Inferring Person-to-person Proximity Using WiFi Signals

PIOTR SAPIEZYNSKI, Technical University of Denmark
ARKADIUSZ STOPCZYNSKI, Technical University of Denmark, MIT Media Lab
DAVID KOFOED WIND, Technical University of Denmark
JURE LESKOVEC, Stanford University
SUNE LEHMANN, Technical University of Denmark, Niels Bohr Institute

Today's societies are enveloped in an ever-growing telecommunication infrastructure. This infrastructure offers important opportunities for sensing and recording a multitude of human behaviors. Human mobility patterns are a prominent example of such a behavior which has been studied based on cell phone towers, Bluetooth beacons, and WiFi networks as proxies for location. While mobility is an important aspect of human behavior, it is also crucial to study physical interactions among individuals. Sensing proximity that enables social interactions on a large scale is a technical challenge and many commonly used approaches—including RFID badges or Bluetooth scanning—offer only limited scalability. Here we show that it is possible, in a scalable and robust way, to accurately infer person-to-person physical proximity from the lists of WiFi access points measured by smartphones carried by the two individuals. Based on a longitudinal dataset of approximately 800 participants with ground-truth Bluetooth proximity collected over a year, we show that our model performs better than the current state-of-the-art. Our results demonstrate the value of WiFi signals as a tool for social sensing and show how collections of WiFi data pose a potential threat to privacy.

CCS Concepts: • **Information systems** → **Location based services**; *Mobile information processing systems*; • **Applied computing** → *Sociology*;

Additional Key Words and Phrases: social sensing; WiFi; proximity; social networks

ACM Reference format:
Piotr Sapiezynski, Arkadiusz Stopczynski, David Kofoed Wind, Jure Leskovec, and Sune Lehmann. 2017. Inferring Person-to-person Proximity Using WiFi Signals. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 2, Article 24 (June 2017), 20 pages.
DOI: http://doi.org/10.1145/3090089

The authors would like to thank Andrea Cuttone for useful discussions as well as Urvashi Khandelwal and Jana Huisman for the important feedback. In this work we used the implementations of machine learning models from the scikit-learn [42] Python package. This work was supported by the Villum Foundation, the Danish Council for Independent Research, and University of Copenhagen (via the UCPH-2016 grant Social Fabric).

Author's address: P. Sapiezynski; email: pisa@dtu.dk; A. Stopczynski; email: arks@dtu.dk; DK Wind; email: dawi@dtu.dk; J. Leskovec; email: jure@cs.stanford.edu; S. Lehmann; email: sljo@dtu.dk;

2474-9567/2017/6-ART24 $15.00
DOI: http://doi.org/10.1145/3090089

1 INTRODUCTION

We are surrounded by an ever-increasing number of telecommunication infrastructures, such as mobile phone networks, WiFi access points, or Bluetooth beacons. In addition to their intended function of providing connectivity, these infrastructures offer an unprecedented opportunity for sensing, modeling, and subsequent analysis of a wide range of human behaviors [31]. Here we show how our proximity with other people can be inferred in a reliable and scalable way, using signals from WiFi access points.

Being able to infer person-to-person proximity events with high spatio-temporal resolution enables modeling of phenomena such as spreading of diseases and information [25], formation of social ties [13], as well as group dynamics [49]. Commercial applications vary from distributed ad hoc networking [32] to romantic matchmaking [11].

Despite the importance of understanding networks of physical proximity, there is a scarcity of scalable and efficient ways to obtain this type of data with high temporal resolution in the case of large populations. This is due to the fact that technology has only recently developed to the point where collection of such data has become technologically feasible. The data sources used for investigating mobility of individuals, such as call detail records (CDRs) from mobile operators [17],[1] are too coarse in terms of temporal and spatial resolution to allow inference of person-to-person proximity. On the other hand, the current state-of-the-art methods for measurement of physical proximity require using specialized hardware (*e.g.*, sociometric badges) [39, 44] or smartphones sensing each other through Bluetooth [2, 12, 56]. Specialized hardware adds cost and complexity to experimental deployments, effectively limiting their scale. Bluetooth scanning realized on participants' mobile phones increases power consumption [15]—limiting temporal resolution that can be achieved—and requires the devices to be in Bluetooth *discoverable mode*. This requirement raises privacy [64] and security concerns [47]. When a phone is in discoverable mode the location of its owner can be tracked by third parties, a fact commonly used by researchers [30, 41], and advertisers [10]. Moreover, whenever a phone is discoverable, a malicious actor can attempt to pair to it in order to steal contact lists or content of messages. For these reasons phone manufacturers make it difficult (or impossible) for a handset to remain discoverable indefinitely. iOS and Android 6.0+ devices disable discoverability whenever the user exits the Bluetooth settings screen. Older Android devices let the user set the discoverability timeout to, at maximum, five minutes. In our study we relied on the fact that in Android it is still possible to *request* (not *set*) an unlimited discoverability timeout programmatically. The Android developer documentation explicitly discourages using the setting because of security concerns [18]. Moreover, the user is required to confirm this operation every time it is requested, for example every time they access the device's Bluetooth settings.

