, kd, g, o, og.* Common phrases represent common paths through the house. The phrases and their frequencies are stored in a tree and are used to calculate the probabilities of each phrase, given the movement history. Recent results [33] (2007) report prediction success rates of ~94% for a retired person, ~90% for an office employee and ~85% for a graduate student. The current implementation of this model predicts the next location and path.

Recent research by Jakkula [34] is looking into using temporal relationships between activities (such as using a “cooker”, an oven and a lamp in a MavHome) to improve predictions of the inhabitant’s next activity. With a synthetic dataset, they found a 7.81% improvement in predictions of the inhabitants next activity. For their small dataset of “real” activities, the addition of temporal information improved activity predictions by 1.86%, showing that adding temporal information can improve prediction accuracy.

2.2.2 Using GPS to Determine Significant Locations

Fig. 11 An illustration of the data reduction that occurs when creating *places* and *locations*. Picture a shows the complete set of data collected in Zürich for one user, around 200,000 data points. Picture b shows the roughly 100 *places* that were found using this data. In picture c, we see that this has been reduced to just 17 significant *locations*

Figure 2-2 Ashbrook & Starner's Technique to produce significant locations from GPS data. From [2]

In [2, 35], Daniel Ashbrook and Thad Starner analyze a large collection of GPS data in order to predict users' next 'significant locations.' Initially, GPS data are collected for a mobile user. The data are pared down into *places* by keeping only the data where the user stopped for more than 10 minutes or lost a GPS signal (entered a building). Because few GPS readings in the same significant location will exactly match, an iterative clustering algorithm is used to collect readings from the same general location to produce a set of *significant locations*.

An example of this process is shown in Figure 2-2. Of the five users studied, three had 11 significant locations, one had 9 and one had 6.

Later, users were asked to label the locations tagged as 'significant' to see if the algorithm was truly capturing locations that were important to the user. At most, users had two locations which they were unsure about (which may be due to the fact that the users were asked for the location names several months after they had left Zurich, the city in the study.)

A second-order Markov model was used to predict the user's next location. Exact results were not reported, but the results achieved by the Markov model were significantly higher than those predicted by random chance using a Monte Carlo simulation.

The results of Ashbrook and Starner's work provide assurance that mobility data can be used to derive the places that are important to people (their 'significant locations') and that prediction can be successfully applied. Other approaches to determining 'places' from latitude and longitude are summarized in [25, 36].

2.2.3 Dartmouth College Mobility Predictions

Researchers at Dartmouth College applied several prediction algorithms to extensive mobility traces of over 6,000 users collected over a two-year period [37]. As in the MavHome, movement history is stored as a string. In this case, each letter in the movement string represents the IEEE 802.11 access point with which the mobile user is associated. The string includes location *changes* only and no time information is recorded. Several predictors were considered, including Markov predictors and LZ-based predictors. They found that the simple low-order Markov predictors worked as well or better than the other predictors, including higher order Markov models, which confirms that recent history is shown to be a better predictor than the probabilities determined over long historical traces. When a predictor failed to make a prediction due to encountering a history it had never seen before, a fallback procedure was implemented which allowed the predictor to use shorter and shorter context strings until it could make a prediction. This approach is similar to the Prediction-by-Partial-Match (PPM) algorithm, which is described in section 2.3.1. This fallback procedure improved the accuracy, resulting in a total accuracy of 65 -72% for the second-order Markov predictor with fallback.

This project is the only large scale project using IEEE 802.11 positioning of a large number of users. The result of 65-72% prediction is my baseline. My goal is to see if adding temporal information to the Markov and LZ-based models will improve the predictions and allow us to ask questions about the users' locations farther out into the future.

2.2.4 Smart Office Buildings

Jan Petzold, in his PhD research [18, 38], applied several machine learning algorithms to predicting an office worker's location. Four office workers manually tracked their location for a summer. A Smart doorplate was developed and attached to their office doors and when a visitor came and found the office empty, the smart doorplate predicted the location of the absent office worker. Five machine learning algorithms were applied to the data: two neural networks (Elman Net and Multi-layer Perceptron), a Bayesian network [39], a State predictor [40] and a Markov predictor. Prediction accuracies ranged from 68% to 91%. However, none of the algorithms was optimal for all four subjects. In other words, there was no universal predictor.

Petzold also applied confidence estimation techniques (the strong state method, the threshold method and the confidence counter) [41] to the State Predictor method in order to decide if it is better to make no prediction rather than one that has a high probability of being incorrect.

2.2.5 Predicting Future Times of Availability

The goal of Computer Supported Collaborative Work (CSCW) research is to enable groups of people to work together more effectively. CSCW includes collaboration software (e.g. NetMeeting), group calendars and systems to support awareness of remote coworkers. One important aspect of working together is knowledge of other's *presence*. We use presence, or knowledge of coworkers patterns of availability and locations, to know when to initiate a conversation or schedule a meeting. A special case is *ambushing*, which "is the practice of waiting for a colleague to become present at a particular location in the interest of meeting with him. The location is known in advance, and the time at which the colleague will appear is typically not." [42]

The Rhythm Awareness project [43], developed by Bo Begole (a Virginia Tech graduate), Tang and Hill, modeled and predicted users' online presence in order to predict the accessability of a coworker. The predictions augmented an Instant Messaging contact list. Figure 2-3 shows an example where:

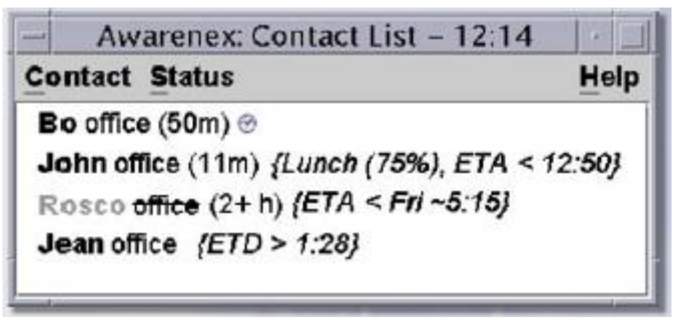


Figure 2-3 Augmenting an Instant Messenger buddy list with rhythm inferences. (From [44])

- Bo has been inactive for 50 minutes, which did not correspond to a known pattern, so no prediction was made,
- there is a 75% probability that John is out to lunch and predicted to return before 12:50 PST, and
- Rosco, a worker on the East Coast, has logged out for the day and is predicted to return the next morning, and
- Jean, also on the East Coast, is still in the office but is predicted to leave around 1:28 PST.

The beginning of this chapter described several research projects where a mobile users' next location was predicted. Other than the Dartmouth study, most of these projects use a small number of subjects and a small number of locations (~10) to predict only the users' next location. Begole does not predict location directly, but predicts the time someone will be online, which implies that they are in their office.

The following section describes the three data compression algorithms that will be considered for this project, where I am trying to predict both time and future location.

2.3 Prediction Based on Text Compression

Effective text compression algorithms rely on predicting the next character given the preceding characters. Because of this ability to predict the next text character, good text compression algorithms also make good predictors for sequential data. The MavHome and Dartmouth studies mentioned previously store locations visited as characters in a text string and then use compression algorithms to predict a mobile user's next location. When a sequence of locations is stored as a character string, then predicting the next location is the same problem as predicting the next character in a string. Compression algorithms have also been used for branch prediction in microprocessors [45], file and cache prefetching[46] and predicting Web pages accessed [47].

The problem with using compression algorithms to prediction location and time is that compression algorithms use only scalar variables, such as text characters or locations.

In their experiments comparing variable-order Markov models, Begleiter et al. [48] applied six algorithms to the prediction of musical selections. Musical notation has multiple variables: notes, their starting times and their durations. These multiple variables were coded as single-variable character strings and the single-variable prediction algorithms were able to recognize the patterns in the music. The specification of music correlates with the specification of a user's path throughout the day, which consists of locations, starting times and durations. **It is my hypothesis that temporal and place information can be embedded in a string of single**

variables which will be used to build a model based on data compression to predict both location and time.

In Begleiter's experiments, prediction performance was measured using the *average log-loss*. Given an alphabet Σ, an algorithm is trained on a sequence $q_1^n = q_1 q_2 q_3 \cdots q_n$, where $q_i \in \Sigma$. Given a test sequence $x_0^T = x_0 x_2 \cdots x_T$, the average log-loss $l(\hat{P}, x_0^T)$ is given by

$$l(\hat{P}, x_0^T) = -\frac{1}{T}\left[\sum_{i=1}^{T} \log_2 \hat{P}(x_i \mid x_0 x_1 \cdots x_{i-1}) + \log_2 \hat{P}(x_0)\right] \tag{2-1}$$

The log-loss equation relates to the number of bits required to compress a test string, given that the algorithms was trained on another string. The average log-loss measures the average compression rate of the test string. In other words, the smaller the log-loss, the fewer bits are required to compress the string. Minimizing the average log-loss corresponds to maximizing the probability assignment for the test sequence. In other words, the smaller the average log-loss, the more predictable the test string. For example, if the model was trained on the string 'abracadabra', the average-log-loss for the test string 'abr' would be less than the average-log-loss for the test string 'car.'

Inspection of the average-log-loss equation reveals a potential problem with probabilistic models. If a the test string contains a sequence that has not been encountered before, such as the test string 'car' after training on 'abracadabra', the probability for 'car' is 0, which results in an infinite value for average-log-loss. This is known as the zero frequency problem. Each prediction model includes a smoothing component which prevents unobserved strings from producing a probability of zero.

Begleiter et al. found that the Prediction by Partial Match (PPM-C) and Context Tree Weighting (CTW) algorithms resulted in the lowest average log-loss when tested on MIDI (music) files. The Lempel-Ziv compression algorithm performed the worst, but will be included in my experiments because of its success in MavHome [3, 49]. Because of these results, in this project, I will focus on these three compression algorithms, PPM-C, CTW, and LZ78.

In the following sections, I will explain each of the these compression algorithms. An example of how the sequence *abracadabra* would be used to train the model and make predictions is also illustrated for each model.

2.3.1 Prediction by Partial Match

The Prediction by Partial Match (PPM) algorithm uses various lengths of previous contexts to build the predictive model [50]. As a training string is processed character-by-character, a table is built for each sub-string and the characters that follow it including a count of the number of times that character has been seen occurring after that sub-string. For example, if the training string is *abracadabra*, the training begins by building an entry for *a* with count 1 in the order-0 table. It then adds an entry for *b* to the order-0 table with a count of 1, and begins the order-1 table, by creating a table for 'characters which follow *a*' with an entry labeled *b* with a count of 1. When the training is over, an ESCAPE character is appended to each table. This character is used during encoding to mark situations where novel characters are seen. In the 'Method C' variation of PPM (called PPM-C), the ESCAPE character is given a count equal to the sum of the number of different symbols that have been seen in that context. Figure 2-4 below shows the PPM-C model after training on the string *abracadabra*, with a maximum order of 2. The values under the heading '*c*' are the number of times that character has been seen following the given context. The '*p*' column is the probability, which is the count for the given character divided by the sum of all the counts in that sub-table.

To use the PPM-C model for prediction, the tables are traversed given the context. For example, if the context given is *ab*, the 'Order k=2' table is searched first to see if there is an entry for *ab*. Since there is an entry for *ab*, the prediction engine simply reports the character(s) with the highest probability, in this case *r*, which is reported to have a probability of 2/3.

Order $k = 2$			Order $k = 1$			Order $k = 0$			Order $k = -1$		
Predictions	c	p	Predictions	c	p	Predictions	c	p	Predictions	c	p
ab $\rightarrow$ r	2	$\frac{2}{3}$	a $\rightarrow$ b	2	$\frac{2}{7}$	$\rightarrow$ a	5	$\frac{5}{16}$	$\rightarrow$ A	1	$1/\|A\|$
$\rightarrow$ Esc	1	$\frac{1}{3}$	$\rightarrow$ c	1	$\frac{1}{7}$	$\rightarrow$ b	2	$\frac{2}{16}$			
			$\rightarrow$ d	1	$\frac{1}{7}$	$\rightarrow$ c	1	$\frac{1}{16}$			
ac $\rightarrow$ a	1	$\frac{1}{2}$	$\rightarrow$ Esc	3	$\frac{3}{7}$	$\rightarrow$ d	1	$\frac{1}{16}$			
$\rightarrow$ Esc	1	$\frac{1}{2}$				$\rightarrow$ r	2	$\frac{2}{16}$			
			b $\rightarrow$ r	2	$\frac{2}{3}$	$\rightarrow$ Esc	5	$\frac{5}{16}$			
ad $\rightarrow$ a	1	$\frac{1}{2}$	$\rightarrow$ Esc	1	$\frac{1}{3}$						
$\rightarrow$ Esc	1	$\frac{1}{2}$									
			c $\rightarrow$ a	1	$\frac{1}{2}$						
br $\rightarrow$ a	2	$\frac{2}{3}$	$\rightarrow$ Esc	1	$\frac{1}{2}$						
$\rightarrow$ Esc	1	$\frac{1}{3}$									
			d $\rightarrow$ a	1	$\frac{1}{2}$						
ca $\rightarrow$ d	1	$\frac{1}{2}$	$\rightarrow$ Esc	1	$\frac{1}{2}$						
$\rightarrow$ Esc	1	$\frac{1}{2}$									
			r $\rightarrow$ a	2	$\frac{1}{3}$						
da $\rightarrow$ b	1	$\frac{1}{2}$	$\rightarrow$ Esc	1	$\frac{1}{3}$						
$\rightarrow$ Esc	1	$\frac{1}{2}$									
ra $\rightarrow$ c	1	$\frac{1}{2}$									
$\rightarrow$ Esc	1	$\frac{1}{2}$									

Figure 2-4 PPM-C Model after training on the string abracadabra, from [38]. The column, c, is the count of the number of times that character (or set of characters) occurred in the training string. The column, p, is the probability of that character occurring in the given context. Escape characters are returned to tell the model to drop to a lower order.

If the given context is not found in the table, the model shortens the context until it finds an entry in the table. For example, if the given context is *ba*, the model first looks for a *ba* sub-table in the second-order (k=2) table. If it's not found, it shortens the context to *a*, and looks for a table for *a* in the first-order (k=1) table. Finding that entry, it reports that the most likely next character is *b*, with a probability of 2/7. If the context is not found in any of the higher order tables, the model falls back to the k=0 table, which simply reports the most commonly occurring character. The k=-1 table is used for characters that have not been seen during training, and does not apply to prediction, since a prediction algorithm will not predict states (or characters) it has never seen before. Arithmetic coding [51] is used to build the tables and to calculate the probabilities.

2.3.2 Context Tree Weighting

The Context-Tree Weighting (CTW) method [52] was developed to compress binary strings. It builds a model consisting of a binary tree where each node corresponds to a substring of the context. It then calculates the probabilities for each leaf of the tree based on the number of times zeroes and ones follow the substring. The probabilities for the internal nodes of the tree are calculated recursively from the leaves to the roots. As bits from the binary source are added to the tree, the appropriate counters are incremented and the probabilities are updated.

Decomposed CTW (DE-CTW) was introduced by Paul Volf [53] in his doctoral thesis as a method to extend the binary CTW method for larger alphabets such as ASCII characters. The DE-CTW method divides the multi-valued alphabet into a series of binary problems. For example, for the string *abracadabra*, the alphabet is the set

$$\Sigma = \{a, b, c, d, r\} \quad \text{with size } k = |\Sigma| = 5.$$

A tree is constructed where the root is the entire string, *abracadabra*. The five leaves from the root solve the binary problem for each symbol in the alphabet, such as "Is the next symbol the character *a* or in the set $\{b, c, d, r\}$? The leaves from the $\{b, c, d, r\}$ node solve the problem "Is the next symbol the character b or in the set $\{c, d, r\}$? The tree continues growing until each leaf corresponds to a single symbol. Once the tree is built, the probabilities for the leaves are calculated, which correspond to the probabilities for each symbol given the context of the leaf.

