Placer++: Semantic Place Labels Beyond the Visit

John Krumm, Dany Rouhana, Ming-Wei Chang
Microsoft Research
Microsoft Corporation
Redmond, WA USA

***Abstract*— Place labeling is the process of giving semantic labels to locations, such as home, work, and school. For a particular person, these labels can be computed automatically based on features of that person's visits to these locations. A previous system called Placer used the person's demographic data and the timing of their visits to label places with a learned decision tree. We developed Placer++ as a more accurate labeler, augmenting Placer's features of individual visits with (1) labeled visits from other people and (2) features about the sequence of the individual's visits. In processing sequences, we adopt structural learning techniques to take into account the relationships between visits. Accuracy increased by 8.85 percentage points over the baseline of Placer. We describe and justify the features and present our experiments on government diary data.**

Keywords*— *Semantic place labels, locations, PSRC

I. Introduction

As more of our digital life becomes mobile, it becomes more important to understand the places we go. One aspect of this understanding is place labeling, which gives semantic names to locations. We are interested in egocentric place labels for individuals, such as a person's home, work, and school. Labels like these are useful for automatically generating comprehensible text, such as "Sally has just arrived at school." They can also be used to annotate a diary of a person's activities. Consolvo *et al.* [1] have shown that when people want to reveal their location, they prefer to use either a street address or a label like home or work. Labels are also useful for triggering automatic behaviors (*e.g.* only certain calls ring through at work) and inferring higher-level activities (*e.g.* watching television at home is more likely than at work).

Place labeling usually begins with a sequence of time stamped location measurements, such as from GPS, WiFi, or cell towers. The first step in processing is to find places where the user spends time. This is fundamentally a process of clustering, and has been the subject of previous research [2-9]. We assume clustering is already complete, and we assume we can accurately detect visits to these places based on location measurements. In this paper we concentrate on the second main step, which is to attach labels to the clusters based on features such as the timing of the visit, nearby businesses, previous & next visits, and labels given by others.

There are multiple research prototypes that allow users to manually give place names to the places they go, such as Reno [10], Connecto [11], and IMBuddy [12]. Google Now™, a commercial product, lets users manually label their home and work locations. We are interested in automatic labeling of places. It is possible to build heuristics to generate labels, such as Zhu *et al.* [13]. They created hand-crafted rules to compare against machine learning. As an example, one rule classified home as the place where a user spent most of their time between midnight and 6 a.m. Heuristics can fail, however, for users whose habits don't fit our notions of where people normally spend their time, such as night workers.

Another approach to place labeling is machine learning. This is a promising method, because it automatically accounts for the way people really behave. The primary limitation is the availability of training data. There was a surge of machine learning techniques aimed at place labeling in response to the Nokia Mobile Data Challenge (MDC) [14]. The MDC provided labeled visit sequences and cell phone logs for 114 people (80 for training, 34 for testing) with an average of 282 days of observation for each one. The cell phone logs contained information about calls, text messages, cell towers, accelerometer samples, Bluetooth & WiFi observations, and other phone activity. The four teams that took on the labeling challenge all used machine learning to find a mapping from the phone features, including the time and duration of visits, to the 10 different place labels [13, 15-17]. Their labeling accuracies were between 65% and 75%. An earlier approach to machine learned place labeling was the work of Liao *et al.* [18]. Their features included the locations of nearby restaurants, grocery stores and bus stops as well as the timing of visits. Testing on four people, they used a conditional random field to exploit the fact that visits to places often occur in a consistent order, such as going home directly after work. Chen *et al.* [19] used a hidden Markov model to take advantage of this ordering. Using almost 6000 users from Whrrl, a location-based social network, Ye *et al.* [20] used a support vector machine to label eight different types of places. Their features included check-in frequency and time of day. Microsoft's Cortana® personal assistant uses a proprietary algorithm to automatically find a user's home and work locations from location data.

Researchers have also shown that place labels can be derived from labels left by others. For example, if one person designates a place as a restaurant, other people can use that label. Loci [21] included a user study that showed that participants in a location based service are willing to provide place labels for others to use. CenceMe [8] augmented a user's place labels with labels left by the user's friends. Getting Places [22] showed how to use collaborative filtering from many people to give useful place labels. In [23], Ames and Naaman examine the motivations for users to tag photos, finding that sharing tags is an important goal. Labeling places may benefit from a similar motivation.

Previous approaches to place labeling thus fall into four categories: manual labeling, heuristics, machine learning, and others' labels. The last two are the most promising, and we incorporate both into our new place-labeling technique called

Placer++. Our technique features two main innovations. The first is that we use machine learning to examine the whole sequence of a person's visits to improve labeling accuracy. For instance, one of our sequence features looks at the co-occurrence of place visits and exploits the fact that people who spend time at a place labeled "college" never spend significant time at a place labeled "childcare". The other innovation is a machine learning approach to use place labels given by other people. This is possible for us since we use labeled data from almost 10,000 people concentrated in a single metro area, giving a dense set of labels that can be exploited for labeling places of new people.

Placer++ builds on a previous technique called Placer [6], which we describe next.

II. From Placer to Placer++

Placer [6] is a place labeling technique based on machine learning. One of the two diary studies it used is the 2006 Puget Sound Regional Council (PSRC) Household Activity Survey [24]. PSRC participants were from the Puget Sound region in the U.S., which consists of four counties near Seattle, WA. Participants filled out a survey covering their visits over two consecutive days. The PSRC diary data includes latitude/longitude data for the visits. Placer showed that features derived from latitude/longitude (such as nearby businesses) significantly increased labeling accuracy.

The PSRC diary study includes data from 86,764 trips taken by 9790 different people who gave labels to 18,888 distinct places. For each participant, PSRC gives their gender and age. The data also indicates which people are in the same household. For each visit to a place by each participant, PSRC gives the arrival date and time, the duration, and the location's latitude/longitude. Each visit is also labeled with a primary activity from the list of 17 activities/places (Table 1), which the Placer paper shortened to 12 by combining similar place types (Figure 5).

Placer used a feature vector of 69 scalar features to classify each visit. One subset of the features characterized the subject (*i.e.* age and gender) and the timing of the visit *(e.g.* day of week, arrival time, and duration). The other subset characterized nearby businesses. We will add to these features subsequently for Placer++.

Placer used a forest of boosted decision trees [25] for classification. The trees are learned based on training data. After learning an initial decision tree, a second decision tree is constructed where more weight has been given to the mistaken training samples of the first tree. This continues up to a prespecified maximum number of trees, which was 100 for Placer. Each tree had a maximum branching factor of 20, at least 10 instances per leaf, and a learning rate of 0.2. We used the same learning procedure for Placer++. In our reimplementation of Placer, the overall classification accuracy was 63.85%, which is the fraction of visits that were classified into the correct place label. The original Placer paper gave an accuracy of 74.0% on this dataset. The reduced accuracy of our reimplementation is due to three factors:

1) Because the original business database for Placer was unavailable, we used a different database for our reimplementation.