Apart from the usability, privacy, and security issues of using Bluetooth for sensing, another shortcoming is that Bluetooth data lacks location context. When co-presence of individuals is inferred through devices sensing each other, an additional step is usually required to estimate the location of the meeting, for example by comparing Bluetooth scans with GPS measurements [49], by using fixed infrastructure of RFID transmitters [52], or Bluetooth beacons [30]. Finally, Bluetooth scanning is a burden on battery life, a full Bluetooth scan lasts as long as 30 seconds and requires more energy than a WiFi scan [33]. In the light of these problems, it is clear that alternative methods for tracking person-to-person proximity are needed. We propose that using WiFi signals for sensing proximity between devices can mitigate the limitations of the Bluetooth based approach. There have been attempts at exploiting WiFi signals for this purpose (*e.g.*, [27, 29, 36, 37] further described in the related work section) but their general applicability is unclear. The previous methods were only trained and tested in controlled environments, and they lack verification on longer timescales.

[1] Call detail records contain metadata about each call - caller and callee phone numbers, timestamps, and call tower IDs translatable to physical locations.

Present work. Here we study the problem of inferring physical proximity between pairs of individuals from a list of WiFi signals measured by their phones. We use a longitudinal dataset containing WiFi and Bluetooth scan results from hundreds of participants, collected over a year as part of the Copenhagen Network Study [56]. Using Bluetooth as ground-truth for physical proximity, we train a model for comparing the results of WiFi scans from two devices to determine whether two individuals were in close physical proximity. We employ a number of interpretable metrics to compare the lists of visible WiFi access points, such as Jaccard similarity or correlation of received signal strengths. In addition to comparing the lists directly, we can derive context from just the number of routers seen in the lists: more populated areas tend to have more routers available. Furthermore, we exploit the characteristics of proximity dynamics, for example that people are more likely to meet during work hours, or on a Friday afternoon than on a Sunday night. As a final step, we are able to combine these insights using machine learning models to achieve the area under receiving operator curve (AUC ROC) scores of up to 0.89 in the proximity inference task. Note that this method infers proximity by directly estimating similarity between WiFi environments and does not rely on positioning routers in physical space and thresholding the estimated distance between pairs of individuals. We show that our model works in a range of environments, does not depend on particular access points, and its performance does not deteriorate over time. Our experiments demonstrate that we are able to track person-to-person proximity (as defined as being within Bluetooth range) over time and in different social and spatio-temporal contexts. Overall, our approach performs better than previously suggested solutions.

2 EXPERIMENTAL DESIGN

The dataset used in this work was collected as part of the Copenhagen Networks Study [56]. It covers mobility and proximity records of approximately 800 students at Technical University of Denmark, over a two year period. Each student was equipped with a LGE Nexus 4 Android smartphone as a data collecting device. The data collection was enabled at all times when the phones were on—both on and outside of campus—regardless of the participants' activity. On each phone, an application based on the Funf Open Sensing framework [2] gathered readings from multiple sensors including:

- **Bluetooth scans** (every 5 minutes):[2] each scan contains a list of discoverable devices, their unique identifiers, user defined names, and received signal strength (RSSI). We can use RSSI to approximate the distance between the participants.
- **WiFi scans** (every 5 minutes): each scan contains a list of WiFi access points (both traditional routers and mobile hotspots), their unique identifiers (BSSIDs or MAC addresses), network names they transmit (SSIDs), and RSSI.

The collector app additionally collected data requested by other applications on the phone. Therefore, the temporal resolution of the data for some of the users can is higher than one sample every 5 minutes.

All data in the Copenhagen Networks Study was collected with the participants' informed consent, with an emphasis on ensuring awareness of the complexity and sensitivity of the collected data [54]. The study setup, including security, privacy, and informed consent has been approved by Danish Data Protection Agency. Further details of the study can be found in Ref. [56].

3 METHODS

In brief, our task is to compare the lists of WiFi routers seen by users A and B approximately at the same time (with at most $\Delta t = 300$ seconds difference) and determine whether the two users were in close physical proximity. We use Bluetooth data as ground truth for physical proximity to train and verify our models.

[2] Smartphones in the study were specifically configured to be in Bluetooth discoverable mode.

Table 1. Summary statistics of the dataset used to infer proximity events.

	training	test
total observations	0.5M	115.5M
% positive	31%	31%
unique users	812	820
median number of access points per observation	7.0	7.0
mean number of access points per observation	11.3	11.3

3.1 Data preparation

WiFi. In the dataset there are multiple physical devices (WiFi routers) that share the same MAC address. This phenomenon might confound our task: two people in different parts of the city might appear proximate if they scan two different routers with the same "unique" identifier. We remove these routers using a simple heuristic. Specifically, we rely on the network name they broadcast. Because the routers at DTU campus broadcast up to four network names (SSID) per MAC address, we remove the scans of routers which broadcast five or more network names throughout the observation. This simple approach might lead to removal of valid devices associated with many WiFi names, but at the same time limits the number of possible false positive detections of proximity. We found 3950 offending MAC addresses, which corresponds to only 0.04% of all unique MAC addresses in the data. However, scans of these routers constitute 1.4% of all scan results.

Next, we identify one home router for each participant per month. We employ the following heuristic for each participant:

(1) Bin the time information of WiFi scan history.
(2) Sort the list of routers by the number of timebins in which they appear, in descending order.
(3) The router that appears in the largest number of timebins is assumed to be the home router.

This procedure is robust to varying bin-size, here we use 10 minutes. The full details of the procedure are described in Ref. [46].

Bluetooth. Due to the imperfect firmware and software running on the phones, Bluetooth data is not always available—not all users are scanning and discoverable at all times. This can introduce a situation in which two persons are proximate, but Bluetooth does not capture that event. We divide the dataset into one hour subsets and select only the WiFi and Bluetooth data from people who were seen and who saw at least one other person through Bluetooth. This strict approach makes the task more difficult, as it removes long periods where individuals are alone, for example night-time samples of students who do not live with other participants.