2.3.3 Lempel-Ziv (LZ78)

The LZ78 compression algorithm [32] is a popular, dictionary-based algorithm, which parses the incoming sequence of characters into a dictionary of unique phrases. The initial dictionary is the empty set, and as each character is encountered, a phrase is added with consists of a phrases already encountered with the new character appended. For example, for the string *abracadabra*, the dictionary consists of $\{\mathrm{a}, \mathrm{b}, \mathrm{r}, \mathrm{ac}, \mathrm{ad}, \mathrm{ab}, \mathrm{ra}\}$.

The prediction portion of the algorithm was developed by Langdon [54] and Rissanen [55]. The tree that contains the phrases in the dictionary is extended to include counters that are used to calculate probabilities. The root of the tree is the empty string. As the tree grows, a leaf is added for every symbol in the alphabet, so that there is a non-zero probability even for symbols that are not in the training string, but which may be encountered in future strings (the zero-frequency problem). The phrases in the dictionary are then used to update the counters. An example of the resulting tree for the training string *abracadabra* is shown in Figure 2-5.

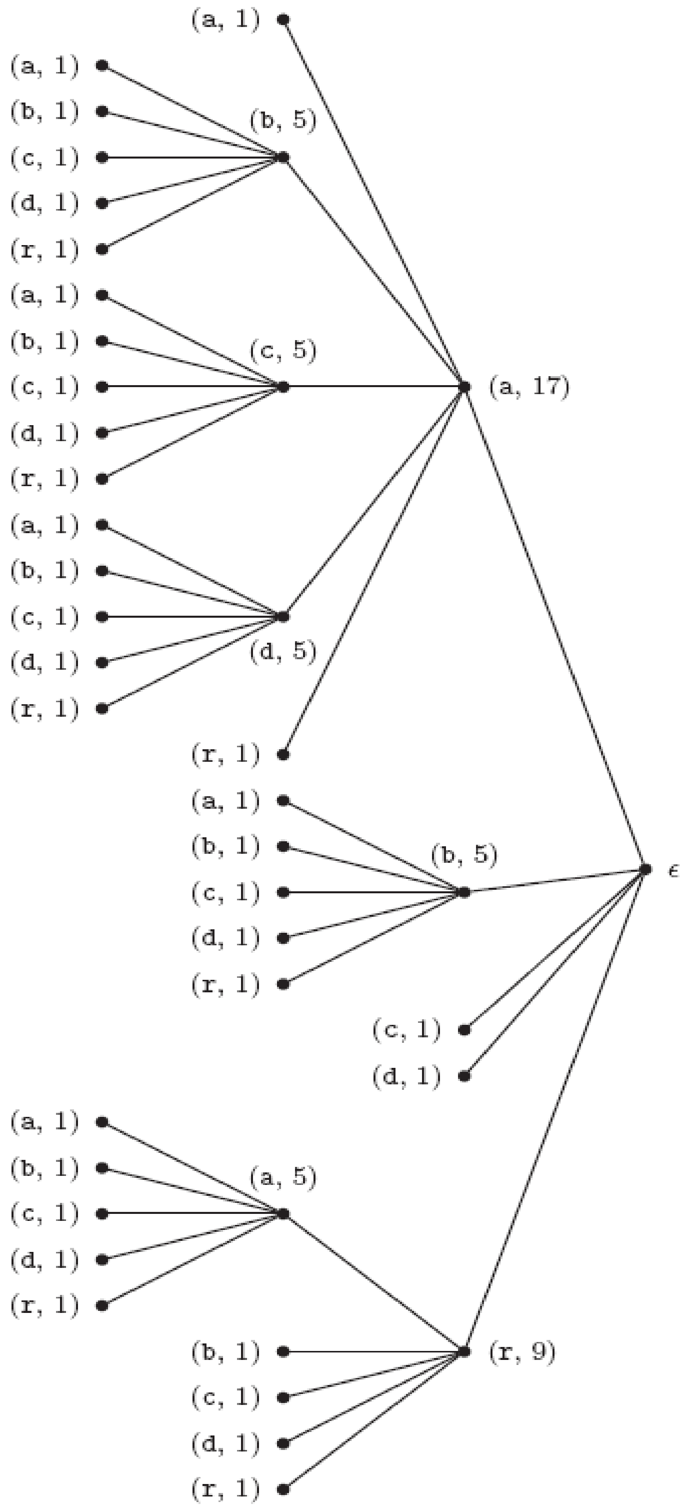
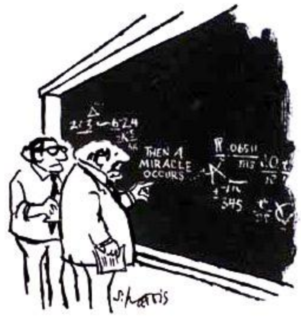


Figure 2-5 Tree produced using LZ78 algorithm on the string *abracadabra*. From [48]

Probabilities are calculated by traversing the tree. For example, in the tree above, the calculation for $\hat{P}(c \mid a)$ is done by traversing the tree $\varepsilon \rightarrow a \rightarrow c$. The probability is the count at the leaf (5) divided by the count at its parent (17) or 5/17. Likewise, $\hat{P}(r \mid a) = 1/17$, which relates to the fact that the sub-string ‘ac’ occurs in the training string but the sub-string ‘ar’ does not.



"I THINK YOU SHOULD BE MORE EXPLICIT HERE IN STEP TWO."

3 Research Plan

This section outlines the plans for this research project, including goals, objectives, methodology, applications and plans for evaluation.

3.1 Goals and Objectives

The goal of this project is to predict a mobile user's future locations and time. This furthers the state-of-the-art, which predicts only the *next* location. The ability to predict future activities enables collaborative opportunities, such as serendipitous meetings or delivery of items.

Within the overall goal of predicting future locations, this project has several objectives:

- Determine users' locations by processing IEEE 802.11wireless networking data to extract important position information,
- Create an accurate and efficient model for determining future locations and times,

- Use the model to provide useful forecasts, such as a prediction when someone will be at a given location, within a given timeframe, along with a measure of confidence in the prediction.

Figure 3-1 shows a simplified block diagram of the process. Each of these steps will be discussed in detail in the following sections.

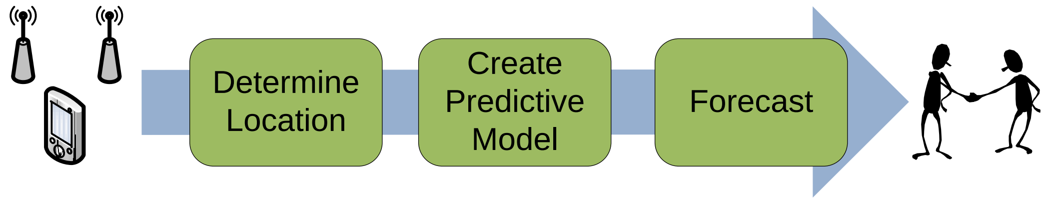


Figure 3-1 Steps to predicting future locations

3.2 Location Determination

The first step in the prediction process is determining and recording past locations in order to predict future locations. Any positioning system can be used to determine location as long as it supplies enough information to determine the users' symbolic location and includes the temporal information such as when the user arrived at that location and how long he stayed. Any positioning system selected would require filtering to remove noise, data mining to extract significant locations and translation to symbolic locations (places). While the implementation of these steps may be different for each location system selected, the general result is that any location system could be used for prediction.

In this project, location will be determined using nearby 802.11 wireless access points as location beacons. The advantages of this technique are that wireless access points are pervasive throughout a campus environment and can be sensed both indoors and outdoors. The location is calculated on the user's device, which supports the user's privacy.

However, there are several difficulties with this technique:

- Location of the access point beacons may not be known,
- A stationary user could be 'ping-ponging' [56] associations between two access points, which could be misinterpreted as movement, and
- Wireless data logs create large volumes of data, which must be mined for critical features.

Each of these issues will be examined in the following sections.

3.2.1 802.11 Access Points with Unknown Locations

The first problem in using wireless access points as beacons to determine location is that the locations of the access points may not be known. In this project, I assume that I know the location of most of the access points in the study. Currently, in the real world, the location of most access points is unknown, but with the growth in popularity of Voice over Internet Protocol (VoIP) and E911 laws, I am assuming that future installations of 802.11 networks will include access points with their location stored in the device and a method to broadcast that location. In addition, proposed standards such as IEEE 802.11k [57] for radio resource management include provisions for access points to broadcast their location.

There are several ways to deal with unknown access point beacons. One method is to monitor other access points that the wireless radio is sensing (but not associating with) at the same moment and either use one of those access points as the location beacon, or trilaterate between their positions to estimate the location of the unknown access point. Another method is to label this AP's location as UNKNOWN, and use that in the model to indicate when someone is in an unknown location. In one aspect, this option promotes privacy because a user could mark certain access points as 'Location Unknown' and thereby exclude those locations from the predictive model. If a corporate user of this application wished to preserve privacy, they could mark all off-campus access points as ‘location unknown’ and provide assurance that the predictive model was only being used for work purposes.

3.2.1.1 Determining Location from Access Points

Even though access points can have a relatively large range of 100 meters indoors [21], it is assumed that each mobile device associates with the physically closest access point, and therefore, the associated access point can be a satisfactory broad indication of location. In reality, however, devices do not always associate with the closest access point. Mobile devices monitor received signal strength and may re-associate with different access points when signals change [58]. In addition, proposed radio resource management protocols such as IEEE 802.11k may recommend that a device associates with a more distant access point in order to evenly distribute loads across multiple access points.

In this project, the associated access point will be considered the user’s location. For example, if the mobile device is associated with an access point in the north half of Torgersen Hall, the user will be considered to be in the north half of Torgersen Hall. If the associated access point’s location is unknown, the next sensed access point will be considered the user’s location. If all of the sensed access points have unknown locations, the user’s location will be marked UNKNOWN.

3.2.2 Ping-pong between access points

The wireless radios in mobile devices can use different criteria to determine when to disassociate with the current access point and associate with a different access point [58]. They may switch access points when the received signal strength from another access point is stronger than the current access point. They may have a threshold value for received signal strength, where the wireless device stays associated with its current access point until the received signal strength falls below the threshold.

Environmental factors can influence received signal strength, including building materials, the weather and even people passing in the hallways. Therefore, received signal strengths can fluctuate even when a mobile device is completely stationary. In the worst case, a stationary device can cycle its associations between two or more nearby access points as received signal strengths fluctuate. This phenomenon is called *ping-ponging* [56], as the device's association ping-pongs between different access points.

As described in detail in the next chapter, it was found that over *one-third* of the transitions between access points in the UCSD dataset were ping-pong events, so the incidence of ping-ponging is significant enough to not be ignored. One contribution of this research is to identify ping-pong events between two or more access points, determine the extent of ping-pong effects, and determine how to merge ping-pong records into the data.

3.2.3 What Will Be Done for Location Determination

The UCSD dataset used in this research consists of polls recorded every 20 seconds of 275 PDAs over an 11-week period, resulting in a file with more than 13 million records. Because of the large amount of data, the first step is to reduce the data to retain the useful features and dispose of noise. This phase of the project draws from Data Mining, where the first step is to format the raw data and then to reduce the data to extract useful cases.

The following steps will need to be performed on the UCSD dataset:

1. The large file needs to be broken into separate files for each user.

2. Records where the device sensed an access point but did not associate will be dropped. If the received signal strength is inadequate for association, it is considered inadequate for positioning.
3. Continuous records will be merged into *sessions*, where each session consists of a starting time, the associated access point, and the duration of the session.
4. Time gaps between records will be analyzed. Gaps in time could indicate diverse events. The wireless radio of the device could have been shut off, could have moved out of range of any access points or could have experienced a short-term difficulty associating with an access point. Large time gaps between records might indicate an end to a user's movement and the beginning of a new pattern, such as the beginning of a new day at work. Short gaps between records could be considered inconsequential and these records could be combined with previous or subsequent records. A suitable threshold value needs to be determined to decide how to treat the time gaps between sessions. Tests will be run to combine sessions with time gaps under 1, 2, 5, and 10 minutes and compare their prediction rates.
5. Ping-pong sessions will be identified and processed, usually by merging with neighboring sessions.
6. Access point locations will be translated into symbolic locations, including unknown locations.

Once this data mining process is completed, the result is a dataset that contains symbolic location data ("places") along with the related time information. Any positioning system which can provide this information can then be used for prediction. In this project, location is determined by using IEEE 802.11 wireless access points with known locations as reference beacons. Symbolic location is determined by looking up the access point in a table of access point locations.

3.3 Developing the Predictive Model

In the previous section (3.2) the problems and procedures for determining location were explained. The next step is to develop the predictive model. Several algorithms will be developed and compared. As discussed in Section 2.3, data compression models operate on single variables. A representation needs to be designed that will embed both temporal and positional information in a string of single variables.

3.3.1 Characteristics of a Successful Algorithm

The algorithm, or model, used to predict future locations needs to have several characteristics:

1. It can predict with an acceptable level of accuracy.
2. It provides a measurement of confidence in the forecast.
3. It is able to model time as well as locations.
4. It can handle many locations and times.
5. It is efficient in resources, not requiring large amounts of memory or processing time.
6. It can periodically be flushed, to handle changes in schedules (e.g. semester breaks).
7. It is adaptable to support different types of forecasts (discussed in Section 3.4).

3.3.2 Representing the Information to be Predicted

In addition to selecting the best predictive model, the best *representation* of the location and time information needs to be determined. Recall that the location prediction models treat location information as a series of text characters in a string. Time information is added as additional characters in the string. For example, assume the following translations of location and time information into characters:

Table 3-1 Sample translation of location and time information into characters

Character	Place	Character	Time	Character	Duration
A	Torgersen Hall	1	9:00 am	a	10 minutes
B	McBryde Hall	2	9:10 am	b	20 minutes
…		3	9:20 am	c	30 minutes
		4	9:30 am	d	40 minutes
		5	9:40 am	e	50 minutes
		6	9:50 am	f	1 hour
		7	10:00 am		
X	UNKNOWN	8	10:10 am		

Our user, Bob, went to Torgersen Hall at 9:00 am, stayed there for 20 minutes, then disconnected and appeared in McBryde Hall at 10:00 am for 10 minutes. There are several different ways to represent this sequence of events. The sequence "9:00, Torgersen, 9:10, Torgersen, 9:20, Torgersen, 10:00, McBryde, 10:10, McBryde" translates to *1A2A3A7B8B* or *1A2A3A4X5X6X7B8B*. His actions could also be expressed as "9:00, Torgersen, 30 minutes, 10:00, McBryde, 10 minutes", which translates to *1Ac7Ba* or *A1cB7a* or *1Ac4Xc7Ba*. The time information could be implicitly included by repeating his location for each 10-minute time slot: *AAAXXXBB*. In Section 4.3.1, I apply different representations to the PPM-C model and produce different prediction results.

Not only does the representation need to be considered, but the information included in the representation also needs to be evaluated. Some of the example sequences above do not explicitly include the duration of Bob's stay. None of the examples include the day of the week, but simply the time of day. It could be that adding day-of-the-week information significantly improves predictions. Then again, adding additional information may simply needlessly complicate the model.