2) Our single evaluation on a train/test split of 80%/20% gave less training data than the 90%/10% 10-fold cross validation used for Placer. We used the 80%/20% split in this paper to match the split we used for testing Placer++.

3) For Placer++, we were careful to disallow place visits from the same household to be split across training and testing, while Placer's cross validation allowed this split, which can increase accuracy.

Placer++ introduces a few hundred new classification features that Placer did not have. More importantly, Placer++ introduces the idea of inferring labels based on features of a whole sequence of visits rather than just features of a single visit itself. This requires a new algorithmic approach that was not possible in the Placer machine learning framework. The new approach is described in Section IV, but before that we justify the addition of new features by looking at the data itself in the next section.

III. Evidence for Additional Signal

The original Placer used features about the demographics of the user, the timing of the visits, and nearby businesses to give a place label for each visit. There are other signals to exploit beyond the isolated visit. This section contains an examination of the PSRC data to show how people behave with respect to the places they go. We explain how we exploit these behaviors for Placer++ later in this paper.

A. Cross Labels

One advantage of using the PSRC data is that the labeled places cover a fairly small area, so we can expect to see instances of different people labeling the same place. It has been shown by Loci [21], CenceMe [8], and Getting Places [22] that exploiting labels given by other people can be effective. We can imagine a mobile phone application that shows a user their place label inferences and lets them confirm or correct the labels. In this way, a central database could build up a table of manually labeled places. These labels, which we call "cross labels," could influence the inferred label of a new visitor.

In our PSRC data, we have 18,888 distinct labeled places. Of these, 9074 (48.0%) have labels from more than one person. Ideally, all the labels at a given location would be the same. However, of the places with at least two visit labels, only 4065 (21.5%) have identical labels from all their visitors. We can assess the "purity" of each label type by looking at how often places are given an unmixed set of labels. For this analysis, we looked at every location in the PSRC data set that had been labeled at least twice. Then for each label, we counted the number of distinct places where the label appears. The purity of the label is the ratio of distinct places that have *only* that label to the total number of places with that label. The results are shown in Figure 1. For example, of all the distinct places labeled as "Home – Other", about 67% had only that label. The most pure places are "Home – Other", "Eat Out", and "Work", but even

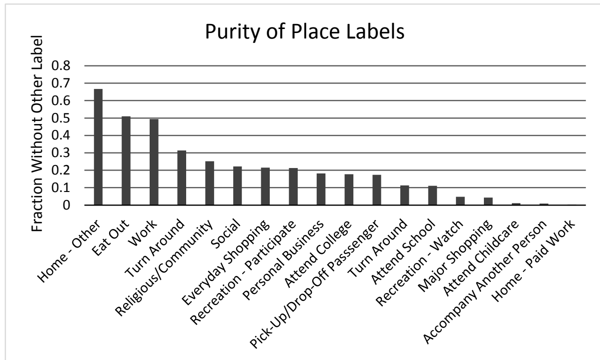

Figure 1: This shows the purity of place labels. For instance, of all the distinct places labeled "Home – Other", about 67% had only that label and no others.

"Work" is below 50%. This means that many places are given different labels, so using labels from other people is not entirely reliable. One reason for the multiple labels is the particular taxonomy of places used by PSRC. For instance, "Major Shopping" and "Everyday Shopping" could easily refer to the same place depending on the goals of the visitor. Another reason for this lack of purity is that different places mean different things to different people. For instance, a restaurant can be a place to eat or a place to work.

We can understand more about mixed labels by looking at common combinations of labels. For each location with at least two labels, we computed the most common combinations of 2, 3, 4, and 5 different labels, shown in Figure 2. The most common 2-label place has labels of "Home – Other" and "Personal Business". The "Personal Business" visits could be tasks like dropping off or picking up something. For places with four different labels, the most common combination was "accompany another person", "personal business", "major shopping", and "everyday shopping". A mall with several stores could easily accommodate all four labels.

A simple algorithm for labeling a visit would be to give it the most frequent label given by others who have visited it before. Of the 86,764 visits in the PSRC data with a latitude/longitude, 82.1% could be labeled this way. Of these, this algorithm would give the correct label to 36.3% of the visits. While this is not a high rate of accuracy by itself, it suggests that features involving cross label data could be helpful.

B. Sequences

We suspect that people maintain some regularity in their sequence of visits. For instance, a given person might usually go directly home after work. The Placer paper gave Markov probabilities giving the probability of the next place type given the current place type. This analysis showed that the next most likely place to go was always home, no matter where a person was currently, unless they were already home. We use this regularity as a feature for Placer++.

We might also expect people's visits often have a 24-hour period. If a person is at home at 3 a.m. on one day, it's likely they will be at home 24 hours later. We can test this assumption with the PSRC data. Given a place at one time, we can compute the probabilities of being at any place 24 hours later. Specifically, we looked at the temporal midpoint of each visit in the PSRC data, along with its place label. For each such midpoint, we found the place label of the visit occurring 24 hours later, if it was available. (Sometimes the visit was missing, and sometimes we moved beyond the end of the survey period.) Table 1 shows the results. Given a place label in the left column, the corresponding row shows the probabilities of being at any other place label 24 hours later. The strongest 24-hour periodicity is for "attend school", where there is a 91% chance of being at school 24 hours after previously being at school. "Home - Other", work, and childcare are also strongly periodic. There is also some structure in the off-diagonal area, where being home is likely 24 hours after most of the less periodic activities.

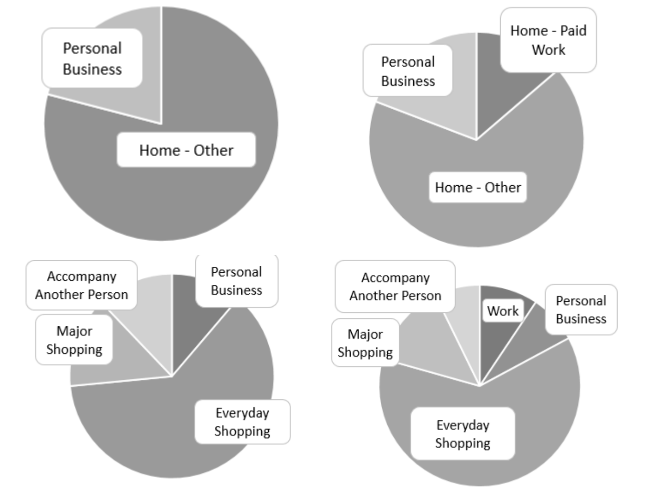

Figure 2: These are the most common combinations of labels for places with 2, 3, 4, or 5 different labels.