Negative samples. To train our model we also need to provide negative examples. For dyads in this category we choose potential proximity events between two people who did not see each other on Bluetooth, but whose lists of scan results share at least one overlapping router. Compared to selecting negative samples by randomly sampling dyads this definition brings the task closer to a real-life scenario of discovering very close physical proximity (up to approximately 10 meters). As a result, the dataset has 31% positive and 69% negative samples.

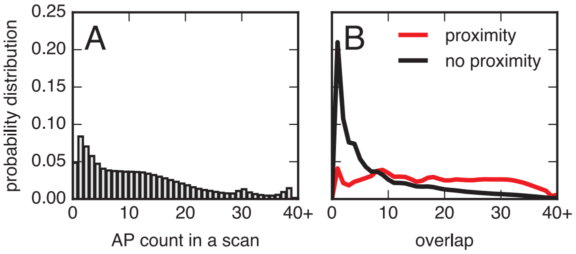

Fig. 1. a. More than 95% of scans report at least one access point, and 12 APs on average. b. People in Bluetooth proximity scan more overlapping routers than those who are not proximate.

3.2 Dataset statistics

Table 1 shows the dataset's overall statistics. Through a year of data we found 116M potential proximity events. We randomly select 0.5M of them to train and validate the models, and the rest to test them.

In our dataset people are near to access points more than 95% of time, and the average count of routers in a single scan is 12, see Figure 1A. We also observe that in 99% of cases of Bluetooth sightings the corresponding WiFi scans overlap by at least one access point. This indicates that there is a potential in using WiFi scan results to infer the co-presence with high recall. Conversely, in more than 31% of cases where there is at least one overlapping access point, the two devices are also close according to Bluetooth. This indicates that WiFi signals can be applied to the task potentially resulting in a high precision solution. In general, pairs of people who are in Bluetooth proximity scan more routers in common than those who are not, see Figure 1B. The majority (53%) of meetings happen during working hours (from 8am to 7pm) on campus.

3.3 Methods of comparison

We use a number of metrics to compare pairs of WiFi scan results and use these metrics as features in a supervised machine learning approach. We divide the features into the following categories: availability of access points, received signal strength, presence + RSSI, timing, popularity, and location. Table 2 lists the features we apply, and Figure 2 shows how the probability of a proximity event changes as a function of each feature's value. In this section we describe each feature in detail. Citations refer to the first articles using each feature for the purpose of person to person contact detection.

Availability of access points (AP presence). First, we compare the list of routers seen by the two phones, without incorporating information regarding their received signal strengths. We introduce the following measures: **overlap**: the raw count of overlapping routers [29]; **union**: size of the union of the two lists; **jaccard**: ratio between the size of the intersection and the size of the union of the two lists [27]. **non-overlap**: the raw count of non-overlapping routers (size of union minus size of overlap) [29]; Figure 2A-C presents the interplay between the values of the three parameters and the probability of proximity. Intuitively, the greater the number of common routers two phones measure in a scan, the higher the probability of them being in close proximity. Perhaps surprisingly, this probability also depends on the size of the union: the larger the union of the two lists the lower the probability of proximity. This can be explained by the fact that the number of available access points is positively correlated with the population density [46]. Hence, popular places are likely to attract people who do not necessarily interact with one another. Conversely, two people in a relatively unpopular location are more likely to be there together. The visible dip in the union plot, corresponding to lower probability of meeting with around 30 routers present, might correspond to a particular location where many people scan the same router

Table 2. Features used to infer person-to-person proximity.

category	features
AP presence	overlap, non-overlap, union, jaccard
RSSI	spearman, pearson, manhattan, euclidean
AP presence + RSSI	top AP, top AP±6*dB*
timing	hour of week
popularity	min popularity, max popularity, Adamic-Adar
location	at home, at DTU