3.3.3 What will be done to develop the Model

As mentioned in Chapter 2, several studies have successfully used the Lempel-Ziv model to predict the next location. However, when Begleiter [48] compared several Markov models along with Lempel-Ziv, he found that Lempel-Ziv was the worst performer for predicting musical selections. His results show that two variable-order Markov models, Prediction by Partial Match (PPM) and Context Tree Weighting (CTW) were the best for predicting musical selections.

Musical selections have multiple variables: notes, the time the notes start and the length each note is held. Since I am also working with multiple variables (locations and times), I will evaluate both the PPM and CTW models along with Lempel-Ziv compression. For each of the three models, I will do the following:

1. Implement the model in C. Because C is a compiled language, it will run faster than Java, and it is easier to measure the amount of memory consumed by the tables and trees built by each of the models.
2. Expand the basic model to support many locations and times. Algorithms based on text-compression, as all three of these are, are restricted to 7-bit characters, limiting their use to 128 different locations and times, which is insufficient to represent many times during the day and several locations. The models will be expanded to use 16-bit symbols.
3. Expand the model to support different representations in order to determine which representation supports the best predictions.
4. Measure the amount of memory used by each model and the processing time, in order to compare resource usage between models.

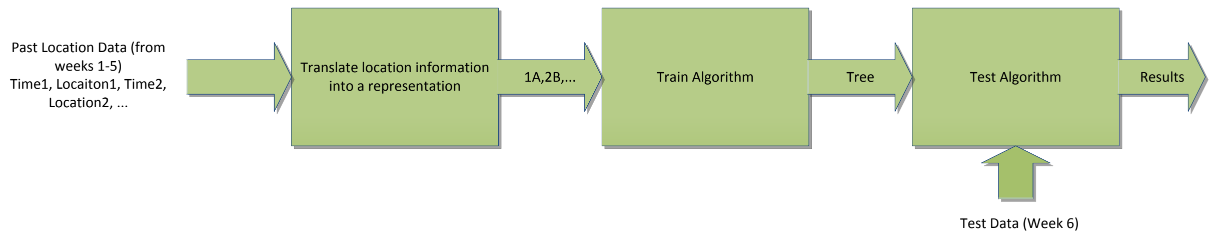


Figure 3-2 Prediction Algorithm

Each of the models will be implemented and trained on the same datasets and the same representations. They will then be tested by predicting a user's locations against a test dataset as shown in Figure 3-2. For example, the first five weeks of the USCD location data will be used for training and then the model will be asked to predict the users' locations and times for the sixth week. The models will then be compared to see:

1. Which is the most accurate, meaning which had the most correct predictions?
2. If the model produces a confidence measure, how often does a high measure of confidence correlate with a correct prediction?
3. Which algorithm runs the fastest?
4. Which algorithm uses the least memory?
5. Which algorithm can best answer the different types of questions required for forecasting? For example, the question, "Where will Bob be next?" is straightforward for all three algorithms, because it basically involves predicting the next character in a string, given some context. But a question such as "When will Bob be in Torgersen?" may involve traversing tables and trees in the opposite direction.

3.4 Forecasting

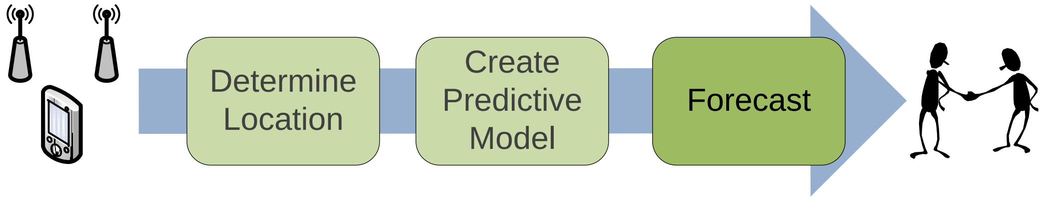


Figure 3-3 The final step is the forecast and answering a query.

In the second phase of the research, a predictive model will be developed and trained on historical location data. Following that, the third phase of this project will apply the model to make actual forecasts. Given some context, a forecast predicts a future location with a time and a measure of confidence in the prediction. There are three types of questions which need to be supported:

1. Where will Bob be next?
2. Where will Bob be at time *x*?
3. When will Bob be at location *y*?

Forecasting also involves translating back from the representations used by the model into real locations and times.

In addition to fulfilling queries, the forecasting portion of the project will attempt to qualify the predictions. If a query results in no prediction or a prediction with low confidence, the forecasting engine will need to introduce fuzziness into the query in an attempt to improve the prediction (Figure 3-4). For example, the predictive model is given a sub-sequence of context, such as the user's five preceding locations, and is asked to predict the next location. The query could be "Bob has been to locations A, B, C, and D; where will he be next?" If Bob always takes the same route, the prediction engine will predict his next location, F, with a high level of confidence. But what if Bob varies his route? The forecasting engine can try different contexts, such as “A,B,D” or “A,C,D” to see if the prediction improves. Locations where the user spends more than 10 minutes are considered significant locations [36] so the forecasting engine could drop intermediary locations with short durations from the query.

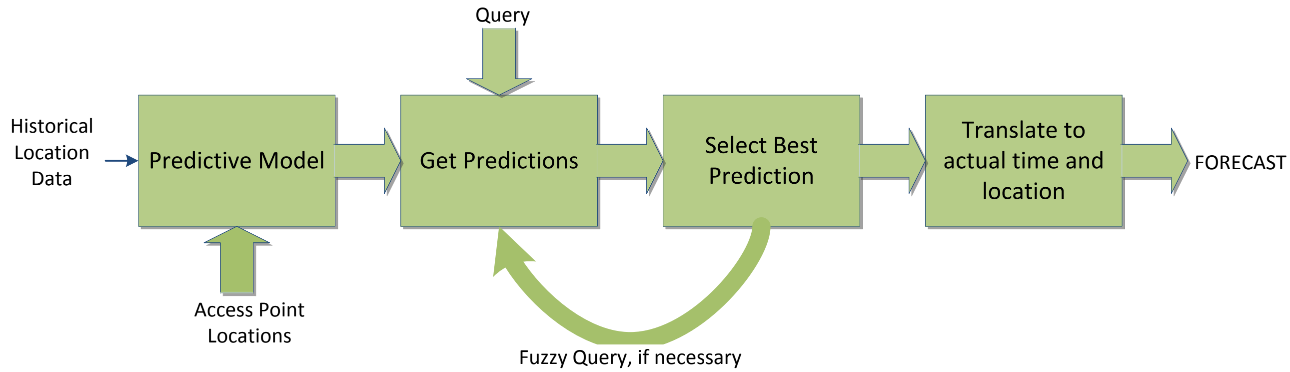


Figure 3-4 Forecasting Engine

3.4.1 What will be done to Forecast

To determine the qualities for the forecasts for different types of queries, the following tests will be run.

1. For each location in the test dataset, given a time and the preceding locations, determine the user’s next location.
2. For each time of day, predict the user’s location or whether the location is unknown.
3. For each location in the test dataset, predict *when* those locations will be visited.

4 Preliminary Research and Results

As explained in Chapter 3, there are three major steps to this project, determining location, selecting and training the predictive model, and using the model to forecast. In this chapter, I will discuss the preliminary research that has been done in the first two areas, location determination and the predictive modeling.

4.1 Determining Location from IEEE 802.11 Wireless Data

As mentioned in the previous chapter, the initial analysis of the large UCSD dataset involves preprocessing the data and mining them for useful features. The 13 million record file needed to be divided into manageable files, one for each user, and then processed and translated to produce useful information. This 'feature extraction' process is described in the following section. It includes removing records where the device was not associated with an access point or associated for less than 30 seconds and identifying and removing ping-pong events.

4.1.1 Extracting Information from the UCSD Raw Data

As part of the Wireless Topology Discovery (WTD) project at UCSD, researchers issued PDAs to 300 freshmen and collected movement traces [23]. The resulting dataset consists of two files: one file containing the access point locations and one file with trace data. The data was collected during an 11 week trace period from 9/22/2002 to 12/8/2002.

While a student's device was powered on, WTD sampled and recorded the following information every 20 seconds for all access points (APs) that it could sense across all frequencies--not just the access point the wireless card was associated with at the time:

USER_ID:	Unique identifier assigned to the user
SAMPLE_TIME:	The time the sample was taken by the WTD software
AP_ID:	Unique identifier assigned to the detected AP
SIG_STRENGTH:	Strength of AP signals received by device
AC_POWER:	Whether the device used AC power (1) or battery (0)
ASSOCIATED:	Whether the device was associated with this AP (1) or not (0)

Since WTD recorded the above information for all APs sensed during a sample, if a user device saw three APs in one sample, there were three entries recorded for that sample (which differ only in the AP detected, signal strength, and associated flag).

The original dataset has the following format:

```
USER_ID,SAMPLE_TIME,AP_ID,SIG_STRENGTH,AC_POWER,ASSOCIATED
123,09-22,00:00:00,359,8,0,0
123,09-22,00:00:00,363,5,0,0
123,09-22,00:00:00,365,11,0,1
191,09-22,00:00:00,355,31,0,1
101,09-22,00:00:00,353,8,1,0
101,09-22,00:00:00,362,30,1,1
129,09-22,00:00:00,369,31,0,1
156,09-22,00:00:00,360,19,1,1
184,09-22,00:00:02,352,29,0,1
```

Figure 4-1 A portion of the UCSD dataset

To be able to practically work with such a large amount of data (over 371 Megabytes), the first step towards data reduction was to parse the large file into separate files, one for each of the 275 users. Further reduction was achieved by using only the records where the device was associated with an access point, dropping the records where the device sensed an access point but did not associate. Continuous records were combined to record *sessions*, with a starting time, the associated access point and the duration of the session.

A sample of the resulting file for user 23 is shown below:

User	Start Time of Session	Associated Access Point	Duration of Session
23	2002-09-22 04:01:03.00	357	01:36:39.00
23	2002-09-22 05:41:15.00	357	00:00:44.00
23	2002-09-22 12:21:11.00	357	00:00:43.00
23	2002-09-22 14:01:02.00	36	00:07:31.00
23	2002-09-22 14:58:18.00	357	00:02:52.00
23	2002-09-22 15:01:32.00	354	00:29:12.00
23	2002-09-23 18:54:49.00	354	00:00:22.00
23	2002-09-23 18:57:42.00	356	00:07:31.00
23	2002-09-23 19:05:34.00	354	00:00:22.00
23	2002-09-23 19:06:18.00	356	00:04:19.00

Figure 4-2 Sample of records for User #23

To confirm that this data represents user mobility, I graphed the number of access point changes in 15-minute intervals, as shown below. The graph reveals a weekly pattern, with users being most mobile (or connected) during the daytime.

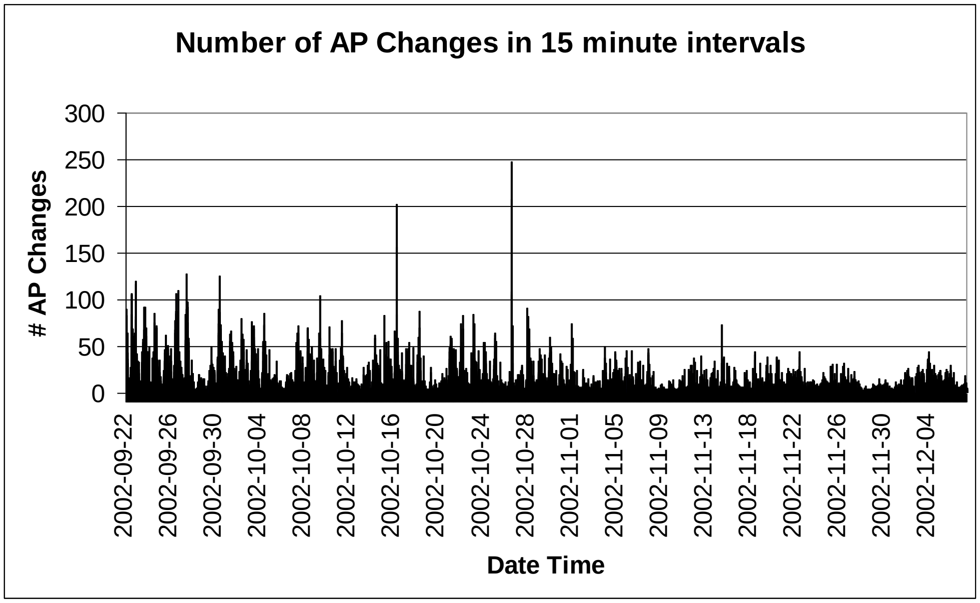


Figure 4-3 Number of Access Point changes by all users in 15-minute intervals over 11 weeks

When the data are counted by the day of the week and time of day, as shown in Figure 4-4 above, the graph shows that users are more mobile in the middle of the day and on weekdays.

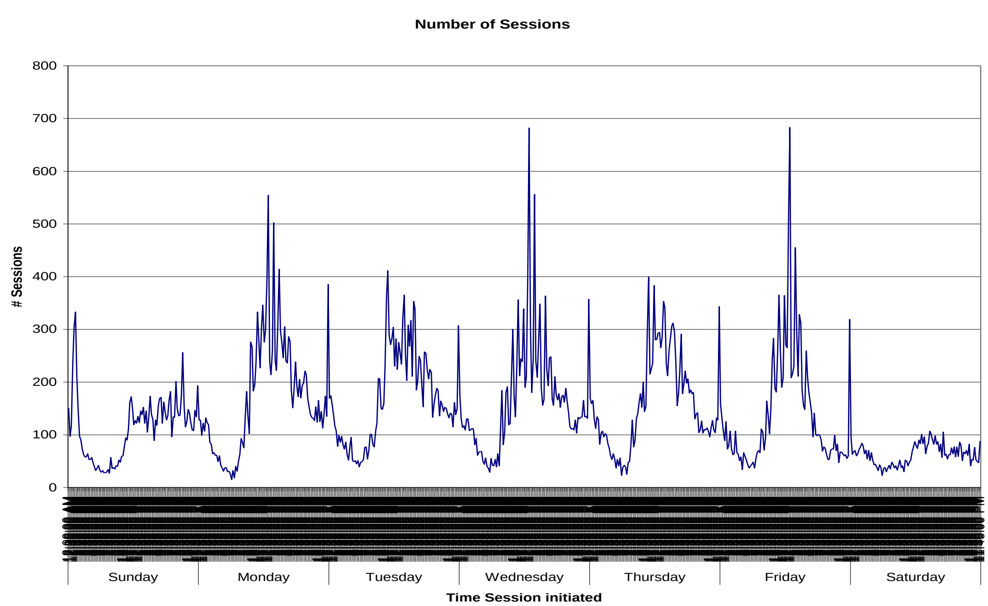


Figure 4-4 Number of AP changes by Day-of-Week

4.1.2 The Ping-Pong Effect

When investigating whether the number of ping-pong events was significant, I initially defined a ping-pong event as a set of sessions where a device switched from one AP to another AP and back to the original AP, where each of the sessions began within 30 seconds of the end of the previous session. Two of the ping-pong events for user 23 are indicated by a green background below:

User	Start Time of Session	Associated Access Point	Duration of Session
23	2002-09-26 01:20:07.00	357	00:05:08.00
23	2002-09-26 01:25:36.00	363	00:00:00.00
23	2002-09-26 01:25:56.00	357	00:06:27.00
23	2002-09-26 01:36:50.00	357	00:02:21.00
23	2002-09-26 01:39:32.00	363	00:00:00.00
23	2002-09-26 01:39:52.00	354	00:00:41.00
23	2002-09-26 01:40:54.00	357	00:16:24.00
23	2002-09-26 01:59:41.00	357	00:05:29.00
23	2002-09-26 02:05:30.00	354	00:00:21.00
23	2002-09-26 02:06:10.00	357	00:33:23.00
23	2002-09-26 12:32:00.00	357	00:00:00.00

Figure 4-5 Example of ping-pong events. Records in green indicate a ping-pong between two access points; the records highlighted in blue indicate a ping-pong between three access points.