C. Place Cooccurrences

We might expect different lifestyles to be reflected in the places people go. Specifically, it may be that trips to one type of

Given Place in Sequence \ Place 24 Hours Later	Home - Paid Work	Home - Other	Work	Attend Childcare	Attend School	Attend College	Eat Out	Personal Business	Everyday Shopping	Major Shopping	Religious/Community	Social	Recreation - Participate	Recreation - Watch	Accompany Another Person	Pick-Up/Drop-Off Passsenger	Turn Around
Home - Paid Work	0.38	0.46	0.07	0.00	0.00	0.00	0.01	0.03	0.01	0.00	0.01	0.01	0.00	0.01	0.01	0.00	0.00
Home - Other	0.01	0.82	0.03	0.00	0.00	0.00	0.00	0.07	0.01	0.00	0.00	0.01	0.01	0.00	0.02	0.00	0.00
Work	0.01	0.11	0.82	0.00	0.00	0.00	0.01	0.02	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00
Attend Childcare	0.00	0.06	0.00	0.73	0.04	0.00	0.00	0.04	0.00	0.00	0.00	0.02	0.03	0.01	0.05	0.01	0.00
Attend School	0.00	0.03	0.00	0.00	0.91	0.00	0.00	0.03	0.00	0.00	0.00	0.01	0.01	0.00	0.01	0.00	0.00
Attend College	0.00	0.24	0.06	0.00	0.00	0.61	0.01	0.01	0.00	0.00	0.00	0.01	0.00	0.01	0.01	0.00	0.00
Eat Out	0.01	0.42	0.20	0.00	0.02	0.01	0.11	0.08	0.03	0.01	0.01	0.04	0.04	0.02	0.01	0.01	0.00
Personal Business	0.01	0.46	0.12	0.00	0.01	0.01	0.02	0.24	0.03	0.01	0.01	0.02	0.03	0.01	0.01	0.01	0.00
Everyday Shopping	0.02	0.54	0.14	0.00	0.01	0.01	0.03	0.08	0.06	0.01	0.02	0.03	0.04	0.01	0.01	0.01	0.00
Major Shopping	0.01	0.49	0.14	0.00	0.00	0.00	0.03	0.13	0.04	0.02	0.01	0.03	0.04	0.00	0.03	0.01	0.00
Religious/Community	0.02	0.55	0.06	0.00	0.00	0.01	0.04	0.06	0.03	0.01	0.12	0.04	0.03	0.01	0.02	0.01	0.00
Social	0.01	0.52	0.06	0.00	0.01	0.01	0.02	0.05	0.03	0.01	0.02	0.19	0.04	0.01	0.03	0.01	0.00
Recreation - Participate	0.01	0.45	0.08	0.01	0.01	0.00	0.02	0.09	0.03	0.00	0.01	0.03	0.22	0.01	0.03	0.00	0.00
Recreation - Watch	0.01	0.52	0.09	0.01	0.02	0.00	0.02	0.09	0.03	0.00	0.00	0.05	0.05	0.05	0.03	0.02	0.00
Accompany Another Person	0.00	0.37	0.04	0.03	0.03	0.00	0.01	0.11	0.01	0.00	0.01	0.03	0.04	0.01	0.30	0.01	0.00
Pick-Up/Drop-Off Passsenger	0.01	0.44	0.11	0.00	0.01	0.00	0.01	0.06	0.02	0.00	0.01	0.02	0.02	0.02	0.02	0.23	0.00
Turn Around	0.03	0.50	0.13	0.00	0.02	0.00	0.02	0.05	0.01	0.00	0.02	0.03	0.03	0.01	0.02	0.01	0.08

Table 1: Given a place type, there is regularity in the place type 24 hours later. For each place type in the left column, this table shows the probabilities of visits to place types 24 hours later.

Given Place in Sequence \ Dependent Place	Home - Paid Work	Home - Other	Work	Attend Childcare	Attend School	Attend College	Eat Out	Personal Business	Everyday Shopping	Major Shopping	Religious/Community	Social	Recreation - Participate	Recreation - Watch	Accompany Another Person	Pick-Up/Drop-Off Passsenger	Turn Around
Home - Paid Work	1.00	0.92	0.68	0.00	0.00	0.01	0.36	0.57	0.48	0.06	0.08	0.14	0.24	0.09	0.08	0.32	0.06
Home - Other	0.04	1.00	0.46	0.02	0.15	0.03	0.28	0.47	0.38	0.05	0.07	0.17	0.23	0.08	0.13	0.23	0.04
Work	0.06	0.99	1.00	0.00	0.01	0.03	0.30	0.41	0.39	0.04	0.05	0.13	0.18	0.07	0.05	0.24	0.04
Attend Childcare	0.00	1.00	0.00	1.00	0.23	0.00	0.18	0.50	0.09	0.02	0.01	0.17	0.30	0.08	0.48	0.22	0.01
Attend School	0.00	1.00	0.03	0.03	1.00	0.00	0.17	0.45	0.12	0.01	0.06	0.19	0.33	0.07	0.28	0.17	0.02
Attend College	0.02	0.99	0.45	0.00	0.01	1.00	0.25	0.37	0.37	0.06	0.05	0.17	0.17	0.04	0.07	0.20	0.03
Eat Out	0.05	1.00	0.50	0.01	0.09	0.02	1.00	0.63	0.52	0.08	0.08	0.24	0.29	0.12	0.14	0.31	0.05
Personal Business	0.05	0.98	0.40	0.02	0.14	0.02	0.37	1.00	0.48	0.07	0.09	0.18	0.27	0.09	0.13	0.27	0.04
Everyday Shopping	0.05	1.00	0.47	0.01	0.05	0.03	0.39	0.61	1.00	0.08	0.09	0.21	0.27	0.09	0.09	0.30	0.05
Major Shopping	0.05	1.00	0.39	0.01	0.03	0.03	0.45	0.64	0.58	1.00	0.08	0.24	0.27	0.09	0.14	0.32	0.06
Religious/Community	0.05	1.00	0.38	0.00	0.13	0.02	0.35	0.61	0.53	0.06	1.00	0.23	0.25	0.06	0.13	0.32	0.05
Social	0.03	0.98	0.36	0.02	0.17	0.03	0.39	0.52	0.47	0.07	0.09	1.00	0.28	0.10	0.16	0.29	0.06
Recreation - Participate	0.04	0.98	0.35	0.03	0.21	0.02	0.34	0.54	0.43	0.06	0.07	0.20	1.00	0.09	0.17	0.28	0.06
Recreation - Watch	0.05	0.99	0.43	0.02	0.14	0.01	0.43	0.56	0.44	0.06	0.06	0.23	0.29	1.00	0.18	0.46	0.06
Accompany Another Person	0.03	1.00	0.17	0.09	0.33	0.01	0.32	0.49	0.27	0.06	0.07	0.21	0.32	0.10	1.00	0.30	0.04
Pick-Up/Drop-Off Passsenger	0.06	1.00	0.49	0.02	0.11	0.02	0.38	0.57	0.49	0.07	0.10	0.22	0.29	0.15	0.17	1.00	0.06
Turn Around	0.07	0.99	0.44	0.01	0.09	0.02	0.40	0.54	0.54	0.08	0.09	0.27	0.37	0.12	0.14	0.36	1.00

Table 2: Given a place type in the left column, this shows the probability of visiting other place types. As an example, given a visit to "Attend Childcare", the probability of also visiting "Attend College" is zero.

place affect the likelihood of going to other types of places. For instance, people who go to childcare likely don't also go to college. We can quantify this by looking at conditional probabilities of the form $p(\text{place 2}|\text{place 1})$, which is the probability of visiting place type 2 given that the person has visited a place type 1. Table 2 gives estimates of these probabilities from the PSRC data. These were computed by looking at the set of places visited by each person in the study. Indeed, given a visit to "Attend Childcare", the probability of also visiting "Attend College" is zero. (Note that "attend" here means more than just a pick-up or drop-off. While a childcare attendee may visit a college briefly, the PSRC data says they don't spend enough time at college to qualify as a full visit.) We also see that the conditional probability of "Eat Out" is maximized by people who also did "Major Shopping". In addition, people who "Work" never "Attend Childcare" and also rarely "Attend School" or "Attend College".