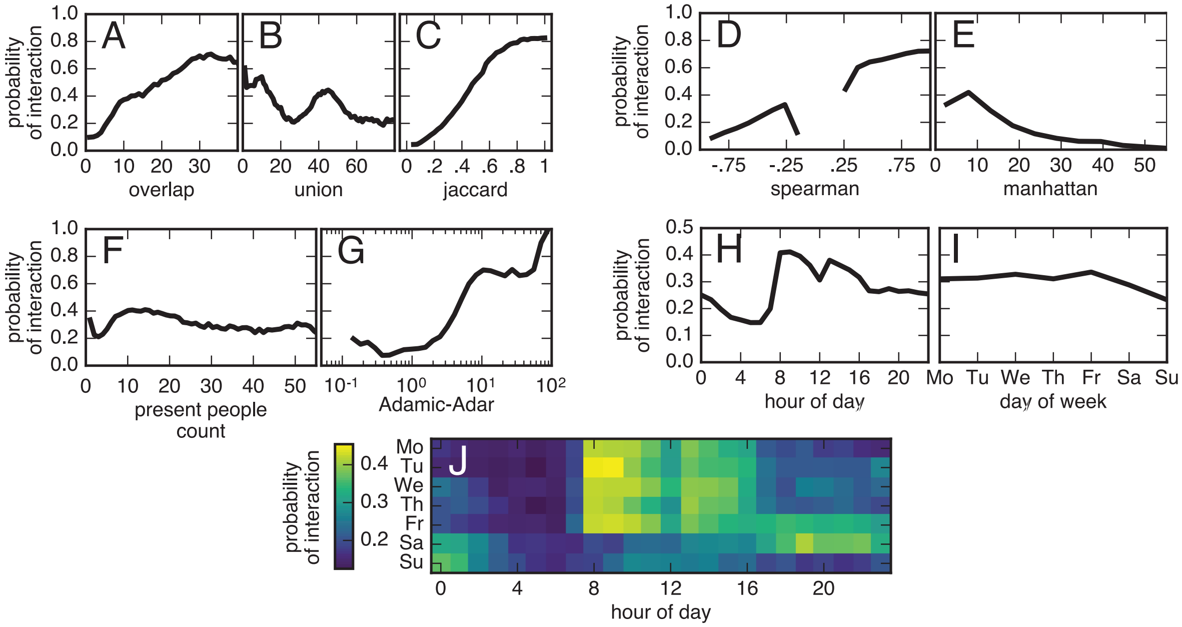

Fig. 2. The larger the number of common routers two phones measure, the higher the probability of close proximity. At the same time, the more routers they see in total, the lower the probability of a proximity event — densely populated areas have more routers and more people who are not necessarily proximate. Jaccard similarity allows us to recognize proximity regardless of the number of visible access points.

even though not in Bluetooth proximity (for example a dining hall). Nevertheless, we expect that, in general, the probability of proximity is negatively correlated with the size of union. Using Jaccard similarity between the two lists allows us to recognize proximity regardless of the number of visible access points.

Received Signal Strength Indicator (RSSI). Next, we focus on comparing the received signal strength of the overlapping routers. While received signal strength (RSSI) is not generally a reliable proxy for distance [45], two co-located people can be expected to have similar RSSI readings for the overlapping routers. We investigate the **spearman** and **pearson** correlation coefficients of received signal strengths of the overlapping routers. For brevity we only present the results for the spearman metric in Figure 2D — the values of the two metrics are highly correlated (Spearman's $\rho = 0.89$, $p_{val} < 0.001$). Note that because there are instances where the correlation is undefined (*not a number*) or not statistically significant (with $p_{val} > 0.05$); we replace such values of the coefficients with the mean values of valid correlations (see section 3.4 for details of the imputation). Therefore, there are no examples of small correlations (which, given only a few values to compare, are not statistically significant) and there is a dip in probability of proximity (corresponding to the mean value of correlation coefficients).

Furthermore, we also calculate the difference between RSSI of overlapping routers by measuring the ℓ_1 and ℓ_2 distances and dividing the results by the number of overlapping routers. For simplicity we call these features **manhattan** and **euclidean** and define them in Equations 1 [37] and 2 [27] respectively.

$$m = \frac{\sum_i \left| RSSI_{A,i} - RSSI_{B,i} \right|}{N} \tag{1}$$

$$e = \frac{\sqrt{\sum_i (RSSI_{A,i} - RSSI_{B,i})^2}}{N} \tag{2}$$

where $RSSI_{A,i}$ is the received signal strength of access point i as measured by user A, and N is the total number of overlapping routers. Figure 2E shows that with growing distance, the probability of proximity falls.

AP presence + RSSI. It has been shown previously that if we want to determine if a user remains in the same position during two subsequent WiFi scans, testing if the two scans share a common strongest router is a good heuristic. [16]. Here, we verify whether this approach can be used for inferring co-location: if two users measure the same router as the strongest one, we assume they are in close proximity. We investigate the strict case, **top AP**. Additionally, we allow for some variability in the measured strength: feature **top AP**$\pm 6dB$ assumes a positive value if there is at least one overlapping access point in the lists of routers of A and B within $6dB$ from the top router.

Popularity. Additionally, we inspect how many different participants of the study scanned the overlapping routers within five minutes of the meeting—intuitively if only a few persons were in a given location they were more likely to be there together, rather than by chance. We find the least and the most popular among the overlapping routers and report **min_popularity** and **max_popularity**. As we show in Figure 2F, this intuition is not entirely confirmed by the data. The correlation between the number of individuals present and the probability that any two of them are proximate is low (Spearman's $\rho = 0.15$, $p_{val} < 0.001$). Note that popularity and the size of union are correlated (Spearman's $\rho = 0.48$, $p_{val} < 0.001$) — more routers are located in popular places, so the more routers there are around, the more people see each of them. However, to achieve a good estimation of popularity, we need data from the entire population, while the number of routers around can be obtained just from data of just the two individuals. Additionally, we use a score inspired by a measure introduced by Adamic and Adar [1], defined as:

$$aa(u_1, u_2) = \sum_i \frac{1}{\log(popularity(AP_i))}. \tag{3}$$

Here, each overlapping router is weighted more the fewer people scanned it. In this case, the higher the value, the higher the probability of a meeting between two people.

Timing. In contrast to the other features we described, timing does not rely on comparing the list of scan results. Instead, we use the timestamp of each potential meeting to exploit the temporal characteristics of human proximity patterns. As a reminder, we only consider a potential proximity event if both parties have WiFi scans within 300 seconds from one another. For simplicity, we assume that the timestamp of the potential proximity is the lower of the two scan timestamps. We notice that the prior probability of two people being proximate depends on the time of day and the day of week, as shown in Figure 2H-J. While there is only a small variability between the days of the week (Figure 2I), the probability of being proximate during a day (Figure 2H) appears to be driven both by the class schedule—the probability is the highest during classes, and drops during lunchtime—and by after-school social activities. Only by combining the two factors (Figure 2J), we get the full picture: the probability of proximity from Monday to Tuesday is driven by the school schedule; Friday is a mixture of scheduled and socially-driven proximity, with the probability remaining high far into the night hours; Saturday is characterized by proximity events starting in the late afternoon and into the night; and on Sunday our participants are proximate mostly during daytime, with no visible lunch breaks. We add a feature to capture these patterns: **hour of week**: simply indexed from 0 to 167.