The events highlighted in blue could also be considered a ping-pong events, where the user associated with two different access points before returning to the original access point. In the initial analysis, only single 'bounces' between access points were counted and it was found that out of 29,757 transitions between access points, 10,624 (36%) were ping-pong events.

I then looked at the time that was spent at the intermediary access point during the ping-pong event. 5,277 of the 10,624 ping-pong records had durations of 0 seconds, which means that intermediary access point was logged for only one poll.

The following histogram summarizes the amount of time the device spent associated with the intermediary access point.

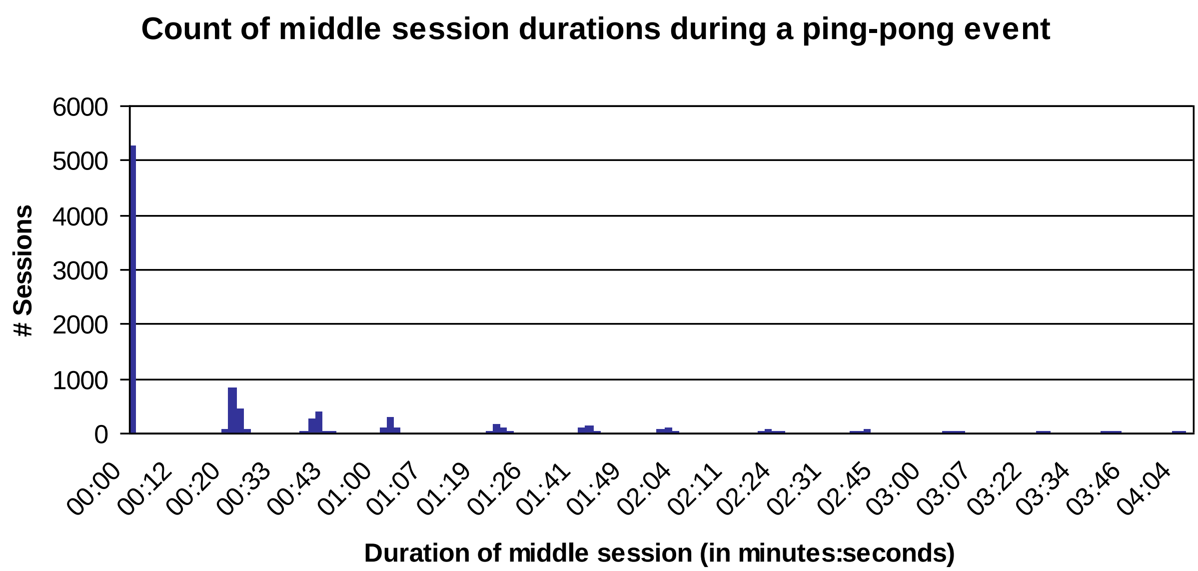


Figure 4-6 Duration of the middle session during a ping-pong event

Enforcing a minimum threshold on the session duration could be used to remove ping-pong events. Removing ping-pong sessions with a duration of 1 minute or less removes approximately 75% of the calculated ping-pong events. Increasing the threshold value to 4 minutes removes over 90% of the ping-pong events. Using a time threshold also removes ping-pong events where the association bounces between more than two access points.

The next step was to go through the data records and purge the ping-pong sessions. Instead of simply deleting these short sessions, they were merged, if appropriate, with the sessions before and/or after. The user sessions were inspected three at a time. If the second session had a duration of less than one minute (the threshold value), it was considered a ping-pong record to be removed. These ping-pong records fall into four categories. If the sessions are labeled A, B, and C, with B being the short middle session, B was either deleted or merged with A or C, depending on certain criteria, as shown in Figure 4-7.

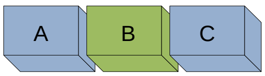
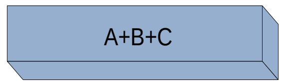
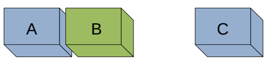
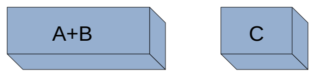
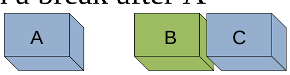
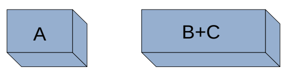
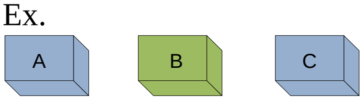
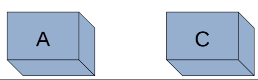
Situation	Resulting Action	# Records which fit this characteristic (out of 94,165 records)
A, B, C are contiguous in time. Ex. 	Merge A, B and C into A 	6892 (7 %)
A and B are contiguous, but there is a time gap before record C. Ex. 	Merge B into A 	2045 (2 %)
B and C are contiguous, with a break after A Ex. 	Merge B and C 	9075 (10 %)
None of the sessions are contiguous Ex. 	Delete record B. 	21641 (23%)

Figure 4-7 Purging Ping-Pong Records

Ping-pongs between multiple access points resulted in multiple records being deleted. The resulting data set was 49% of the size of the original dataset.

One more pass was made to the resulting dataset to merge records which were not ping-pongs, but were contiguous. If two consecutive records recorded the same access point, and the second record started within 30 seconds of the first, they were combined. Of 45,990 records, 292 were found to be contiguous and merged.

The result of this data mining is a set of files, one for each of the 275 users, containing a list of the user’s sessions, meaning the starting time their device associated with an access point, the ID of the access point and the length of time the device was associated. Ping-pong events and sessions shorter than 30 seconds were removed.

4.1.3 Experiments on Access Points with Unknown Locations

Of the 524 access points logged in the UCSD dataset, only 200 have known location coordinates. There are a few approaches that could be used to locate the access points whose position is unknown.

- If the records are short, they could be deleted or merged with other records.
- The original dataset not only includes the associated access point, but two other access points that are sensed. The locations of the sensed access points could be used to calculate the location of the unknown access point. The weighted centroid method, where the average of each of the coordinates is calculated, weighted by the received signal strength, has been shown to be the most accurate method of calculating access point location [21].
- Using WiFi access points to determine location is only accurate to approximately 32 meters [21]. Because of this, a clustering algorithm may be applied to determine significant locations [2]. The access points could be merged with the other access points which were sensed at the same time. This approach removes the need to calculate the access point's location.

To test the accuracy of the weighted centroid algorithm, I calculated the location of a known access point by calculating the centroid of it's neighboring access points, weighted by the number of times each access point appears as a neighbor, and the sum of their signal strengths. The access point in question (#364) was included in 733,665 records of the 13 million in the original dataset. Twenty-six neighboring access points were found, of which 16 had known locations. The location coordinates of these 16 access points were used to calculate the postion of the test access point. The difference between the true location and the calculated location of access point #364 was 73 feet (22 meters). When a *simple centroid* method was used, which disregards received signal strength, the error was 96 feet (29 meters).

The sizes of these error measurements shows that some decision will be required concerning location accuracy when attempting prediction. Due to the coarseness of the calculations, we can not predict a location within a few meters, but would be more successful predicting in which section of a building or neighborhood the device is located. These results justify my decision to use the location of the associated access point as the user's position.

4.1.3.1 Calculating the Position of Unknown Access Points

In another experiment, the weighted centroid method of trilateration was used to calculate the location of the 324 access points without known locations, with the final results shown below in Figure 4-8. Of these 324 access points, 183 had no known neighbors or neighbors whose location was also unknown and therefore their location can not be determined using trilateration. These access points will be marked as UNKNOWN.

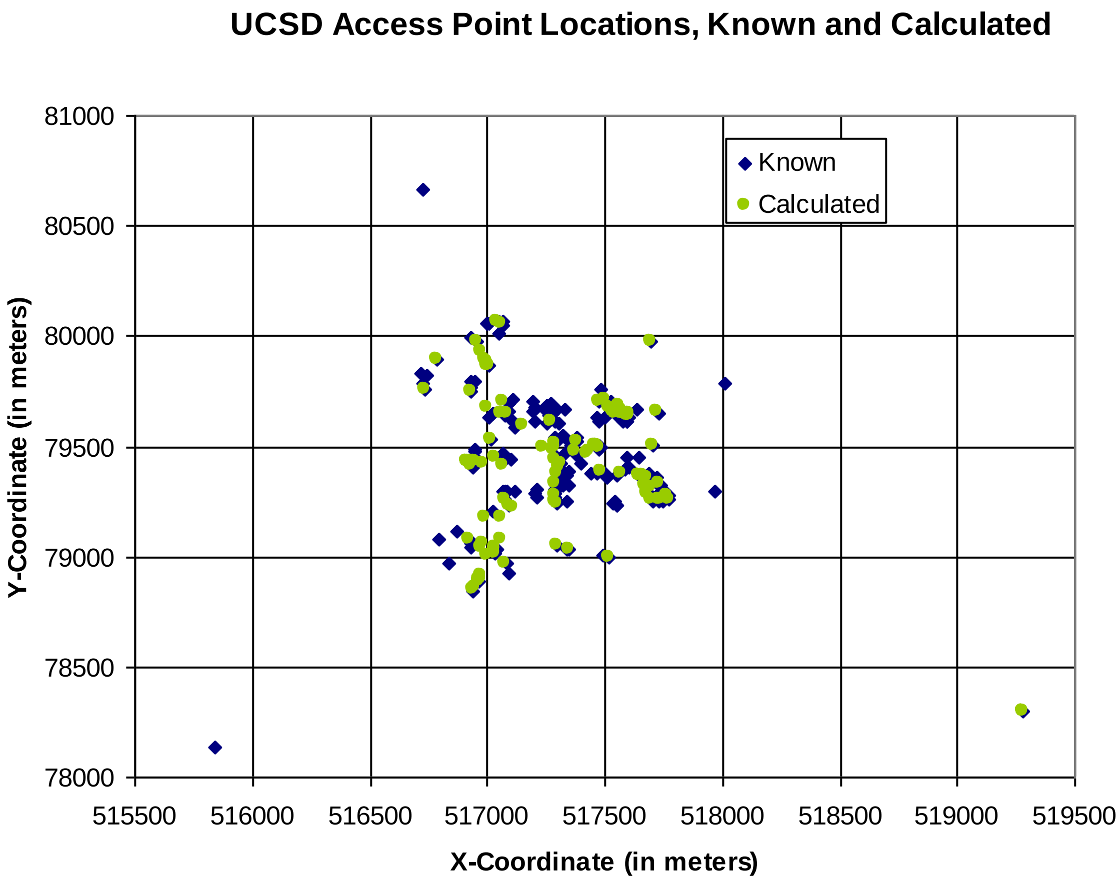


Figure 4-8 Access Point Locations

4.2 Developing the Predictive Model

Before the predictive algorithms could be run, the 11 weeks of data needed to be divided so that one portion could be used for training and one for testing. Experiments were run to determine how to divide the data for testing and training.

4.2.1 Partitioning the Data for Training and Testing

In order to test the predictive models, they need to be trained on a set of data and then queried to predict some test data that are not known to the model. So the first step in developing any of the predictive models is to divide the training data into two parts: the portions used for training and the portions used for testing. The code used by Begleiter, Java-based with a MATLAB wrapper, is available online [48] and was used to determine how to divide the data.

The UCSD data covers 11 weeks. The week fragments at the beginning and the end of the dataset were dropped to result in 10 full weeks of data. A series of tests were run using various subsets of the data for training and testing, as shown below.

Table 4-1 Division of the dataset by weeks for training and testing

Weeks used for Training	Week(s) used to test
1	6, 10
1,2	6,10
1-3	6,10
1-4	6,10
1-5	6,10
2-5	6
3-5	6
4-5	6
5	6
4-9	10
5-9	10
6-9	10
7-9	10
8-9	10
9	10

Six users were selected from the dataset. The most mobile users, who associated with the largest number of access points (72, 67, and 64), were selected, along with three fairly mobile users who each associated with 33 access points.

The location data for each of the combinations of training and test weeks were culled for each user, creating a series of text strings that represented the users' locations for the weeks indicated. For each test, the PPM-C model was trained on the training data and the average average-log-loss was calculated for the test data. The averages of these tests were compared and are shown in Figure 4-9. Recall from Section 2.3, that lower values for average average-log-loss indicate better prediction. Comparison of the various training and testing weeks reveals that using the weeks immediately preceding the test week produces the best results.

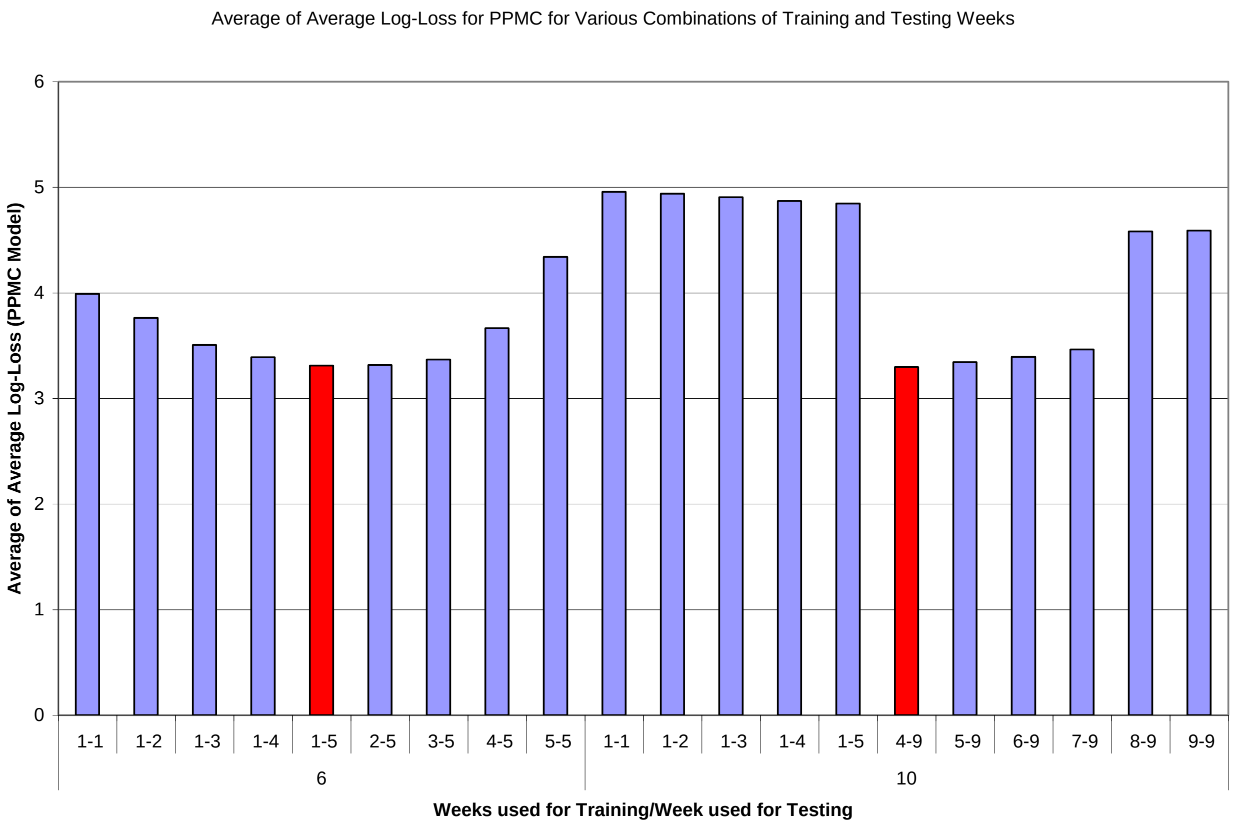


Figure 4-9 Average Log-Loss for Different Training and Testing Schemes (for sequences containing only locations). The lowest values (indicated by red bars) indicate the best predictions.