D. Travel Distance

Another potential feature for finding place labels is the distance between places. We can take advantage of how far people typically travel between different types of places. As an example, Figure 3 shows the great circle distance people travel to different place types when starting from home, based on the PSRC data. From this plot, we see that people are willing to travel much farther from home to "Work" than to "Attend School". "Major Shopping" generally requires a longer trip from home than "Everyday Shopping".

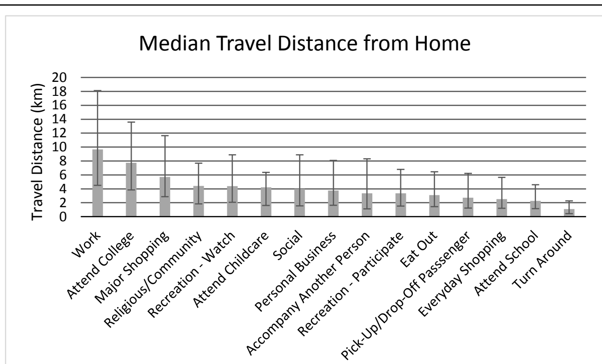

Figure 3: When the starting point is home, this is how far people travel to different types of places. The error bars show the 25th and 75th percentiles.

This section shows that features beyond the individual visit have the potential to increase classification accuracy. We explore the use of these new types of features in Placer++, which we describe in detail next.

IV. PLACER++

Our new Placer++ algorithm has two stages of inference using the feature categories described above. The Placer algorithm, augmented by cross label features, makes up the first stage. There are two options for the second stage, both of which use features derived from the sequences of visits. Figure 4 summarizes the flow of features through our different combinations of algorithms. We use the same 12 place types as Placer, which are listed in Figure 5.

A. Stage 1: Cross Label Features

We showed in Section III.A that cross labels (labels left at a location by other people) are potentially useful for labelling places. In this first stage of Placer++, we use the same machine learning scheme and features as Placer, but we augment the 69 Placer features with 76 cross label features. We label these features as CrossX, which indicates cross label feature X. The first four of these features are computed based on the majority place label left by others at the same location as the visit to be classified:

Cross1 - Label Winner - most frequent place label at this location
Cross2 - Winning Count - number of visits at this location with most frequent place label
Cross3 - Winning Fraction - fraction of visits at this location with most frequent place label
Cross4 - Average Duration Winner - average duration of visits at this location with most frequent place label

While the above set of cross label features looks at the most frequent label at the visit's location, the next set of cross label features looks at all 12 place labels at that location:

Cross5-6 (2 features) - Colocated "Home" by Others - fraction of total visits and average visit duration of "Home" visits by others to this latitude/longitude, similarly for following "Colated X by Others" features
Cross7-8 (2 features) - Colocated "Work" by Others
Cross9-10 (2 features) - Colocated "School" by Others
Cross11-12 (2 features) - Colocated "Restaurant or Bar" by Others
Cross13-14 (2 features) - Colocated "Personal Business" by Others
Cross15-16 (2 features) - Colocated "Store for Shopping" by Others
Cross17-18 (2 features) - Colocated "Place of Worship" by Others
Cross19-20 (2 features) - Colocated "Social" by Others
Cross21-22 (2 features) - Colocated "Recreation" by Others
Cross23-24 (2 features) - Colocated "Accompany Another Person" by Others
Cross25-26 (2 features) - Colocated "Transportation" by Others
Cross27-28 (2 features) - Colocated "Turn Around" by Others

In reality, exact colocation likely would not occur, due to the inevitable noise in location sensors, so we could not expect to find previous location labels whose coordinates exactly matched with a new label. This could be solved by setting a small distance tolerance of 10-20 meters to look for collocated labels. In the PSRC data, exact colocation *does* occur, because participants gave their locations in terms of street addresses which were then geocoded to latitude/longitude coordinates. Giving the same street address yields the same coordinates, leading to exact matches.

In Placer, one useful set of features gave counts of nearby business types and the distance to the nearest example of each type. We made a similar set of features for Placer++ using cross labels. These features give the count of nearby labels left by others and the distance to the nearest example of a label left by someone else. These nearby cross features from the 12 possible place labels are:

Cross29-32 (4 features) - Nearby "Home" by Others - count of "Home" labels by others within 50, 100, and 200 meters and distance to nearest "Home" by another, similarly for following "Nearby X by Others"
Cross33-36 (4 features) - Nearby "Work" by Others
Cross37-40 (4 features) - Nearby "School" by Others
Cross41-44 (4 features) - Nearby "Restaurant or Bar" by Others
Cross45-48 (4 features) - Nearby "Personal Business" by Others
Cross49-52 (4 features) - Nearby "Store for Shopping" by Others
Cross53-56 (4 features) - Nearby "Place of Worship" by Others
Cross57-60 (4 features) - Nearby "Social" by Others
Cross61-64 (4 features) - Nearby "Recreation" by Others
Cross65-68 (4 features) - Nearby "Accompany Another Person" by Others
Cross69-72 (4 features) - Nearby "Transportation" by Others
Cross73-76 (4 features) - Nearby "Turn Around" by Others

There are a total of 145 scalar features for Stage 1. These are the 69 original Placer features and the 76 additional cross label features (Cross1 – Cross76). For each visit, we compute the 145-element feature vector and submit it to our decision tree classifier. The classifier produces a discrete probability distribution over the 12 place types, giving a probability for each of the 12 place types.

We will explain later how we trained and tested our classifier. Next, however, we explain how we exploited sequence features in Stage 2.

B. *Stage 2a: Reranking from Whole Sequence Features*

We showed previously in this paper that there are certain consistencies in peoples' sequences of visits. For instance, there is often a 24-hour period in visits to home. Neither Placer nor Stage 1 of our classification process can take advantage of features related to a sequence of visits, because they classify each visit independently of the others based only on features of that visit. It is in Stage 2 that we start to use sequence features to further increase classification accuracy. The key difference between Stage 1 and Stage 2 is that Stage 2 can exploit features that depend on other visits in the sequence. This allows us to use features beyond just a visit, such as periodicity, cooccurrences, and driving distances. We do this by adopting a technique from natural language processing (NLP) that parses sentences.