Location. The last category, location, contains two binary features. A meeting is considered **at home** if at least one of the routers in the union corresponds to the home router of one of the users (the heuristic for home location detection is explained in Section 3). A meeting is assumed to take place **at DTU** if at least one of the routers in the union broadcasts a WiFi network name of `dtu`, as all access points on the campus do.

3.4 Imputing missing values

In the case of the Pearson and Spearman features discussed above, there are two cases in which it is not possible to calculate the correlation: (1) if there are fewer than three routers available for comparison, (2) if at least one person reads all the signal strengths at the same level. In such cases we assume a NaN (not-a-number) value of ρ to be imputed later on. Additionally, we assume a NaN value of ρ if the correlation is not significant with the $p_{val} < 0.05$. This results in multiple missing values for the two features. The simplest approach is to ignore such observations, but that would imply not training the model in cases with few routers available. We therefore impute the values by assigning the mean value of the feature (averaged over all the non-NaN training examples) when we encounter NaN values. This average from training is preserved and used to impute missing values in the test set. We verified in our data that other approaches, such as using the median value of the feature or using k nearest neighbors to impute the missing value [59], do not improve the consecutive predictive performance.

4 RESULTS

In this section we evaluate the performance of each feature and each featureset in the task of proximity inference. Then, we examine the robustness of our best model to short training as well as the various types of environments in which the proximity events happen. Note that the performance of single features as well as models trained on feature sets is near-identical in the randomly selected training set and the remainder of the dataset. This indicates that the training sets are large enough and representative of the rest of the dataset, and that the models to not overfit to the training data.

4.1 Performance of single features

We first show how well one can infer proximity using single features. Here, we do not apply machine learning, but use the raw values of each feature. We report the area under Receiver Operating Characteristic curve (AUC ROC) as the first metric of performance in Table 3. Then, we select the threshold at which the F_1 score (the harmonic mean between precision and recall) is maximized in the training set. We also report the F_1 score at the threshold optimal for the training set along with the AUC ROC for the test data (111.5 million previously unseen samples).

The results are presented in Table 3. We find that the best performing single feature is Jaccard similarity between the two lists of routers. As expected, thresholding on time information is not meaningful (it is equivalent to assuming that all possible proximity events after a certain hour of a certain day of week are positive). This feature can still prove informative in conjunction with other features in the machine learning task. It is important to note that the performance in test does not drop compared to training, which means that the thresholds are not just specific to the training data.

4.2 Performance of feature sets

We now want to to explore the gains from combining features. In order to do this, we begin by training a Gradient Boosting Classifier for each category of features and present the results in Table 4. The parameters of the classifier are tuned each time through a grid search of the parameter space with 5-fold cross validation. The reported performance in the training dataset is the performance of the fine-tuned model on validation dataset, not on the training data itself. Furthermore, we compare models based on the features proposed by Krumm *et al.* [29] to models based on richer sets of features, see Table 4. In the original work, Krumm *et al.* found that the physical distance between two devices is best estimated using a linear model based on overlap and Spearman correlation coefficient. In Table 4 we refer to a model based on this features as NearMe - original. Here, we show that combining all the features they proposed—as opposed to only two they use in the final model—does improve the performance in the classification task. In Table 4 we refer to a model based on this features as NearMe - full. Our Simple model is based on features that do not require long term data collection and are not specific to our deployment. It performs better than any single feature or group of features, and it slightly outperforms the models based on the features introduced by Krumm. Enhancing the model with the information on popularity (the General model) further improves the performance. Finally, we find that using all features, including timing and location (which might be specific to this experiment as they depend on our campus as location and the time schedule typical for students), does not improve the performance of the classifier.

4.3 WiFi similarity and physical proximity

Next, we verify whether there is a correlation between how close people are in physical space (approximated by the received Bluetooth signal strength measured on their phones) and the probability that our models misclassify the sample as "non-proximity". As we show in Figure 3, the closer the proximity (approximated by high Bluetooth RSSI), the lower the probability of missing that event. This shows that the similarity measure between WiFi lists introduced by our models has a physical interpretation: a more similar WiFi environment indicates proximity in a more granular way than just the Bluetooth 10 meter range.

4.4 Training period and performance in test

Figure 4 shows how the number of samples used for training influences the performance of the full model in test. We compare the performance of a random forest classifier and a gradient boosted classifier and find that the latter has a slightly higher performance for training sets larger than 1000 samples. On the other hand, training of the random forest classifier can parallelized, thus making the process faster.

4.5 Testing in unseen locations

Models proposed in earlier literature (for example [29]) performed worse if it was tested in locations which were not part of the training set. Here, we verify to which extent our models suffer from this limitation. We annotate each possible proximity event with the ID of strongest visible access point as a proxy for location. We train

Table 3. Performance of single features and feature categories in the task of inferring proximity events.

		AUC ROC		F_1	
category	feature	train	test	train	test
AP presence	overlap	0.77	0.77	0.61	0.61
	jaccard	0.84	0.84	0.69	0.68
	union	0.53	0.53	0.48	0.48
	non-overlap	0.74	0.74	0.58	0.57
RSSI	spearman	0.70	0.70	0.57	0.58
	pearson	0.71	0.71	0.59	0.59
	manhattan	0.60	0.60	0.51	0.51
	euclidean	0.59	0.59	0.51	0.51
Presence + RSSI	top AP	0.60	0.60	0.48	0.48
	top AP±6dB	0.75	0.74	0.65	0.65
Popularity	min_popularity	0.54	0.54	0.48	0.48
	max_popularity	0.59	0.59	0.49	0.50
	adamic_adar	0.77	0.77	0.62	0.62
Timing	hour of week	0.51	0.51	0.48	0.48
Location	at DTU	0.61	0.61	0.51	0.51
	at home	0.64	0.64	0.55	0.55

The jaccard similarity between lists of routers seen by the two devices is the best performing single feature. F_1 are given for a threshold that maximizes F_1 in the training set.