For the rest of the project, weeks 1-5 will be used for training and the model will be tested using the data from week #6. Then weeks 4-9 will be used to re-train the model and week 10 will be used for testing.

4.3 Maximum Order

The maximum order, or depth, of the Markov models is a significant parameter. The tree that is constructed as part of the PPM-C model is complete, meaning that every branch has leaves that extend to the full maximum order of the model. In the worst case, the size of the tree grows exponentially by the size of the alphabet. For a tiny alphabet of size 4, the model has 4 leaves at level 0 and 16 at level 1. Incrementing the order adds another 64 leaves, bring the total to 84, a 4-fold increase. This implementation of the model assumes an alphabet size of 256 characters, which means that for each increase in maximum order, the tree grows by 256**(maximum_order – 1).

My goal is to find the minimum value for the maximum order that yields satisfactory results. This test was run using various representations of time and location. The training data consist of the 5 weeks which occurred right before the week of testing data. For example, the model was trained on a particular user's locations during weeks 1 – 5 and then asked to predict the user's locations during week 6. The histogram below shows the results of varying the maximum order of the model.

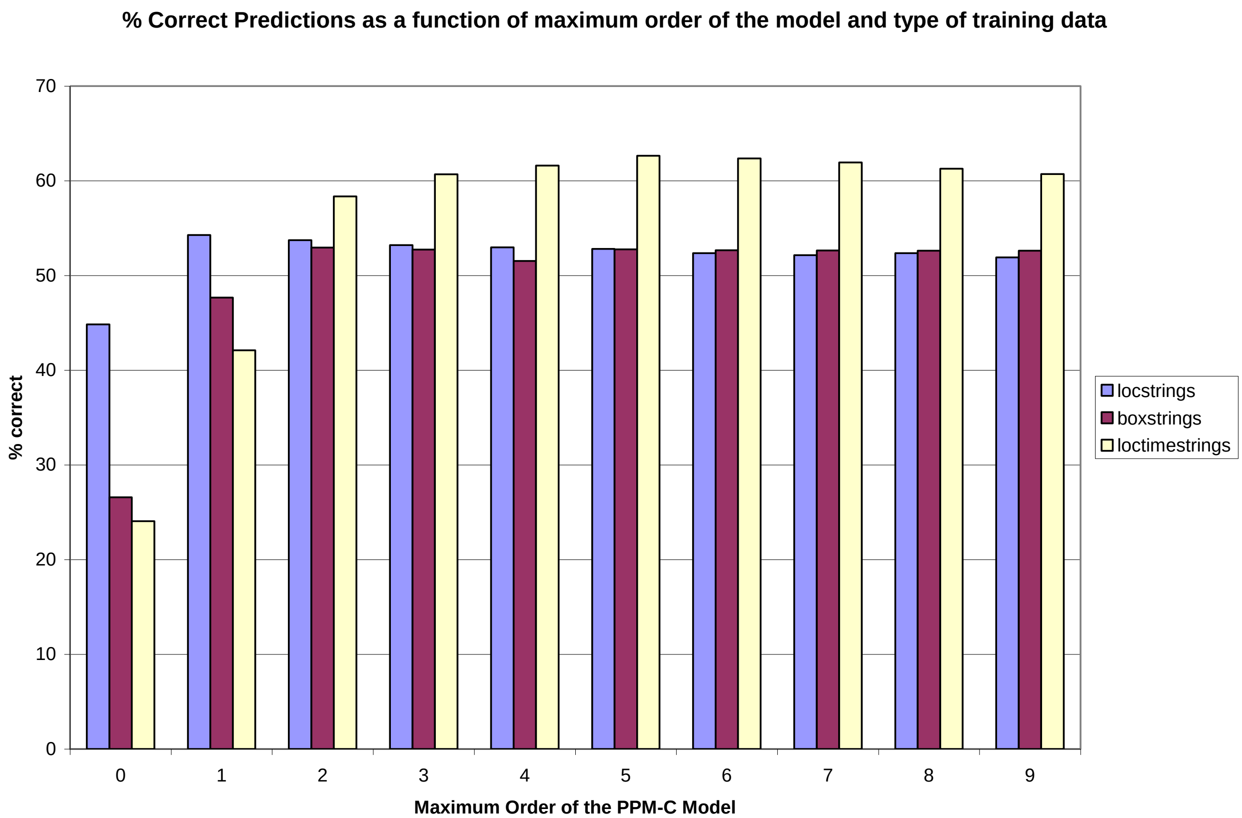


Figure 4-10 Correct Predictions as a function of the Maximum Order of the Model

Once the maximum order reaches 2 or 3, depending on the input type, the percentage of correct answers does not significantly improve. Tests on English text compression using the PPMC model show that a maximum order of 5 yields the best results [59]. I chose order = 2 for the preliminary tests of the PPM model. I expect that the optimal order will change for different representations as more context is added to the representations. For example, for a sequence that consists of only locations, an order of 1 or 2 seems reasonable, since someone’s location is most dependent on the immediately preceding location [37], which only requires one character to express. However, for a sequence that includes time, more characters are needed to express the previous location and the previous time, so I expect that higher orders may produce better results for representations with different types of information.

4.3.1 Representations

I then selected one model to work with, Prediction-by-Partial-Match (PPM-C) model, as I began experimenting with different representations of the location and time information. For each ‘session’, I knew the location (access point), the starting time and how long the user was at that location (the duration). The representation involves translating the location and time information into a sequence of characters, as discussed earlier in Section 3.3.2.

Several different combinations of representations were tried, from the simplest sequence of locations only, to combinations that included times and/or session durations. The original PPM model supported only 8-bit characters. 256 characters was not enough to translate each time segment into a unique character. In order to represent time, I tried mapping times to two-character strings (e.g. 8:00am = “aa”) and using the time string directly (“08:00”). I also tried adding delimiters, such as semi-colons, between sessions. I hypothesized that the use of the delimiters would enhance the model by indicating which fields were times, which were locations and when the end-of-the day occurred. The opposite result occurred. The use of delimiters reduced the accuracy of the predictions.

Mapping locations and times onto two-character strings degraded the performance of the model, so I found an implementation of the basic PPM-C data compression algorithm in C [60] and expanded it to report predictions and to support 16-bit symbols. This model will support an alphabet size of 65,535, which is enough for the several hundred locations and possibly thousands of timeslots to be mapped to single character values. The 16-bit version was used for the remaining experiments.

An experiment was run to determine the effect of using different types of representations for someone’s location and time information, as listed in Table 4-2.

Table 4-2 Types of Representations

Symbol	Meaning
L (Location)	Location
T (Time)	Starting time and location
D (Duration)	Starting time, location and duration
P (Polled)	Polled (time and location at 10-minute intervals)
S (Sequence)	Sequence (location at 10-minute interals)

For example, the representation called 'L' is simply a list of all the locations the user visited over the course of several weeks. The representation labeled 'TL', inserts the starting time, eg. at 10:00, the user was at location A, then at 10:30, the user was at location B. The 'D' label includes the duration; eg. At 10:00, for one hour, the user was at location A. Then at noon, for 20 minutes, the user was at location C. Different orderings and combinations of the location, starting time and duration symbols were tested.

The 'P' representation (short for 'polled') records the users' locations (if known) at 10 minute intervals. For example, if the user was at location A for a half hour starting at 9:00, and then at location B for 10 minutes starting at noon, the 'P' representation would look like this:

9:00, A, 9:10, A, 9:20, A, 12:00, B…

The 'S' representation (sequence) is the same as the polled representation without the time information. It lists the user's location at each 10-minute interval. In the first set of trials, intervals where the device is not connected were not included.

The test was run over 9 datasets covering 6 users. The PPM model was trained over a 5-week period and then asked to predict the locations for the following week. For example, if the test data consisted of the string of locations *abcdef*, the model would be given no context and tested to see if it predicted 'a', then it would be given the context 'a' and tested to see if it predicted 'b', etc. Even if the data included time information, this initial experiment only tested the model's ability to predict the user's next location. The percentage of locations that were correctly predicted for each representation is shown below.

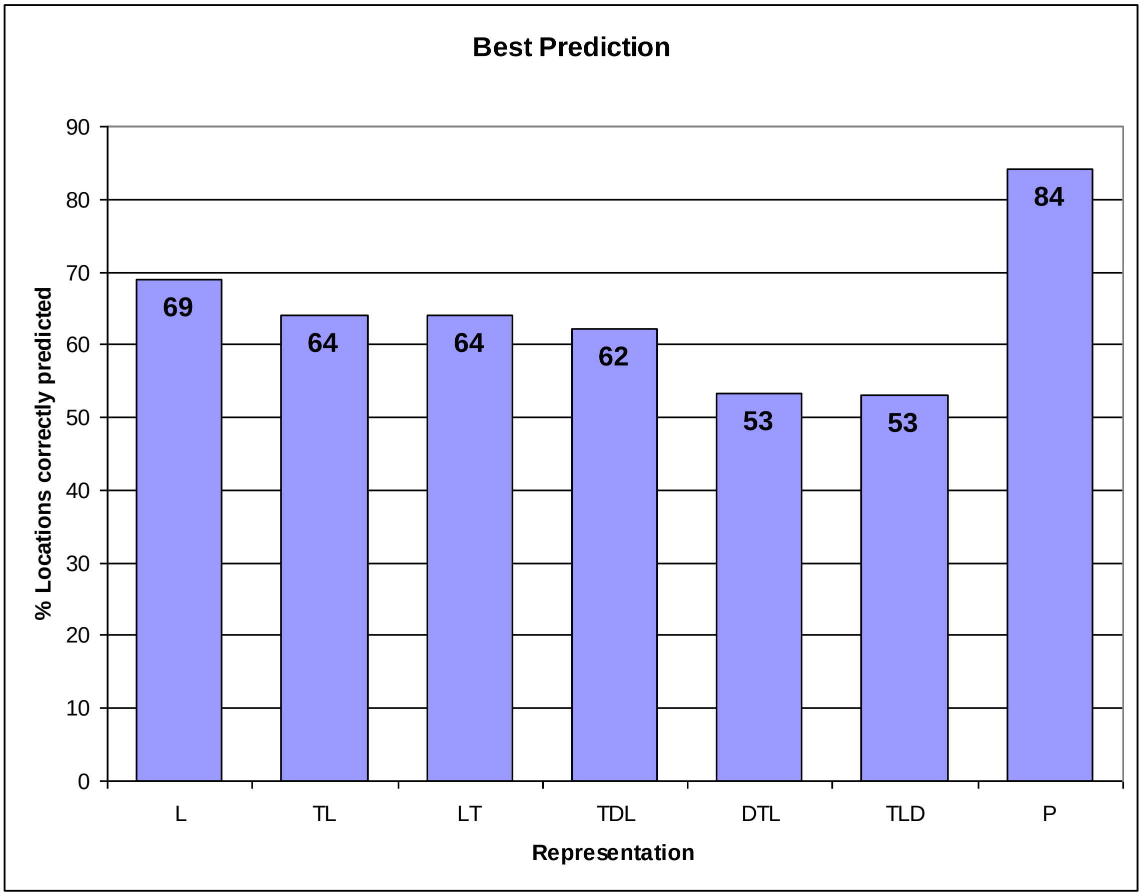


Figure 4-11 Accuracy in predicting the next location in the PPM model for various representations of the data.

Further experiments were done on the 'P' (polled) representation. The number of times that the time *and* the location were correctly predicted was calculated. An additional representation, called 'S' (for sequence) was added. The 'S' representation is similar to the 'P' representation, except that the time information is removed. For example, if Bob's morning is shown in Table 4-3 below, the various representations are shown in Table 4-4.

Table 4-3 Bob's Morning

Starting Time	Location	Duration
9:00 am	A	30 minutes
10:00 am	B	20 minutes
Noon	C	10 minutes

Table 4-4 Examples of different types of representations

Representation	Example for Bob's Morning
P	9:00, A, 9:10, A, 9:20, A, 10:00, B, 10:10, B, 12:00,C
L	ABC
S	AAABBC

The number of correct predictions for *polled* and *sequence* representations are shown below.

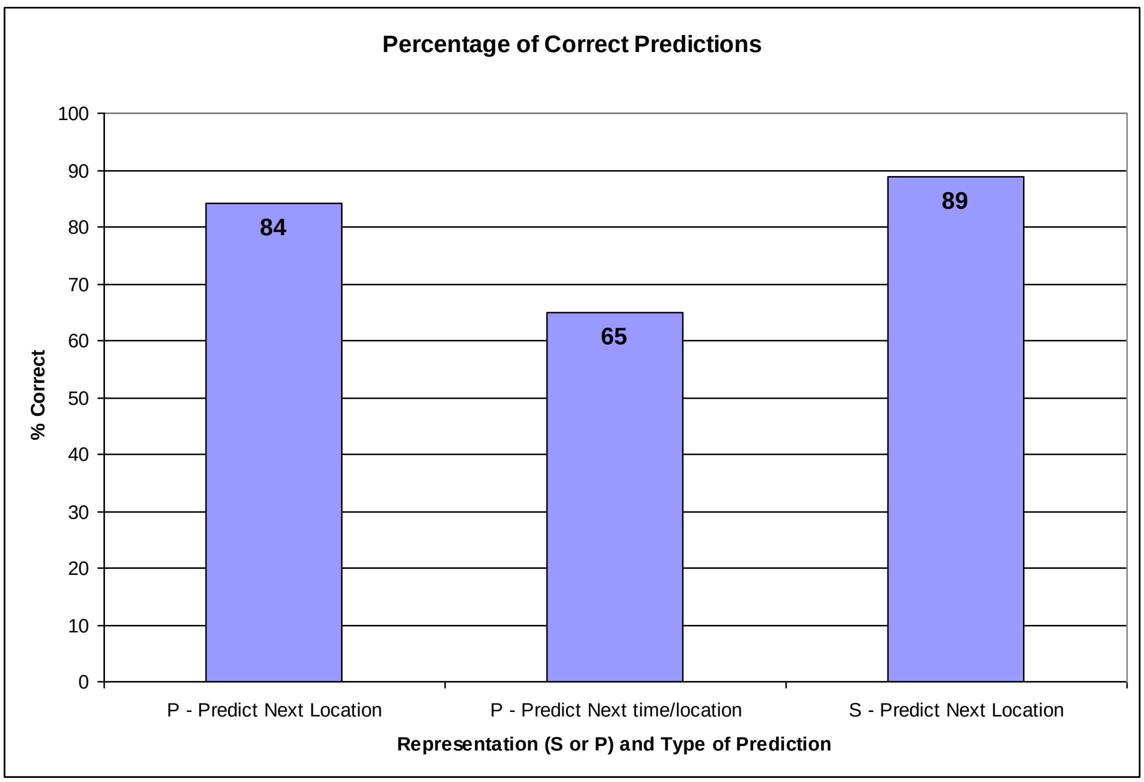


Figure 4-12 Correct Predictions when predicting time and location

The leftmost bar shows that for the polled sequence, the PPM-C model correctly predicts our test users' next location 85% of the time. The rightmost bar is the successful prediction rate for a sequence of locations (e.g. where the user is every 10 minutes). The PPM-C model correctly predicts the next location 89% of the time. This representation of the data does not explicitly include time information, so the model is unable to predict time of arrival at the next location. **65% of the time, it correctly predicts the next location *and* the time the user arrived at that location!**

5 Conclusion

In summary, the objectives of this project are to determine location, train a predictive model, and use that model to forecast future locations and times. To achieve these objectives, the following steps will be completed (as described in detail in Chapter 3). If a step is marked with a checkmark, that step has been completed (as described in Chapter 4).