1) *Candidate Sequences*

We start by describing a technique to generate likely candidate visit sequences from the results of Stage 1. Stage 2a will examine these candidates and pick the best one. The forest of boosted decision trees in Stage 1 gives a discrete distribution of classification probabilities for each visit. We say a sequence has V visits, and there are L possible visit labels. (For us, $L = 12$ possible visit labels given in Figure 5.) More precisely, for a

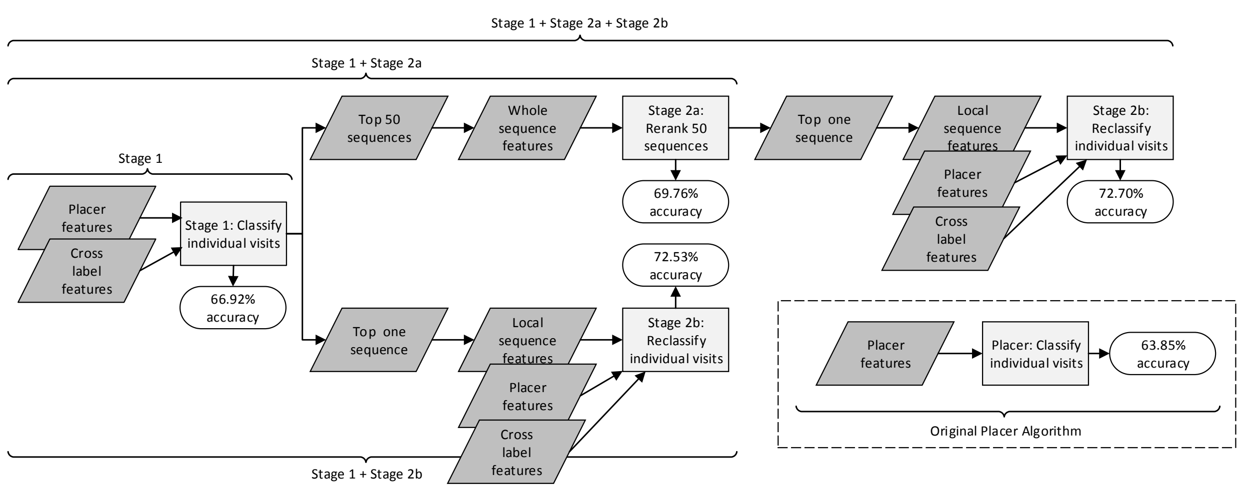

Figure 4: This is how features flow through our various combinations of classification algorithms. The parallelograms represent feature data, and the rectangles represent processing. For comparison, we show the original Placer algorithm in the box in the lower right.

given user, Stage 1 gives class probabilities p_{vl} for each visit $1 \leq v \leq V$ and each class label $1 \leq l \leq L$. If we stop with Stage 1, then the visit label we choose for visit k of a sequence is $l^* = \underset{l \in [1 \ldots L]}{\operatorname{argmax}}(p_{vl})$. The log-likelihood of this Stage 1 sequence is $\sum_{v=1}^{V} log(p_{vl^*})$. For Stage 2a, we use beam search to generate the top 50 candidate sequences for each user in descending order of log-likelihood according to the class probabilities from Stage 1. A reranking process, described next, is used to infer the best of these candidate sequences.

2) Sequence Reranking

A sequence of visits can be thought of as a sentence, where each word in the sentence corresponds to a visit. In NLP, one common task is to classify each word in a sentence into a part of speech, which is analogous to our classifying each visit into one of our 12 place types. This analogy led us to adopt a technique pioneered in NLP for parsing sentences into parts of speech. Collins and Koo [26] noticed that traditional NLP parse trees made it difficult to incorporate certain features, such as the feature that every sentence should contain a verb. Their solution takes a set of candidate parses and attempts to rank them such that the most accurate parse ranks first.

We will adopt the notation of Collins and Koo to explain how we used their technique for reranking sequences of visits. To be precise, one of their sentences corresponds to a sequence of visits from one person, which we will simply call a sequence. There are n sequences in a training set of sequences, each notated as s_i, $1 \leq i \leq n$. (We explain later how we split our data into training and testing.)

For training, the process starts by subjecting each training sequence s_i to our Stage 1 classification process. We compute the best $n_i = 50$ candidate sequences for each training sequence based on log-likelihood as described above. These candidate sequences are called x_{ij}, $1 \leq i \leq n$ and $1 \leq j \leq n_i$. The log likelihood of each candidate sequence is $L(x_{ij})$.

We define a set of $m+1$ sequence features $h_k(x)$, $0 \leq k \leq m$. In addition to the log likelihood feature $L(x)$, the other m features are very flexible, being based on all or part of the sequence. We describe the specific features we used in the next subsection. A linear combination of the features gives a ranking function for a sequence x:

$$F(x, \bar{\alpha}) = \alpha_0 L(x) + \sum_{k=1}^{m} \alpha_k h_k(x) \qquad (1)$$

The parameter vector $\bar{\alpha}$ is learned from with a fast algorithm [26] for linear RankSVM [27] that attempts to give the highest ranking to sequences with the highest fraction of correct visits.

For classifying visits in a new sequence, each visit is first passed through Stage 1, which gives a vector of classification probabilities to each visit. The classification probabilities are in turn used to compute the 50 most likely candidate sequences. The $m+1$ features for each of these sequences is given to the ranking function above, and we take the candidate sequence with the maximum ranking function value. It is important to note that the correct sequence might not appear in our top 50 candidate sequences. However, the goal of the reranking stage is to find a better sequence compared to the Stage 1 results by making use of sequence features that we gather from the whole sequence.

3) Whole Sequence Reranking Features

These are the 194 features that we used for reranking. None of them could be computed based on individual visits, which makes them quite different from the features used for Stage 1. We call them "WholeSequence" features because they each apply to the sequence as a whole. The features are:

- WholeSequence1 - Log Likelihood - Sum of logs of classification probabilities from Stage 1
- WholeSequence2 - 24 Hours +/- - Fraction of time label in sequence is same as label +/- 24 hours, based on sampling each visit in 5-minute increments
- WholeSequence3-14 (12 features) - Label Fractions - Fraction of total visits in sequence accounted for by each label
- WholeSequence15-26 (12 features) - Location Counts - Count of different locations for each label type in sequence
- WholeSequence27-170 (144 features) - Markov Probabilities - Given label A, how often does label B follow, A and B range over all 12 possible label pairs
- WholeSequence171-182 (12 features) - Duration Fraction - Fraction of time spent with each label
- WholeSequence183-194 (12 features) - Label Occurrence - Whether or not each label type occurred in sequence, binary

The output of Stage 2a is a sequence of visit labels, with one label for each visit of a user. An alternative to Stage 2a is Stage 2b, which we describe next.