Table 4. Performance of feature sets in the task of inferring proximity events.

	AUC ROC		F_1	
featureset	train	test	train	test
AP presence: overlap, non-overlap, jaccard, union	0.85	0.85	0.69	0.69
RSSI: spearman, pearson, manhattan, euclidean	0.78	0.79	0.62	0.62
Presence+RSSI: top AP, top AP±6dB	0.75	0.75	0.65	0.65
Popularity: min, max, adamic_adar	0.79	0.79	0.62	0.62
Location: at DTU, at home	0.65	0.65	0.55	0.55
NearMe (original): overlap, spearman	0.82	0.82	0.62	0.62
NearMe (full): overlap, non-overlap, spearman, euclidean	0.87	0.87	0.71	0.71
Simple: **AP presence**, **RSSI**, **Presence + RSSI**	0.88	0.88	0.72	0.72
General: **AP presence**, **RSSI**, **Presence + RSSI**, **Popularity**, at home	0.89	0.89	0.73	0.73
Full: all features	0.89	0.89	0.73	0.73

We train a Gradient Boosted Classifier on selected subsets of features: each feature category listed in Table 3, NearMe [29], Simple (no features that are specific to this experiment or require longer term data collection), General (without features that could be specific to this experiment), and Full (all listed features). Using features which could be specific to the experiment does not improve performance further.

the model on k potential proximity events and test it on events which did not happen in the locations used for training. As we show in Figure 4, models trained on a few data samples do perform worse in unseen environments. However, this disadvantage is no longer present with 10 000+ samples used for training. Importantly, we note

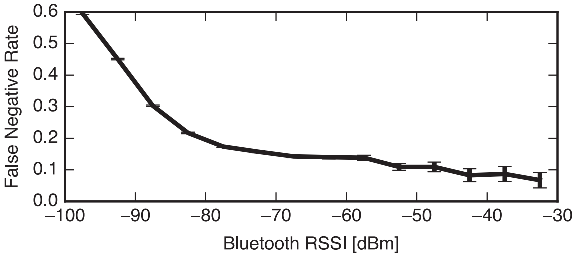

Fig. 3. The distance between to proximate devices can be approximated using Bluetooth received signal strength (RSSI). Very close proximity contacts are unlikely to be misclassified as non-proximity. The lower the RSSI (approximating larger distance between individuals), the higher the probability, that our models miss the proximity event.

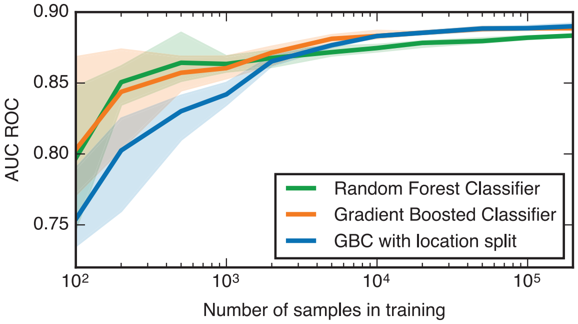

Fig. 4. The more samples we use for training the proximity detection models, the better they perform in test, but after a certain thresholds, the gains are negligible. The performance of the Gradient Boosted Classifier saturates at a higher level, but the time it takes to train the classifier is longer than it is the case with the Random Forest Classifier. When the models are tested in locations which were not part of the training set (blue line), they perform worse, but this limitation is no longer a problem for larger training sets. Each of the model is trained 20 times for each number of samples, the shaded areas correspond to 25-75 percentiles and the solid lines to medians of the results for each training set size.

that the actual location identifier is not a feature used by the models. Therefore, the models do not overfit for particular locations based on their ID.

4.6 Importance of features

In this section we discuss how important each feature is for the machine learning model. In the implementation we use [42] the feature importance is defined as the total decrease in node impurity weighted by the probability of reaching that node, averaged over all trees of the ensemble [34]. Figure 5 shows the accumulated results from 30 training rounds of the gradient boosted classifier on randomly selected subsets of the training data, each with 100 000 samples. We find that Jaccard similarity is the most important, followed by the overlap among the strongest routers, Pearson's correlation of signal strengths, and Adamic-Adar (which exploits the overlap and the popularity of routers).

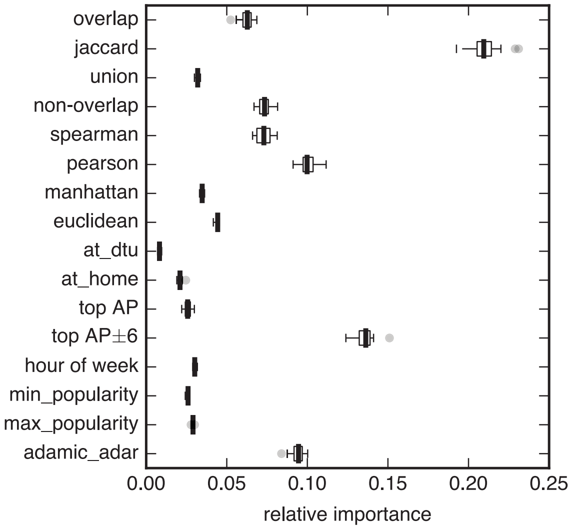

Fig. 5. Gradient Boosted Classifier reports the relative importance of each feature (the decrease in node impurity it provides). After 30 training rounds we see that Jaccard is the most important feature, followed by overlap among the strongest routers (top AP±6*dB*), Adamic-Adar, and Pearson correlation between the signal strengths.