5.1 Location Determination

1. Divide the large dataset into files for each individual. ☑
2. Delete records without associated access points. ☑
3. Create sessions. ☑
4. Create datasets where sessions with time gaps of less than 1, 2, 5 and 10 minutes between them are combined. (The current sessions, used for the preliminary results described in Chapter 4, use a threshold of 30 seconds.)
5. Find and merge ping-pong sessions. ☑

6. Translate into symbolic locations (including UNKNOWN) as explained in Section 3.2.

5.2 Predictive Models

As explained in Section 3.3.3, the tasks for developing the predictive models include:

1. Implement PPM, expand to use 16-bit symbols. ☑
2. Implement LZ-78, expand to use 16-bit symbols.
3. Implement CTW, expand to use 16-bit symbols.
4. Test against different representations.
5. Measure accuracy, speed, memory used, and ability to answer different types of forecasting questions.

5.3 Forecasting

As explained in Section 3.4, the tasks involved in the forecasting component of this project include:

1. Test accuracy when predicting next location and time.
2. Test accuracy when predicting future locations.
3. Test accuracy when predicting times that future locations will be visited.

5.4 Anticipated Problems – the CTW Algorithm

The Context-Tree-Weighting algorithm is based on a binary tree, which can easily grow to consume large quantities of memory, especially with the large 16-bit alphabet proposed in this project. Numerous optimizations of Context-Tree-Weighting algorithms have been proposed, but it is beyond the scope of this current project to research them all and implement a complex algorithm in C. I will initially test the predictive performance of the CTW algorithm using the Java implementation available from [48]. If the results are promising, I will proceed to implement the algorithm in C. If the results are not better than the other algorithms tested, I will discontinue my investigation into the CTW algorithm.

5.5 Timeline

The remaining tasks for the location determination portion of this project are fairly straight-forward. The bulk of the research will be in developing, implementing and testing the predictive models. Finally, I will implement forecasting queries and write the dissertation.

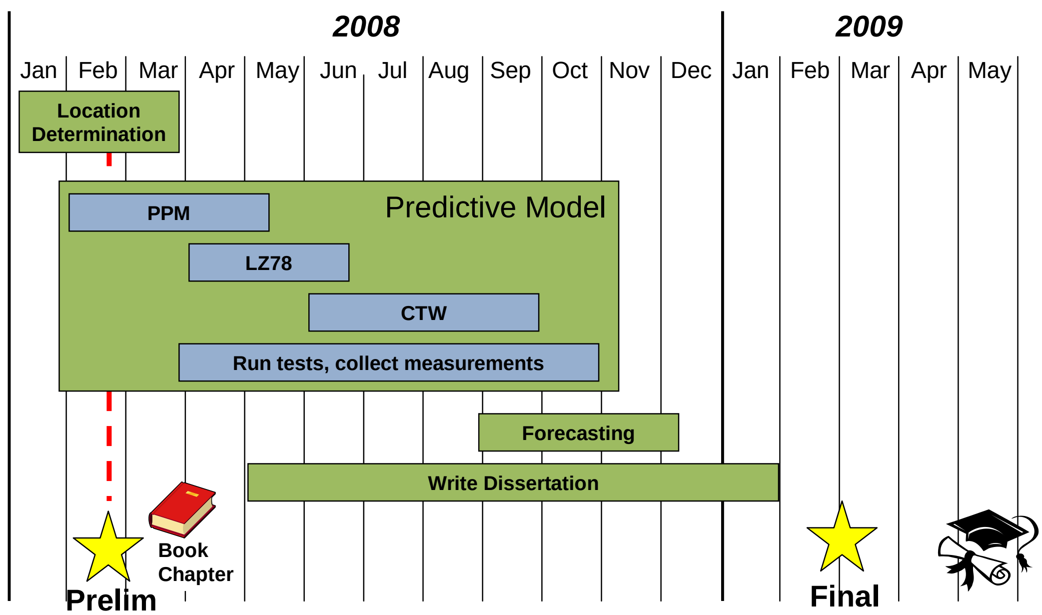


Figure 5-1 Research Timeline

"Nothing would be done at all if a man waited until he could do it so well that no one could find fault with it."
- Cardinal Newman

6 Personal Qualifications

I enjoy learning, and through the past twenty years, I have almost been continually in school, taking a wide variety of courses in Computer Engineering and other subjects. For the majority of the past 18 years, I have also been self-employed, developing software in C, Visual Basic, and 8051-assembler for lottery companies, state lotteries, and small businesses. During my time at Virginia Tech, I have:

- Been an IGERT/IREAN trainee (2004-2006). It was my pleasure to represent our IGERT program and present a poster at the 2006 NSF IGERT Project Meeting.
- Received an Honorable Mention in the NSF Graduate Research Fellowship Program (2005).
- Maintained a 3.97 GPA over the twelve courses I have taken from six different departments (CS, ECE, ACIS, STAT, FIN, ISE).

- Received (and turned down) the 2006-07 QUALCOMM “Q Award of Excellence” scholarship.
- Been certified as a Woman-Owned (SWAM) Business in Virginia and I have started (with others) two small local businesses..

6.1 Conference & Workshop Publications

- D. Raymond, I. Burbey, Y. Zhao, C.P. Koelling (2006). “Impact of Mobility Models on Simulated Ad Hoc Network Performance.” 9th International Symposium on Wireless Personal Multimedia Communications, San Diego, CA, September 17-20, 2006.
- U. Murthy, I. Burbey, G. Kwon (2006). “Refinding from a Human Information Processing Perspective.” Personal Information Management Workshop at ACM Special Interest Group on Information Retrieval (SIGIR 2006), Seattle, WA, August 6-11, 2006.
- S. Midkiff, G. Morgan, C.P. Koelling, S. Ball, I. Burbey, M.J. Zukoski, D. Maldonado-Febus (2006). “Catalyzing Multidisciplinary Research through Innovative Multidisciplinary Graduate Education.” 9th International Conference on Engineering Education, San Juan, PR, July 23-28,2006.
- C. L. Bowen, T. K. Buennemeyer, I. Burbey, V. Joshi (2006). “Using Wireless Networks to Assist Navigation for Individuals with Impaired Mobility.” 21st Annual International Technology and Persons with Disabilities Conference, Los Angeles, California, California State University Northridge Center on Disabilities.
- M. Tungare, I. Burbey, M.A. Perez-Quinones, and A. Raghavan (2004). “Evaluation of a Location Linked Notes System.” ACM South East Conference 2006.

6.2 Book Chapters

- I.Burbey, "Ubiquitous Internet Computing," Chapter in World Wide Web - Beyond the Basics, Prentice Hall, 1998.

References

[1] M. Satyanarayanan, “A catalyst for mobile and ubiquitous computing,” *Pervasive Computing, IEEE,* vol. 1, no. 1, pp. 2-5, 2002.

[2] D. Ashbrook, and T. Starner, “Using GPS to learn significant locations and predict movement across multiple users,” *Personal and Ubiquitous Computing,* vol. 7, no. 5, pp. 275-286, 2003.

[3] S. K. Das, D. J. Cook, A. Battacharya *et al.*, “The role of prediction algorithms in the MavHome smart home architecture,” *IEEE Wireless Communications,* vol. 9, no. 6, pp. 77-84, 2002.

[4] J. Richards, and J. Christensen, “People in Our Software,” *ACM Queue,* vol. 1, no. 10, pp. 80-86, 2004.

[5] G. Iachello, I. Smith, S. Consolvo *et al.*, “Control, Deception, and Communication: Evaluating the Deployment of a Location-Enhanced Messaging Service,” in Ubicomp 2005, Tokyo, Japan, 2005.

[6] J. Gemmell, G. Bell, and R. Lueder, “MyLifeBits: a personal database for everything,” *ACM Queue,* vol. 49, no. 1, pp. 88-95, 2006.

[7] DARPA. "Location and Connection Aware Content Pushing (LOCO) Program," 2007; http://www.darpa.mil/sto/solicitations/LOCO/index.html.

[8] P. Dourish, and V. Bellotti, “Awareness and coordination in shared workspaces,” in Proceedings of the 1992 ACM conference on Computer-supported cooperative work, Toronto, Ontario, Canada, 1992.

[9] E. Isaacs, A. Walendowski, and D. Ranganathan, “Mobile instant messaging through Hubbub,” *Commun. ACM,* vol. 45, no. 9, pp. 68-72, 2002.

[10] D. J. Patterson, L. Liao, K. Gajos *et al.*, “Opportunity Knocks: A System to Provide Cognitive Assistance with Transportation Services,” in UbiComp, 2004, pp. 433-450.

[11] M. G. Guillemette, I. Fontaine, and C. Caron, "Hybrid RFID-GPS Real-Time Location System for Human Resources: Development, Impacts and Perspectives." pp. 406-406.

[12] B. Christensen. "'Minority Report' Ads Are In Your Future," 17 October, 2006; http://www.livescience.com/scienceoffiction/060917_bluscreen.html.

[13] C. Diot. "Haggle - A European Union funded project in Situated and Autonomic Communications," 9 January, 2008; http://www.haggleproject.org.

[14] S. Jing, A. Chin, A. Popivanova *et al.*, "User mobility for opportunistic ad-hoc networking." pp. 41-50.

[15] M. Weiser, “The Computer for the Twenty-First Century,” *Scientific American,* vol. 265, no. 3, pp. 94-104, 1991.

[16] G. Borriello, “The Invisible Assistant,” *ACM Queue,* vol. 4, no. 6, pp. 44-49, 2006.

[17] N. B. Archive, "Niels Bohr Archive," 2005.

[18] J. Petzold, F. Bagci, W. Trumler *et al.*, “Next Location Prediction Within a Smart Office Building,” in Workshop on Exploiting Context Histories in Smart Environments - Pervasive 2005, Munich, Germany, 2005.

[19] A. LaMarca, Y. Chawathe, S. Consolvo *et al.*, “Place Lab: Device Positioning Using Radio Beacons in the Wild.,” in Pervasive 2005, Munich, Germany, 2005.

[20] "Skyhook Wireless," February 11, 2008; www.skyhookwireless.com.

[21] M. Kim, J. J. Fielding, and D. Kotz, "Risks of using AP locations discovered through war driving," *Lecture Notes in Computer Science.* pp. 67-82.

[22] D. Kotz, and T. Henderson, “CRAWDAD: A Community Resource for Archiving Wireless Data at Dartmouth,” *IEEE Pervasive Computing,* vol. 4, no. 4, pp. 12-14, 2005.

[23] M. McNett, and G. M. Voelker, "UCSD Wireless Topology Discovery Trace," Support for the UCSD Wireless Topology Discovery Trace was provided by DARPA Contract N66001-01-1-8933., 2005.

[24] J. Hightower, and G. Borriello, “Location systems for ubiquitous computing,” *Computer,* vol. 34, no. 8, pp. 57-66, August 2001, 2001.

[25] J. Hightower, S. Consolvo, A. LaMarca *et al.*, "Learning and Recognizing the Places We Go," *Lecture Notes in Computer Science.* pp. 159-176.

[26] C. Zhou, P. Ludford, D. Frankowski *et al.*, “An experiment in discovering personally meaningful places from location data,” in CHI '05 extended abstracts on Human factors in computing systems, Portland, OR, USA, 2005.

[27] N. Priyantha, A. Chakraborty, and H. Balakrishnan, “The Cricket Location-Support System,” in MOBICOM, Boston, MA, 2000.

[28] R. Want, A. Hopper, V. Falcao *et al.*, “The Active Badge Location System,” *ACM Transactions on Information Systems,* vol. 10, no. 1, pp. 91-102, January, 1992.

[29] M. Balazinska, and P. Castro, "CRAWDAD data set ibm/watson (v. 2003-02-29)," 2003.

[30] D. J. Cook, M. Youngblood, E. O. Heierman, III *et al.*, "MavHome: an agent-based smart home." pp. 521-524.

[31] A. Bhattacharya, and S. K. Das, “LeZi-update: an information-theoretic approach to track mobile users in PCS networks,” in 5th annual ACM/IEEE international conference on Mobile computing and networking, Seattle, Washington, United States, 1999.

[32] J. Ziv, and A. Lempel, “Compression of individual sequences via variable-rate coding,” *IEEE Transactions on Information Theory,* vol. 24, no. 5, pp. 530-536, 1978.

[33] A. Roy, S. K. Das, and K. Basu, “A Predictive Framework for Location-Aware Resource Management in Smart Homes,” *IEEE Transactions on Mobile Computing,* vol. 6, no. 11, pp. 1284-1283, 2007.

[34] V. R. Jakkula, and D. J. Cook, “Using Temporal Relations in Smart Environment Data for Activity Prediction,” in 24th International Conference on Machine Learning, Corvallis, OR, 2007.

[35] D. Ashbrook, and T. Starner, “Learning significant locations and predicting user movement with GPS,” in Sixth International Symposium on Wearable Computers, 2002, pp. 101.

[36] R. Hariharan, and K. Toyama, “Project Lachesis: Parsing and Modeling Location Histories,” in Second International Conference on Geographic Information Science, Adelphi, MD, 2004, pp. 106-124.

[37] L. Song, D. Kotz, R. Jain *et al.*, “Evaluating location predictors with extensive Wi-Fi mobility data,” in Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOMM 2004, 2004, pp. 1414-1424.

[38] J. Petzold, “Zustandspradiktoren zur Kontextvorhersage in ubiquitaren Systemen,” der Fakult¨at f¨ur Angewandte Informatik, Universitat Augsburg, Augsburg, 2005.

[39] J. Petzold, A. Pietzowski, F. Bagci *et al.*, *Prediction of Indoor Movements Using Bayesian Networks*, 2005.

[40] J. Petzold, F. Bagci, W. Trumler *et al.*, “Global State Context Prediction Techniques Applied to a Smart Office Building,” in Communications Networks and Distributed Systems Modeling and Simulation Conference (CNDS), San Diego, CA, 2004.

[41] J. Petzold, F. Bagci, W. Trumler *et al.*, "Confidence Estimation of the State Predictor Method," *Lecture Notes in Computer Science*, pp. 375-386, Berlin / Heidelberg: Sprinter, 2004.

[42] J. Tullio, “Exploring the Design and Use of Forecasting Groupware Applications with an Augmented Shared Calendar,” Computer Science, Georgia Institute of Technology, 2005.

[43] R. Hill, and J. Begole, “Activity rhythm detection and modeling,” in CHI '03 extended abstracts on Human factors in computing systems, Ft. Lauderdale, Florida, USA, 2003.

[44] J. B. Begole, C. T. John, and H. Rosco, “Rhythm modeling, visualizations and applications,” in Proceedings of the 16th annual ACM symposium on User interface software and technology, Vancouver, Canada, 2003.

[45] I. C. K. Chen, T. C. John, and N. M. Trevor, “Analysis of branch prediction via data compression,” in Proceedings of the seventh international conference on Architectural support for programming languages and operating systems, Cambridge, Massachusetts, United States, 1996.

[46] J. S. Vitter, and P. Krishnan, “Optimal prefetching via data compression,” *J. ACM,* vol. 43, no. 5, pp. 771-793, 1996.

[47] M. Deshpande, and G. Karypis, "Selective Markov models for predicting Web page accesses," 2, ACM Press, 2004, pp. 163-184.

[48] R. Begleiter, R. El-Yaniv, and G. Yona, “On Prediction Using Variable Order Markov Models,” *Journal of Artificial Intelligence Research,* vol. 22, pp. 385-421, 2004.