C. Stage 2b: Reclassifying Visits from Local Sequences

The forest of boosted decision trees we used in Stage 1 was a powerful classifier, and we wanted to exploit its power for sequences. The input to Stage 2b is the maximum likelihood sequence of visits from Stage 1, *i.e.* the top sequence of the 50 generated for input to Stage 2a. (Recall that the flow of features through the various algorithm combinations is shown in Figure 4.) Stage 2b uses per visit features, as in Stage 1, but the Stage 2b features are based on the sequence of visits from the output of Stage 1. We computed per-visit features by splitting the input sequence back into individual visits, and we used features derived from the input sequence to reclassify each visit. Thus, each visit again had its own set of features, but these were derived from the whole input sequence. Although the Stage 2b features come from the sequence, most of them could not have been used as Stage 2a features. This is because Stage 2a features apply to the sequence as a unit. By splitting the sequence back into individual visits for Stage 2b, we can compute a separate set of features for each visit, yielding a potential boost in classification accuracy. As an example, one of our Stage 2b features gives the driving time to the next visit. While this is derived from the sequence, it applies only to the visit in question, not the whole sequence.

The 31 features we used for classifying visits in this this stage follow below.

- LocalSequence1-4 (4 features) - Previous/Next Labels - Inferred labels at relative visits -2, -1, +1, and +2
- LocalSequence5-8 (4 features) - Previous/Next Durations - Durations at relatives visits -2, -1, +1, and +2

LocalSequence9 - 24 Hours +/- - Inferred visit 24 hours previous (or 24 hours ahead if previous unavailable)

LocalSequence10-21 (12 features) - Label Occurrence - Whether or not each label type occurred in inferred sequence, binary

LocalSequence22-23 (2 features) - Previous/Next Drive Time - Drive time from previous visit to this visit, and drive time to next visit, computed from commercial routing program

LocalSequence24-25 (2 features) - Previous/Next Drive Distance - Drive distance from previous visit to this visit, and drive distance to next visit, computed from commercial routing program

LocalSequence26-27 (2 features) - Previous/Next Great Circle Distance - Great circle distance from previous visit to this visit, and drive distance to next visit

LocalSequence28 - Arrival Range - Timespan of range of arrival times to all instances of this label in sequence

LocalSequence29 - Departure Range - Timespan of range of departure times from all instances of this label in sequence

LocalSequence30 - Duration Minimum - Minimum duration of all instances of visits to this label in sequence

LocalSequence31 - Duration Maximum - Maximum duration of all instances of visits to this label in sequence

V. EXPERIMENTS

We used the PSRC diary study data [24] explained previously and in [6] for our training and testing. We split the data into three parts: A, B, and C, with A containing data for 40% of the study's households, B with another 40% of the households, and C with the remaining 20%. With multiple people from some households contributing data, we thought it was important to *not* scramble households between the three sets. This could give an unrealistic advantage to the cross label features, with possibly one household member in the training set contributing a "home" label to a location for a peer household member in the testing set.

We were faced with the problem of using our data efficiently and fairly across the three stages of processing. If we had much more data, we could split it such that there was no overlap in data between the three stages, generating a model from each stage that is used on fresh data in the next stage. In order to make more efficient use of our data, we devised a scheme that used the same data in all three stages, but that carefully avoided unfair leaks from the test data into the classification models. In general, we used data from subsets A and B for training, leaving the data in C for testing. The remainder of this section describes how we used the subsets for creating models and testing in the three stages.

A. Stage 1: Cross Label Features

Our first stage of visit classification is similar to that used in the original Placer algorithm. Each visit spawns a feature vector. Most elements of a visit's feature vector come from either characteristics of the person (*i.e.* age and gender) or characteristics of the visit (*i.e.* timing and surrounding businesses). If this were the extent of the features, then generating the training and testing data would be straightforward. However, there are also cross label features. In an actual implementation, cross labels are place labels that have been left prior to the visit that is being classified, by either the user or other users. The machine learning model would have an unrealistic advantage if it could use a person's own place labels to label their other places. It may even be considered unfair to use labels given by a family member to help compute labels. This is why we split our data into subsets A (40%), B (40%), and C (20%), with no persons or households split between any two or three subsets.

We save the C data for testing and use A and B for training. To compute cross label features for A, we use labels from B, and vice-versa. By using labels from B as cross labels for A, we ensure that none of the features for any visit in A have leaked from other visits in A. The same holds true for features in B. We concatenate all these feature vectors from A and B and build an inference model in the form of a forest of decision trees as described previously.

To test on data in C, we must first compute a feature vector for each visit in C. The cross label features for C come exclusively from visits in A and B. This ensures that features for visits in C are not affected by any other visits in C. We test the accuracy of our labeling by inferring a label for each visit in C from the model built from A and B, as above.

In order to establish notation for the next stage, we will use $DT_1^{(AB)}$ as the name of the forest of decision trees computed in Stage 1. The subscript 1 stands for Stage 1, and the superscript AB indicates that the decision tree was trained on data from both subsets A and B.

B. Stage 2a: Reranking from Whole Sequence Features

In Stage 2a, we again test on data in subset C, which means we use A and B for training. The training data for this stage consists of a list of 50 candidate visit sequences for each user. These sequences come from the classification probabilities computed by decision trees from Stage 1. Specifically, for the data in A, we compute a Stage 1 forest of decision trees called $DT_1^{(A)}$ from the visits in A, where the cross label features come from the data in B. Interchanging A and B, we compute another Stage 1 forest of decision trees called $DT_1^{(B)}$. As a reminder, each of these decision tree forests takes a feature vector describing a visit and gives a probability for each possible label of the visit. Given a sequence of visits by a user, we use these probabilities and a beam search to generate likely candidate sequences of visits by that user. To generate training data for Stage 2a, we apply $DT_1^{(A)}$ to visits in B to make $n_i = 50$ candidate sequences for each user in B. We will call this set of sequences $SEQ^{(B)}$. Vice-versa, we apply $DT_1^{(B)}$ to visits in A to make candidate sequences for each user in A, resulting in sequences $SEQ^{(A)}$. All these candidate sequences are concatenated to make a training set $SEQ^{(A)} + SEQ^{(B)}$. Using the reranking procedure described in previously, we use this training set to compute a weight vector called $\overline{\alpha}^{(AB)}$. For testing, we generate $SEQ^{(C)}$, which are candidate sequences from C using $DT_1^{(AB)}$, and we rerank these with the ranking function in Equation (1) using $\overline{\alpha}^{(AB)}$ as the weights. After reranking, the top-ranked sequence is the inference output of Stage 2a. This intertwining of data and models is intended to make efficient use of our data while still avoiding unrealistic cross talk between training and testing.