4.7 Validity of the model in various scenarios

Figure 6 shows the performance of the gradient boosting classifier in different contexts and across time.

Number of routers. We divide the test data in three equally-sized subsets, depending on the size of the union of routers seen by two people: low - $[1 - 28)$ routers, mid - $[28 - 44)$ routers, and high - $[44 - 240)$ routers. While these numbers might appear surprisingly large, it is important to remember that (1) most meetings in the dataset happen at the campus where many routers are always visible, and (2) we consider size of the union, so not all the routers are expected to be visible at once from a single location. Figure 6A shows that the performance of the model is best in the low and mid sets ($AUC \geqslant 0.9$) and observably lower ($AUC \approx 0.84$) for environments with the highest number of routers. Thus, we show that the model performs well in typical environments.

Number of persons present. Previously, researchers observed that human bodies attenuate the Bluetooth signal. We, therefore, expect received signal strengths to be lower in densely populated locations, and some proximity events might not even be registered. WiFi routers operate at the same frequency as Bluetooth (2.4 Ghz) and the WiFi signal is subject to the same problem. Here, we verify the performance of our model in such situations. We divide the test data in three equally-sized subsets, depending on the maximum popularity feature: low (up to six people), mid (seven to 19), and high popularity (20 and more persons). Figure 6B shows that the performance of the model drops as a function of the number of people present. It is best in the low set ($AUC \approx 0.94$) and worst ($AUC \approx 0.83$) for environments with the highest number of people. We provide a further interpretation of this result in the analysis section.

Location. Because our the data was collected by students of a single university, with the majority of proximity events taking place on campus, there is a risk that the model would overfit towards such situation. This is, in fact,

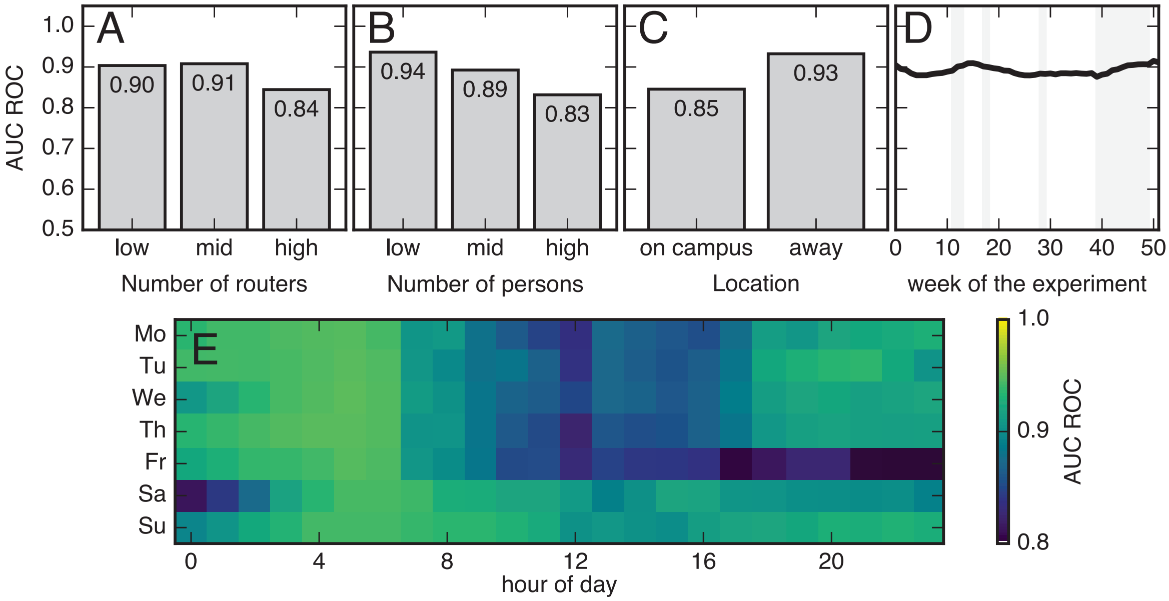

Fig. 6. Our model for detecting person-to-person proximity events performs with high AUC regardless of the number of available routers (A), the number of people present (B), and the location of potential proximity (C). The performance does not drop during holidays (marked with gray areas in D). Performance is worst during Friday evenings and nights (F), but even then, the AUC ROC is high.

not the case. Figure 6C shows that while the performance of the model is high on campus, it is even better for the meetings outside campus.

Timing. As shown in Figure 6D the performance of the model does not drop significantly during special periods, such as Christmas of summer vacation (gray areas in the plot correspond to periods with no university classes). Instead, it remains stable throughout the experiment.

The performance does vary with the hour of week, as shown in Figure 6E When we compare it to Figure 2K, we see that the model performs better in situations where the prior probability of meeting is lower (for example during week nights). This might be related to the problem of dense crowds attenuating Bluetooth signals and failing to register devices within normal operating range. Nevertheless, the model retains high performance of $AUC \geqslant 0.8$ throughout the week.

5 RELATED WORK

In this section we discuss related work that explores the application of mobile data to deepen our understanding of aspects relevant to this paper.