[49] K. Gopalratnam, and D. J. Cook, “Online Sequential Prediction via Incremental Parsing: The Active LeZi Algorithm,” *Intelligent Systems,* vol. 22, no. 1, pp. 52-58, 2007.

[50] J. Cleary, and I. Witten, “Data Compression Using Adaptive Coding and Partial String Matching,” *IEEE Transactions on Communications,* vol. 32, no. 4, pp. 396-402, 1984.

[51] D. J. C. MacKay, *Information Theory, Inference & Learning Algorithms*, 1st edition ed.: Cambridge University Press, 2002.

[52] F. M. J. Willems, Y. M. Shtarkov, and T. J. Tjalkens, “The context-tree weighting method: basic properties,” *IEEE Transactions on Information Theory,* vol. 41, no. 3, pp. 653-664, May 1995, 1995.

[53] P. A. J. Volf, “Weighting techniques in data compression theory and Algorithms,” Technische Universiteit Eindhoven, 2002.

[54] G. Langdon, Jr., “A note on the Ziv - Lempel model for compressing individual sequences (Correspondence),” *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 284-287, 1983.

[55] J. Rissanen, “A universal data compression system,” *IEEE Transactions on Information Theory,* vol. 29, no. 5, pp. 656-664, 1983.

[56] M. Kim, and D. Kotz, “Classifying the Mobility of Users and the Popularity of Access Points,” in First International Workshop on Location- and Context-Awareness, Oberpfaffenhofen, Germany, 2005, pp. 198.

[57] S. D. Hermann, M. Emmelmann, O. Belaifa *et al.*, “Investigation of IEEE 802.11k-based Access Point Coverage Area and Neighbor Discovery,” in 32nd IEEE Conference on Local Computer Networks, 2007, pp. 949-954.

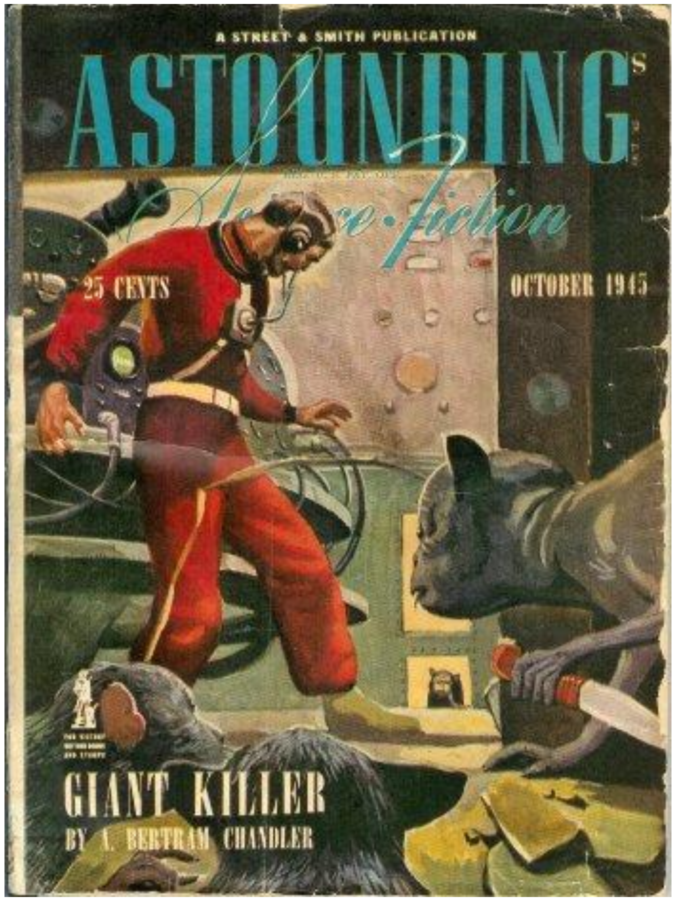
[58] M. Gast, "802.11 Wireless Networks: The Definitive Guide," O'Reilly, 2005.

[59] J. G. Cleary, W. J. Teahan, and I. H. Witten, "Unbounded length contexts for PPM." pp. 52-61.

[60] M. Nelson, "Arithmetic Coding + Statistical Modeling = Data Compression," *Dr. Dobb's Journal*, February, 1991, 1991.

[61] A. Ganzer, "What You Need," *The Twilight Zone*, 1959.

Appendix – What You Need



Most of us came to technology through science fiction; our imaginations remain secretly moved by science-fictional ideas.
- Jason Pontin

Sometime science is inspired by Science Fiction. This Science Fiction story, from the October 1945 issue of Astounding Science Fiction explores what can happen when a device can predict the future. [I would like to thank John M. Jackson of the Special Collections department of the Virginia Tech library for finding and photocopying this short story for me.] The following short story was changed slightly and used as an episode in the TV series, “The Twilight Zone” in its first season, in 1959 [61].

What You Need

by LEWIS PADGETT

What you need may turn out to be a pair of scissors, or a hen's egg, or any of a number of remarkably uninteresting things—till you find why you need it more than anything else on Earth!

Illustrated by Williams

WE HAVE WHAT YOU NEED

That's what the sign said. Tim Carmichael, who worked for a trade paper that specialized in economics, and eked out a meager salary by selling sensational and untrue articles to the tabloids, failed to sense a story in the reversed sign. He thought it was a cheap publicity gag, something one seldom encounters on Park Avenue, where the shop fronts are noted for their classic dignity. And he was irritated.

He growled silently, walked on,

133

WHAT YOU NEED

then suddenly turned and came back. He wasn't quite strong enough to resist the temptation to unscramble the sentence, though his annoyance grew. He stood before the window, staring up, and said to himself, "We Have What You Need. Yeah?"

The sign was in prim, small letters on a black painted ribbon that stretched across a narrow glass pane. Below it was one of those curved, invisible-glass windows. Through the window Carmichael could see an expanse of white velvet, with a few objects carefully arranged there. A rusty nail, a snowshoe, and a diamond tiara. It looked like a Dali decor for Cartier's or Tiffany.

"Jewelers?" Carmichael asked silently. "But why *what you need?*" He pictured millionaires miserably despondent for lack of a matched pearl necklace, heiresses weeping inconsolably because they needed a few star sapphires. The principle of luxury merchandising was to deal with the whipped cream of supply and demand; few people needed diamonds. They merely wanted them and could afford them.

"Or the place might sell jinniflasks," Carmichael decided. "Or magic wands. Same principle as a Coney carny, though. A sucker trap. Bill the Whatzit outside and people will pay their dimes and flock in. For two cents—"

He was dyspeptic this morning, and generally disliked the world. Prospect of a scapegoat was attractive, and his press card gave him a certain advantage. He opened the door and walked into the shop.

It was Park Avenue, all right. There were no showcases or counters. It might be an art gallery, for a few good oils were displayed on the walls. An air of overpowering luxury, with the bleakness of an unlived-in place, struck Carmichael.

Through a curtain at the back came a very tall man with carefully-combed white hair, a ruddy, healthy face, and sharp blue eyes. He might have been sixty. He wore expensive but careless tweeds, which somehow jarred with the decor.

"Good morning," the man said, with a quick glance at Carmichael's clothes. He seemed slightly surprised. "May I help you?"

"Maybe." Carmichael introduced himself and showed his press card.

"Oh? My name is Talley. Peter Talley."

"I saw your sign."

"Oh?"

"Our paper is always on the lookout for possible write-ups. I've never noticed your shop before—"

"I've been here for years," Talley said.

"This is an art gallery?"

"Well—no."

The door opened. A florid man came in and greeted Talley cordially. Carmichael, recognizing the client, felt his opinion of the shop swing rapidly upward. The florid man was a Name—a big one.

"It's a bit early, Mr. Talley," he said, "but I didn't want to delay. Have you had time to get . . . what I needed?"

"Oh, yes. I have it. One moment." Talley hurried through the draperies and returned with a small, neatly-wrapped parcel which he gave to the florid man. The latter forked over a check—Carmichael caught a glimpse of the amount and gulped—and departed. His town car was at the curb outside.

Carmichael moved toward the door where he could watch. The florid man seemed anxious. His chauffeur waited stolidly as the parcel was unwrapped with hurried fingers.

"I'm not sure I'd want publicity, Mr. Carmichael," Talley said. "I've a select clientele—carefully chosen."

"Perhaps our weekly economic bulletins might interest you—"

Talley tried not to laugh. "Oh, I don't think so. It really isn't in my line."

The florid man had finally unwrapped the parcel and taken out an egg. As far as Carmichael could see from his post near the door, it was merely an ordinary egg. But its possessor regarded it almost with awe. Had Earth's last hen died ten years before, he could have been no more pleased. Something like deep relief showed on the Florida-tanned face.

He said something to the chauffeur, and the car rolled smoothly forward and was gone.

"Are you in the dairy business?" Carmichael asked abruptly.

"No."

"Do you mind telling me what your business is?"

"I'm afraid I do, rather," Talley said.

Carmichael was beginning to scent a story. "Of course I could find out through the Better Business Bureau—"

"You couldn't."

"No? They might be interested in knowing why an egg is worth five thousand dollars to one of your customers."

Talley said, "My clientele is so small I must charge high fees. You . . . ah . . . know that a Chinese mandarin has been known to pay thousands of *taels* for eggs of proved antiquity."

"That guy wasn't a Chinese mandarin," Carmichael said.

"Oh, well. As I say, I don't welcome publicity—"

"I think you do. I was in the advertising game for a while. Spelling your sign backwards is an obvious baited hook."

"Then you're no psychologist," Talley said. "It's just that I can afford to indulge my whims. For five years I looked at that window every day and read the sign backward—from inside my shop. It annoyed me. You know how a word will begin to look funny if you keep staring on it? Any word. It turns into something in no human tongue. Well, I discovered I was getting a neurosis about that sign. It makes no sense backwards, but I kept finding myself trying to read sense into it. When I started to say 'Deen uoy tahw evah ew' to myself and looking for philological derivations, I called in a sign painter. People who are interested enough still drop in."

"Not many," Carmichael said

shrewdly. "This is Park Avenue. And you've got the place fixed up too expensively. Nobody in the low-income brackets—or the middle brackets—would come in here. So you run an upper-bracket business."

"Well," Talley said, "yes, I do."

"And you won't tell me what it is?"

"I'd rather not."

"I can find out, you know. It might be dope, pornography, high-class fencing—"

"Very likely," Mr. Talley said smoothly. "I buy stolen jewels, conceal them in eggs, and sell them to my customers. Or perhaps that egg was loaded with microscopic French postcards. Good morning, Mr. Carmichael."

"Good morning," Carmichael said, and went out. He was overdue at the office, but annoyance was the stronger motivation. He played sleuth for a while, keeping an eye on Talley's shop, and the results were thoroughly satisfactory—to a certain extent. He learned everything but why.

Late in the afternoon, he sought out Mr. Talley again.

"Wait a minute," he said, at sight of the proprietor's discouraging face. "For all you know, I may be a customer."

Talley laughed.

"Well, why not?" Carmichael compressed his lips. "How do you know the size of my bank account? Or maybe you've got a restricted clientele?"

"No. But—"

Carmichael said quickly. "I've been doing some investigating. I've been noticing your customers. In fact, following them. And finding out what they buy from you."

Talley's face changed. "Indeed?"

"In-*deed*. They're all in a hurry to unwrap their little bundles. So that gave me my chance to find out. I missed a few, but—I saw enough to apply a couple of rules of logic, Mr. Talley. *Item*, your customers don't know what they're buying from you. It's a sort of grab bag. A couple of times they were plenty surprised. The man who opened his parcel and found an old newspaper clipping. What about the sunglasses? And the revolver? Probably illegal, by the way—no license. And the diamond—it must have been paste, it was so big."

"M-mm," Mr. Talley said.

"I'm no smart apple, but I can smell a screwy set-up. Most of your clients are big shots, in one way or another. And why didn't any of 'em pay you, like the first man—the guy who came in when I was here this morning."

"It's chiefly a credit business," Talley said. "I've my ethics. I have to—for my own conscience. It's responsibility. You see, I sell . . . my goods . . . with a guarantee. Payment is made only if the product proves satisfactory."

"So. An egg. Sunglasses. A pair of asbestos gloves—I think they were. A newspaper clipping. A gun and a diamond. How do you take inventory?"

Talley said nothing.

Carmichael grinned. "You've an errand boy. You send him out and he comes back with bundles. Maybe he goes to a grocery on Madison and buys an egg. Or a pawnshop on Sixth for a revolver. Or—well, anyhow, I told you I'd find out what your business is."

"And have you?" Talley asked.

"'We have what you need,'" Carmichael said. "But how do you *know*?"

"You're jumping to conclusions."

"I've got a headache—I didn't have sunglasses!—and I don't believe in magic. Listen, Mr. Talley. I'm fed up to the eyebrows and 'way beyond on queer little shops that sell peculiar things. I know too much about 'em—I've written about 'em. A guy walks along the street and sees a funny sort of store and the proprietor won't serve him—he sells only to pixies—or else he *does* sell him a magic charm with a double edge. Well—*pfui!*"

"Mph," Talley said.

"'Mph' as much as you like. But you can't get away from logic. Either you've got a sound, sensible racket here, or else it's one of those funny magic-shops set-ups—and I don't believe that. For it isn't logical."

"Why not?"

"Because of economics," Carmichael said flatly. "Grant the idea that you've got certain mysterious powers—let's say you can make telepathic gadgets. All right. Why the devil would you start a business so you could sell the gadgets so you could make money so you could live? You'd simply put on one of your gadgets, read a stockbroker's mind, and buy the right stocks. That's the intrinsic fallacy in these crazy-shop things—if you've got enough stuff on the ball to be able to stock and run such a shop, you wouldn't need a business in the first place. Why go round Robin Hood's barn?"

Talley said nothing.

Carmichael smiled crookedly. "'I often wonder what the vintners buy one half so precious as the stuff they sell,'" he quoted. "Well—what do *you* buy? I know what you sell—eggs and sunglasses."

"You're an inquisitive man, Mr. Carmichael," Talley murmured. "Has it ever occurred to you that this is none of your business?"

"I may be a customer," Carmichael repeated. "How about that?"

Talley's cool blue eyes were intent. A new light dawned in them; Talley pursed his lips and scowled. "I hadn't thought of that," he admitted. "You might be. Under the circumstances. Will you excuse me for a moment?"

"Sure," Carmichael said. Talley went through the curtains.

Outside, traffic drifted idly along Park. As the sun slid down beyond the Hudson, the street lay in a blue shadow that crept imperceptibly up the barricades of the buildings. Carmichael stared at the sign—"We have what you need"—and smiled.

In a back room, Talley put his eye to a binocular plate and moved a calibrated dial. He did this several times. Then, biting his lip—for he was a gentle man—he called his errand boy and gave him directions. After that he returned to Carmichael.

"You're a customer," he said. "Under certain conditions."

"The condition of my bank account, you mean?"

"No," Talley said. "I'll give you reduced rates. Understand one thing. I really do have what you need. You don't *know* what you need, but I know. And as it happens—well, I'll sell you what you need for, let's say, five dollars."

Carmichael reached for his wallet. Talley held up a hand.

"Pay me after you're satisfied. And the money's the nominal part of the fee. There's another part. If you're satisfied, I want you to promise that you'll never come near this shop again and never mention it to anyone."

"I see," Carmichael said slowly. His theories had changed slightly.

"It won't be long before . . . ah, here he is now." A buzzing from the back indicated the return of the errand boy. Talley said "Excuse me," and vanished. Soon he returned with a neatly-wrapped parcel, which he thrust into Carmichael's hands.

"Keep this on your person," Talley said. "Good afternoon."