C. *Stage 2b: Reclassifying Visits from Local Sequences*

Stage 2b is an alternative to Stage 2a. It reclassifies each visit based on the inferred visit sequence from Stage 1. To be specific, from Stage 1, we assemble the most likely sequence for each user, which are the maximum probability label classification results from Stage 1. Once again we use data in A and B for training, and we cope with the cross labels in the same way as training for Stage 1. For each visit, we use all the features from Placer, the cross labels, and LocalSequence features. We build a decision tree from the training data in A and B, and we reclassify all the visits in C based on their preliminary classifications from Stage 1. For the decision trees, we use the same parameters as used in Placer and Stage 1.

VI. EXPERIMENTAL RESULTS

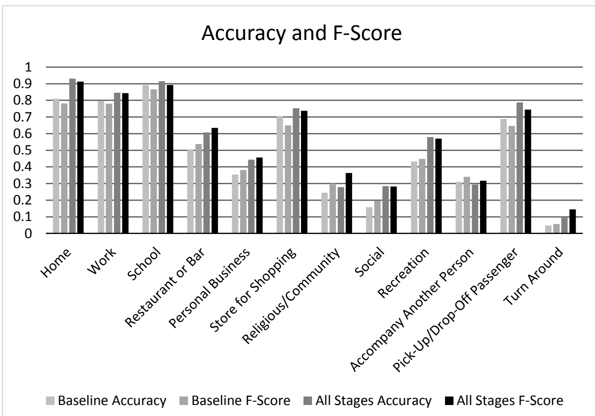

Figure 5: This shows the accuracies and F-scores of the baseline Placer algorithm and the 3-stage algorithm. We see improvement in all labels except "Accompany Another Person".

We tested our place labeling technique on the test data in subset C as described above. We first established a baseline by using the Placer algorithm [6], trained on A and B, and then tested on C. The Placer algorithm is the same as Stage 1 of the Placer++ algorithm, but omits the cross visit features. The overall accuracy for Placer was 63.85% with our particular set of test and training data. We used the same decision tree learning parameters as in the Placer paper. Because we used a single, predetermined set of learning parameters, we did not need a separate validation data set for experimenting with these parameters.

Applying Stage 1 of the new algorithm gave an overall classification accuracy of 66.92%, which is an improvement of 3.07 percentage points over the Placer baseline. This shows the improvement attributable to using labels from other users. These relative accuracy percentages are shown in Figure 5.

Stage 2a uses reranking to find likely sequences from the probabilistic output of Stage 1. Tested on subset C, its accuracy was 69.76%, an improvement of 5.91 percentage points over the Placer baseline.

Stage 2b performs a reclassification of each visit from Stage 1 based on other visits in the likely sequences. Testing on C, it gave an improvement of 8.68 percentage points over the Placer baseline, with an overall accuracy of 72.53%. We see that Stage 2b outperforms Stage 2a in terms of accuracy.

We can also apply all three stages to our problem. Stage 2a produces a sequence of visit labels that can serve as the input to Stage 2b. When we do this, we get, barely, the best overall accuracy of all our algorithm combinations at 72.70%, an improvement of 8.85 percentage points over the Placer baseline. Table 3 shows the confusion matrix after applying all three stages. Exceeding 80% accuracy are Home, Work, School, and Pick-Up/Drop-Off Passenger. Below 30% are Religious/Community, Social, Accompany Another Person, and Turn Around. Accuracies and F-scores of the baseline and 3-stage algorithm are shown in Figure 5. There is improvement for every label category except for "Accompany Another Person".

	Home	Work	School	Restaurant or Bar	Personal Business	Store for Shopping	Religious/Community	Social	Recreation	Accompany Another Person	Pick-Up/Drop-Off Passsenger	Turn Around	Recall
Home	0.94	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.94
Work	0.01	0.84	0.02	0.01	0.04	0.01	0.01	0.01	0.03	0.00	0.02	0.00	0.84
School	0.00	0.01	0.92	0.00	0.01	0.00	0.00	0.01	0.02	0.01	0.01	0.00	0.92
Restaurant or Bar	0.00	0.03	0.00	0.63	0.10	0.11	0.01	0.03	0.04	0.03	0.02	0.00	0.63
Personal Business	0.17	0.04	0.01	0.05	0.45	0.10	0.00	0.04	0.04	0.05	0.07	0.01	0.45
Store for Shopping	0.00	0.00	0.00	0.05	0.12	0.75	0.00	0.00	0.01	0.05	0.01	0.00	0.75
Religious/Community	0.00	0.14	0.01	0.03	0.09	0.02	0.29	0.15	0.21	0.01	0.05	0.00	0.29
Social	0.11	0.07	0.05	0.04	0.15	0.03	0.03	0.25	0.14	0.04	0.09	0.00	0.25
Recreation	0.02	0.05	0.03	0.04	0.09	0.02	0.01	0.07	0.56	0.05	0.05	0.01	0.56
Accompany Another Person	0.15	0.01	0.02	0.04	0.20	0.08	0.01	0.03	0.05	0.27	0.14	0.00	0.27
Pick-Up/Drop-Off Passsenger	0.01	0.03	0.00	0.01	0.06	0.01	0.00	0.02	0.02	0.03	0.80	0.00	0.80
Turn Around	0.01	0.00	0.00	0.10	0.10	0.04	0.00	0.01	0.05	0.04	0.56	0.11	0.11
Precision	0.89	0.85	0.87	0.64	0.47	0.72	0.48	0.32	0.58	0.38	0.70	0.23	

Table 3: This is the confusion matrix for the 12 different place types using all three stages of our algorithm.

VII. SUMMARY AND CONCLUSIONS

We have shown how to increase the accuracy of place labeling using two new types of features: cross labels and sequence features. We justified the use of these features by first analyzing statistics of daily visits.

Cross labels are place labels left by others that can help infer the proper label for a new visitor to that place. As Figure 2 shows, however, different people sometimes give different labels to the same place, so cross labels are not completely reliable. Cross labels were embodied in Stage 1 of our new algorithm, which achieved a classification accuracy of 66.92%, an improvement of 3.07 percentage points over the Placer baseline of 63.85%.

Sequence features exploit regularities in the order and relative timing of visits to different place types, embodied in Stage 2a and Stage 2b of our algorithm. Using all three stages, we increased classification accuracy to 72.70% over the baseline accuracy of the original Placer algorithm, an increase of 8.85 percentage points.

We expect one way to increase accuracy further would be to use more training data. For cross visits, we mentioned above that only 48.0% of locations in the PSRC data had labels from more than one person, and only 21.5% of places had identical labels from at least two people. With denser labeling, cross labeling should work better. Sequence features would likely help more if the training and test sequences were longer. The PSRC diary surveys lasted only two days, which may not always be enough to detect regularities in the sequences of places people visit. This is why we retained Stage 2a of our algorithm: despite its small boost in accuracy, it would likely help more for longer sequences.