Carmichael nodded, pocketed the parcel, and went out. Feeling affluent, he hailed a taxi and went to a cocktail bar he knew. There, in the dim light of a booth, he unwrapped the bundle.

Protection money, he decided. Talley was paying him off to keep his mouth shut about the racket, whatever it was. O.K. live and let live. How much would be—

Ten thousand? Fifty thousand? How big was the racket?

He opened an oblong cardboard box. Within, nesting upon tissue paper, was a pair of shears, the blades protected by a sheath of folded, glued cardboard.

Carmichael said something softly. He drank his highball and ordered another, but left it untasted. Glancing at his wrist watch, he decided that the Park Avenue shop would be closed by now and Mr. Peter Talley gone.

"'One half so precious as the stuff they sell,'" Carmichael said. "Maybe it's the scissors of Atropos. Blah." He unsheathed the blades and snipped experimentally at the air. Nothing happened. Slightly crimson around the cheekbones, Carmichael reholstered the shears and dropped them into the side pocket of his topcoat. Quite a gag!

He decided to call on Peter Talley tomorrow.

Meanwhile, what? He remembered he had a dinner date with one of the girls at the office, and hastily paid his bill and left. The streets were darkening, and a cold wind blew southward from the Park. Carmichael wound his scarf tighter around his throat and made gestures toward passing taxis.

He was considerably annoyed.

Half an hour later a thin man with sad eyes—Jerry Worth, one of the copy-writers from his office—greeted him at the bar where Carmichael was killing time. "Waiting for Betsy?" Worth said, nodding toward the restaurant annex. "She sent me to tell you she couldn't make it. A rush deadline. Apologies and stuff. Where were you today? Things got gummed up a bit. Have a drink with me."

They worked on rye. Carmichael was already slightly stiff. The dull crimson around his cheekbones had deepened, and his frown had become set. "What you need," he remarked. "Double-crossing little—"

"Huh?" Worth said.

"Nothing. Drink up. I've just decided to get a guy in trouble. If I can."

"You almost got in trouble yourself today. That trend analysis of ores—"

"Eggs. Sunglasses!"

"I got you out of a jam—"

"Shut up," Carmichael said and ordered another round. Every time he felt the weight of the shears in his pocket he found his lips moving.

Five shots later Worth said plaintively, "I don't mind doing good deeds but I do like to mention them. And you won't let me. All I want is a little gratitude."

"All right, mention them," Carmichael said. "Brag your head off. Who cares?"

Worth showed satisfaction. "That ore analysis—it was that. You weren't at the office today, but I caught it. I checked with our records and you had Trans-Steel all wrong. If I hadn't altered the figures, it would have gone down to the printer—"

"What?"

"The Trans-Steel. They—"

"Oh, you fool," Carmichael groaned. "I know it didn't check with the office figures. I meant to put in a notice to have them changed. I got my dope from the source. Why don't you mind your own business?"

Worth blinked. "I was trying to help."

"It would have been good for a five-buck raise," Carmichael said. "After all the research I did to uncover the real dope—listen. Has the stuff gone to bed yet?"

"I dunno. Maybe not. Croft was still checking the copy—"

"O.K.!" Carmichael said. "Next time—" He jerked at his scarf, jumped off the stool, and headed for the door, trailed by the protesting Worth. Ten minutes later he was at the office, listening to Croft's bland explanation that the copy had already been dispatched to the printer.

"Does it matter? Was there . . . incidentally, where were you today?"

"Dancing on the rainbow," Carmichael snapped, and departed. He had switched over from rye to whiskey sours, and the cold night air naturally did not sober him. Swaying slightly, watching the sidewalk move a little as he blinked at it, he stood on the curb and pondered.

"I'm sorry, Tim," Worth said. "It's too late now, though. There won't be any trouble. You've got a right to go by our office records."

"Stop me now," Carmichael said. "Lousy little—" He was angry and drunk. On impulse he got another

138 ASTOUNDING SCIENCE-FICTION

WHAT YOU NEED 139

taxi and sped to the printers, still trailing a somewhat confused Jerry Worth.

There was rhythmic thunder in the building. The swift movement of the taxi had given Carmichael a slight nausea; his head ached, and alcohol was in solution in his blood. The hot, inky air was unpleasant. The great Linotypes thumped and growled. Men were moving about. It was all slightly nightmarish, and Carmichael doggedly hunched his shoulders and lurched on until something jerked him back and began to strangle him.

Worth started yelling. His face showed drunken terror. He made ineffectual gestures.

But this was all part of the nightmare. Carmichael saw what had happened. The ends of his scarf had caught in moving gears somewhere and he was being drawn in-

exorably into meshing metal cogs. Men were running. The clanking, thumping, rolling sounds were deafening. He pulled at the scarf.

Worth screamed. ". . . knife! Cut it—"

The warping of relative values that intoxication gives saved Carmichael. Sober, he would have been helpless with panic. As it was, each thought was hard to capture, but clear and lucid when he finally got it. He remembered the shears, and he put his hand in his pocket—the blades slipped out of their cardboard sheath—and he snipped through the scarf with fumbling, hasty movements.

The white silk disappeared. Carmichael fingered the ragged edge at his throat and smiled stiffly.

Mr. Peter Talley had been hoping that Carmichael would not come back. The probability lines had shown two possible variants; in one, all was well; in the other—

Carmichael walked into the shop the next morning and held out a five-dollar bill. Talley took it.

"Thank you. But you could have mailed me a check."

"I could have. Only that wouldn't have told me what I wanted to know."

"No," Talley said, and sighed. "You've decided, haven't you?"

"Do you blame me?" Carmichael asked. "Last night—do you know what happened?"

"Yes."

"How?"

"I might as well tell you," Talley said. "You'd find out anyway. That's certain, anyhow."

Carmichael sat down, lit a cigarette, and nodded. "Logic. You couldn't have arranged that little accident, by any manner of means. Betsy Hoag decided to break our date early yesterday morning. Before I saw you. That was the beginning of the chain of incidents that led up to the accident. Ergo, you must have known what was going to happen."

"I did know."

"Prescience?"

"Mechanical. I saw that you would be crushed in the machine—"

"Which implies an alterable future."

"Certainly," Talley said, his shoulders slumping. "There are innumerable possible variants to the future. Different lines of probability. All depending on the outcome of various crises as they arise. I happen to be skilled in certain branches of electronics. Some years ago, almost by accident, I stumbled on the principle of seeing the future."

"How?"

"Chiefly it involves a personal focus on the individual. The moment you enter this place"—he gestured—"you're in the beam of my scanner. In my back room I have the machine itself. By turning a caliberated dial, I check the possible futures. Sometimes there are many. Sometimes only a few. As though at times certain stations weren't broadcasting. I look into my scanner and see what you need—and supply it."

Carmichael let smoke drift from

his nostrils. He watched the blue coils through narrowed eyes.

"You follow a man's whole life —in triplicate or quadruplicate or whatever?"

"No," Talley said. "I've got my device focused so it's sensitive to crisis curves. When those occur, I follow them farther and see what probability paths involve the man's safe and happy survival."

"The sunglasses, the egg and the gloves—"

Talley said, "Mr. . . . uh . . . Smith is one of my regular clients. Whenever he passes a crisis successfully, with my aid, he comes back for another checkup. I locate his next crisis and supply him with what he needs to meet it. I gave him the asbestos gloves. In about a month, a situation will arise where he must—under the circumstances —moves a red-hot bar of metal. He's an artist. His hands—"

"I see. So it isn't always saving a man's life."

"Of course not," Talley said. "Life isn't the only vital factor. An apparently minor crisis may lead to —well, a divorce, a neurosis, a wrong decision, and the loss of hundreds of lives indirectly. I insure life, health, and happiness."

"You're an altruist. Only why doesn't the world storm your doors? Why limit your trade to a few?"

"I haven't got the time or the equipment."

"More machines could be built."

"Well," Talley said, "most of my customers are wealthy. I must live."

"You could read tomorrow's stock-market reports if you wanted dough," Carmichael said. "We get back to that old question. If a guy has miraculous powers, why is he satisfied to run a hole-in-the-wall store?"

"Economic reasons. I . . . ah . . . I'm averse to gambling."

"It wouldn't be gambling," Carmichael pointed out. " 'I often wonder what the vintners buy—' Just what *do* you get out of this?"

"Satisfaction," Talley said, "Call it that."

But Carmichael wasn't satisfied. His mind veered from the question and turned to the possibilities. Insurance, eh? Life, health, and happiness.

"What about me? Won't there be another crisis in my life sometime?"

"Probably. Not necessarily one involving personal danger."

"Then I'm a permanent customer."

"I . . . don't—"

"Listen," Carmichael said, "I'm not trying to shake you down. I'll pay. I'll pay plenty. I'm not rich, but I know exactly what a service like this would be worth to me. No worries—"

"It wouldn't be—"

"Oh, come off it. I'm not a blackmailer or anything. I'm not threatening you with publicity, if that's what you're afraid of. I'm an ordinary guy. Not a melodramatic villain. Do I look dangerous? What are you afraid of?"

"You're an ordinary guy, yes," Talley admitted. "Only—"

"Why not?" Carmichael argued. "I won't bother you. I passed one crisis successfully, with your help. There'll be another one due sometime. Give me what I need for that. Charge me anything you like. I'll get the dough somehow. Borrow it if necessary. I won't disturb you at all. All I ask is that you let me come in whenever I've passed a crisis, and get ammunition for the next one. What's wrong with that?"

"Nothing," Talley said soberly.

"Well, then. I'm an ordinary guy. There's a girl—it's Betsy Hoag. I want to marry her. Settle down somewhere in the country, raise kids, and have security. There's nothing wrong with that either, is there?"

Talley said, "It was too late the moment you entered this shop today."

Carmichael looked up. "Why?" he asked sharply.

A buzzer rang in the back. Talley went through the curtains and came back almost immediately with a wrapped parcel. He gave it to Carmichael.

Carmichael smiled. "Thanks," he said. "Thanks a lot. Do you have any idea when my next crisis will come?"

"In a week."

"Mind if I—" Carmichael was unwrapping the package. He took out a pair of plastic-soled shoes and looked at Talley, bewildered.

"Like that, eh? I'll need—shoes?"

"Yes."

"I suppose—" Carmichael hesitated. "I guess you wouldn't tell me why?"

"No, I won't do that. But be sure to wear them whenever you go out."

"Don't worry about that. And—I'll mail you a check. It may take me a few days to scrape up the dough, but I'll do it. How much—?"

"Five hundred dollars."

"I'll mail a check today."

"I prefer not to accept a fee until the client has been satisfied," Talley said. He had grown more reserved, his blue eyes cool and withdrawn.

"Suit yourself," Carmichael said. "I'm going out and celebrate. You—don't drink?"

"I can't leave the shop."

"Well, good-by. And thanks again. I won't be any trouble to you, you know. I promise that!" He turned away.

Looking after him, Talley smiled a wry, unhappy smile. He did not answer Carmichael's good-by. Not then.

When the door had closed behind him, Talley turned to the back of his shop and went through the door where the scanner was.

The lapse of ten years can cover a multitude of changes. A man with the possibility of tremendous power almost within his grasp can alter, in that time, from a man who will not reach for it to a man who will—and moral values be damned.

The change did not come quickly to Carmichael. It speaks well for his integrity that it took ten years to work such an alteration in all he had been taught. On the day he first went into Talley's shop there was little evil in him. But the temptation grew stronger week by week, visit by visit. Talley, for reasons of his own, was content to sit idly by, waiting for customers, smothering the inconceivable potentialities of his machine under a blanket of trivial functions. But Carmichael was not content.

It took him ten years to reach the day, but the day came at last.

Talley sat in the inner room, his back to the door. He was slumped low in an ancient rocker, facing the machine. It had changed little in the space of a decade. It still covered most of two walls, and the eyepiece of its scanner glittered under amber fluorescents.

Carmichael looked covetously at the eyepiece. It was window and doorway to a power beyond any man's dreams. Wealth beyond imagining lay just within that tiny opening. The rights over the life and death of every man alive. And nothing between that fabulous future and himself except the man who sat looking at the machine.

Talley did not seem to hear the careful footsteps or the creak of the door behind him. He did not stir as Carmichael lifted the gun slowly. One might think that he never guessed what was coming, or why, or from whom, as Carmichael shot him through the head.

Talley sighed and shivered a little, and twisted the scanner dial. It was not the first time that the eyepiece had shown him his own lifeless body, glimpsed down some vista of probability, but he never saw the slumping of that familiar figure without feeling a breath of indescribable coolness blow backward upon him out of the future.

He straightened from the eyepiece and sat back in his chair, looking thoughtfully at a pair of rough-soled shoes lying beside him on a table. He sat quietly for awhile, his eyes upon the shoes, his mind following Carmichael down the street and into the evening, and the morrow, and on toward that coming crisis which would depend on his secure footing on a subway platform as a train thundered by the place where Carmichael would be standing one day next week.

Talley had sent his messenger boy out this time for two pairs of shoes. He had hesitated long, an hour ago, between the rough-soled pair and the smooth. For Talley was a humane man, and there were many times when his job was distasteful to him. But in the end, this time, it had been the smooth-soled pair he had wrapped for Carmichael.

Now he sighed and bent to the scanner again, twisting the dial to bring into view a scene he had watched before.

Carmichael, standing on a crowded subway platform, glittering with oily wetness from some overflow. Carmichael, in the slick-soled shoes Talley had chosen for him. A commotion in the crowd, a surge toward the platform edge. Carmichael's feet slipping frantically as the train roared by.

"Good-by, Mr. Carmichael," Talley murmured. It was the farewell he had not spoken when Carmichael left the shop. He spoke it regretfully, and the regret was for the Carmichael of today, who did not yet deserve that end. He was not now a melodramatic villain whose death one could watch unmoved. But the Tim Carmichael of today had atonement to make for the Carmichael of ten years ahead, and the payment must be exacted.

It is not a good thing to have the power of life and death over one's fellow humans. Peter Talley knew it was not a good thing—but the power had been put into his hands. He had not sought it. It seemed to him that the machine had grown almost by accident to its tremendous completion under his trained fingers and trained mind.

At first it had puzzled him. How ought such a device to be used? What dangers, what terrible potentialities, lay in that Eye that could see through the veil of tomorrow? His was the responsibility, and it had weighed heavily upon him until the answer came. And after he knew the answer—well, the weight was heavier still. For Talley was a mild man.

He could not have told anyone the real reason why he was a shopkeeper. Satisfaction, he had said to Carmichael. And sometimes, indeed, there was deep satisfaction. But at other times—at times like this—there was only dismay and humility. Especially humility.

We have what you need. Only Talley knew that message was not for the individuals who came to his shop. The pronoun was plural, not singular. It was a message for the world—the world whose future was being carefully, lovingly reshaped under Peter Talley's guidance.

The main line of the future was not easy to alter. The future is a pyramid shaping slowly, brick by brick, and brick by brick Talley had to change it. There were some men who were necessary—men who would create and build—men who should be saved.

Talley gave them what they needed.

But inevitably there were others whose ends were evil. Talley gave them, too, what the world needed—death.

Peter Talley had not asked for this terrible power. But the key had been put in his hands, and he dared not delegate such authority as this to any other man alive. Sometimes he made mistakes.

He had felt a little surer since the simile of the key had occurred to him. The key to the future. A key that had been laid in his hands.

Remembering that, he leaned back in his chair and reached for an old and well-worn book. It fell open easily at a familiar passage. Peter Talley's lips moved as he read the passage once again, in his room behind the shop on Park Avenue.

"And I say also unto thee. That thou art Peter— And I will give unto thee the keys of the kingdom of heaven—"

THE END.