One interesting area for future research is to extend classification algorithms to label places rather than visits. Our algorithm is aimed at classifying each visit to a place into a certain type, but it is not designed to classify each place. As we showed in this paper, different people give different labels to the same place, so a global label for each place is not appropriate. In fact, when we look at all 47,060 unique person/place pairs in the PSRC dataset, we find that 7.39% of the time the same person gave a different label to the same place. For example, we found instances of the same person labeling a place both "Eat Out" and "Everyday Shopping". This leads to other questions regarding the boundary between a place label and an activity. For understanding the context of the person, both are likely important, and one implies the other, but place and activity are still two different, albeit related concepts whose relationship and inference are ripe for more research.

REFERENCES

[1] S. Consolvo, I. E. Smith, T. Matthews, A. LaMarca, J. Tabert, and P. Powledge, "Location Disclosure to Social Relations: Why, When, & What People Want to Share," in *SIGCHI Conference on Human Factors in Computing Systems (CHI 2005)*, Portland, Oregon USA, 2005, pp. 81-90.

[2] D. Ashbrook and T. Starner, "Learning Significant Locations and Predicting User Movement with GPS," in *Sixth International Symposium on Wearable Computers (ISWC 2002)*, Seattle, Washington USA, 2002, pp. 101-108.

[3] X. Cao, G. Cong, and C. S. Jensen, "Mining Significant Semantic Locations from GPS Data," *Proceedings of the VLDB Endowment,* vol. 3, pp. 1009-1020, 2010.

[4] R. Hariharan and K. Toyama, "Project Lachesis: Parsing and Modeling Location Histories," presented at the Geographic Information Science (GIScience 2004), Adelphi, Maryland USA, 2004.

[5] J. H. Kang, W. Welbourne, B. Stewart, and G. Borriello, "Extracting Places from Traces of Locations," in *2nd ACM International Workshop on Wireless Mobile Applications and Services on WLAN Hotspots (WMASH 2004)*, Philadelphia, Pennsylvania USA 2004, pp. 110-118.

[6] J. Krumm and D. Rouhana, "Placer: Semantic Place Labels from Diary Data," presented at the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp 2013), Zurich, Switzerland, 2013.

[7] N. Marmasse and C. Schmandt, "Location-Aware Information Delivery withComMotion," in *Handheld and Ubiquitous Computing*, 2000, pp. 157-171.

[8] E. Miluzzo, N. D. Lane, S. B. Eisenman, and A. T. Campbell, "CenceMe–Injecting Sensing Presence into Social Networking Applications," in *Smart Sensing and Context*, ed: Springer, 2007, pp. 1-28.

[9] A. T. Palma, V. Bogorny, B. Kuijpers, and L. O. Alvares, "A Clustering-Based Approach for Discovering Interesting Places in Trajectories," in *2008 ACM Symposium on Applied Computing (SAC 2008)*, Fortaleza, Ceara, Brazil, 2008, pp. 863-868.

[10] I. Smith, S. Consolvo, A. Lamarca, J. Hightower, J. Scott, T. Sohn, *et al.*, "Social Disclosure of Place: From Location Technology to Communication Practices," presented at the Third International Conference on Pervasive Computing (Pervasive 2005), Munich, Germany, 2005.

[11] L. Barkhuus, B. Brown, M. Bell, S. Sherwood, M. Hall, and M. Chalmers, "From Awareness to Repartee: Sharing Location within Social Groups," in *SIGCHI Conference on Human Factors in Computing Systems (CHI 2008)*, Florence, Italy, 2008, pp. 497-506.

[12] G. Hsieh, K. P. Tang, W. Y. Low, and J. I. Hong, "Field Deployment of IMBuddy: A Study of Privacy Control and Feedback Mechanisms for Contextual IM," in *9th International Conference on Ubiquitous Computing (UbiComp 2007)*, Innsbruck, Austria, 2007, pp. 91-108.

[13] Y. Zhu, E. Zhong, Z. Lu, and Q. Yang, "Feature Engineering for Place Category Classification," in *Workshop on the Nokia Mobile Data Challenge*, Newcastle, UK, 2012.

[14] J. K. Laurila, D. Gatica-Perez, I. Aad, J. Blom, O. Bornet, T.-M.-T. Do, *et al.*, "The Mobile Data Challenge: Big Data for Mobile Computing Research," in *Workshop on the Nokia Mobile Data Challenge*, Newcastle, UK, 2012, pp. 1-8.

[15] C.-M. Huang, J. J.-C. Ying, and V. S. Tseng, "Mining Users' Behaviors and Environments for Semantic Place Prediction," in *Workshop on the Nokia Mobile Data Challenge*, Newcastle, UK, 2012.

[16] R. Montoliu, A. Martinez-Uso, and J. Martinez-Sotoca, "Semantic Place Prediction by Combining Smart Binary Classifiers," in *Workshop on the Nokia Mobile Data Challenge*, Newcastle, UK, 2012.

[17] Y. Zhu, Y. Sun, and Y. Want, "Nokia Mobile Data Challenge: Predicting Semantic Place and Next Place via Mobile Data," in *Workshop on the Nokia Mobile Data Challenge*, Newcastle, UK, 2012.

[18] L. Liao, D. Fox, and H. Kautz, "Extracting Places and Activities from GPS Traces using Hierarchical Conditional Random Fields," *The International Journal of Robotics Research,* vol. 26, pp. 119-134, 2007.

[19] Z. Chen, Y. Chen, S. Wang, and Z. Zhao, "A Supervised Learning based Semantic Location Extraction Method using Mobile Phone Data," in *2012 IEEE International Conference on Computer Science and Automation Engineering (CSAE)*, Zhangjiajie, China, 2012, pp. 548-551.

[20] M. Ye, D. Shou, W.-C. Lee, P. Yin, and K. Janowicz, "On the Semantic Annotation of Places in Location-Based Social Networks," in *17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2011)*, San Diego, California USA, 2011, pp. 520-528.

[21] D. H. Kim, K. Han, and D. Estrin, "Employing User Feedback for Semantic Location Services," in *13th International Conference on Ubiquitous Computing (UbiComp 2011)*, Beijing, China, 2011, pp. 217-226.

[22] M. Monibi and D. J. Patterson, "Getting Places: Collaborative Predictions from Status," presented at the Ambient Intelligence, Salzburg, Austria, 2009.

[23] M. Ames and M. Naaman, "Why We Tag: Motivations for Annotation in Mobile and Online Media," in *SIGCHI Conference on Human Factors in Computing Systems (CHI 2007)*, 2007, pp. 971-980.

[24] *Puget Sound Regional Council Household Activity Survey*. Available: http://www.psrc.org/data/surveys/2006-household/

[25] J. H. Friedman, "Greedy Function Approximation: A Gradient Boosting Machine," *Annals of Statistics,* vol. 29, pp. 1189-1232, 2001.

[26] M. Collins and T. Koo, "Discriminative Reranking for Natural Language Parsing," *Computational Linguistics,* vol. 31, pp. 25-70, 2005.

[27] C.-P. Lee and C.-J. Lin, "Large-Scale Linear RankSVM," *Neural computation,* vol. 26, pp. 781-817, 2014.