Location and mobility. CDR data has been used as a proxy for human mobility at large scale. It has been shown that our movements are regular [17], stable [35], and predictable [50]. Several works argue that many unpredictable travels observed in real data can be attributed to individuals seeking interaction with their social contacts [9, 19, 58]. It yet remains to be verified whether these findings hold fully if the analysis were to be performed on data with higher spatial and temporal resolution (such as WiFi data). At smaller scales, the scientific community investigated the potential of WiFi for both indoor [4, 20, 43] and outdoor [8, 14, 22, 38] localization. Our recent work investigates how large companies can crowd source the creation of databases with router locations [14, 38, 45] and how people's mobility on societal scale can be described using only a small subset of available routers [46]. WiFi signals can also be analyzed to discover places of interest and stop locations in an unsupervised manner, *i.e.* without explicit location information as reference [26, 63].

We reiterate that the work presented in this article does not rely on location estimation (in terms of geographical coordinates) but instead on relative comparison between the environments sensed by two parties.

Proximity and interactions. Complementary to mobility, the question of person-to-person proximity has been recently considered in various contexts, with the results indicating that collection of high-resolution behavioral traces is instrumental for understanding of complex processes in society [12, 49, 53, 55]. However, from a technical point of view, collection of such data remains a challenge. Popular approaches to sensing proximity can be divided in three categories: (1) *peer-to-peer*, in which devices sense each other and then report the proximity data to a central entity, (2) *peer-to-environment*, in which devices sense the presence of beacons, report the measurements to a central entity, and the proximity is estimated by comparing the list of available beacons, (3) *environment-to-peer*, in which beacons, controlled by a party interested in measuring proximity, sense the presence of mobile devices. In this work we focused on the second approach (peer-to-environment), but in this section we describe the work related to all three categories of methods.

Peer-to-peer sensing can be realized for example using specialized hardware (*e.g.*, sociometric badges) [39, 44] or Bluetooth enabled smartphones [2, 12, 56]. In case of badges, proximity is usually inferred using radio-frequency identification (RFID) transmissions or infrared. This way, badges worn around participants' necks can usually sense not just proximity but also whether individuals are facing each other, resulting in recordings of face-to-face interactions. Sensing performed using Bluetooth-enabled mobile phones is less granular. Proximity can be detected in a binary fashion or further refined using the received signal strength as a proxy for distance [48]. However, the orientation of the individuals can not be sensed. The subjects' devices must remain in Bluetooth-discoverable state, which raises a number of security and privacy concerns, as described in the Introduction. There has been studies aiming to substitute Bluetooth with WiFi, an approach in which one phone acts as a hotspot and is sensed by others [7]. In controlled test environments this approach appears to offer a distance estimation resolution of 0.5 m [40], providing a better understanding of the nature of the contacts [21]. However, the claim has not been tested in the wild and the method potentially introduces even more privacy and security problems than Bluetooth.

An alternative way of sensing proximity between two persons with smartphones relies on comparing the two devices' radio frequency perceptions of the environment, *peer-to-environment sensing*. If a similarity is above a certain threshold, the two devices are assumed to be in physical proximity. The idea of comparing WiFi signals to measure proximity was initially explored more than a decade ago. Initially, researchers relied on single-feature measures of similarity, such as Manhattan distance [37] or overlap [36]. NearMe project [29] introduced more features, such as rank correlation between the lists of overlapping routers sorted by signal strength, Euclidean distance, and the number of non-overlapping APs. The authors explored combining the features into a regression model, but this approach did not outperform single features. Moreover, their model overfits for the rooms where it was trained and thus under-perform in previously unseen environments. In fact, Kjærgaard and Nurmi name differences in environments where the sensing takes place among the most important obstacles in using WiFi for social sensing [27]. Carlotto *et al.* combine a number of previously suggested features using a Gaussian Mixture Model and claim that their model is not environment-dependent (performs equally well in both buildings where it was tested) [6]. We note that the differences in environments can actually be used to increase the performance of the model. We can exploit the characteristics of human proximity patterns: from a technical standpoint, environments with a smaller number of routers offer lower accuracy of distance estimation; however, two people in an environment with fewer access points are more likely to be actually interacting (see Figure 2).

Finally, an *environment-to-peer sensing* system can be realized using a set of radio beacons (for example WiFi, GSM, or Bluetooth) that detect the presence of mobile devices. In a simple implementation of this approach proximity is assumed whenever two devices are sensed by the same beacon. This method has been applied to study urban space utilization [28, 41], traffic optimization [23], congestion in security-critical areas [5], as well as group dynamics during festivals [30], conferences [24], and other events [51].

6 DISCUSSION

In this paper we evaluated the applicability of WiFi for sensing proximity between pairs of individuals. Using WiFi signals allows us to mitigate a number of problems associated with Bluetooth sensing described in the Introduction:

- **usability** - the WiFi approach does not rely on discoverability of devices, and maintaining its operability does not require user interaction;
- **privacy** - as long as the operating system ensures MAC randomization, 3rd parties cannot track the location of the devices as it is the case with Bluetooth discoverable phones;
- **security** - as long as the phone is configured to not connect to open WiFi networks WiFi scanning does not introduce security problems;
- **energy expenditure** - Android phones by default scan for WiFi at all times, even when the user disables WiFi. On the other hand, scanning for Bluetooth requires more energy than the system would spend anyhow, and remaining Bluetooth discoverable is also associated with increased energy consumption [15]. Moreover, while in some phones Bluetooth power draw during scan is lower than that of WiFi, the total energy spent on a Bluetooth scan is higher than the cost associated with a WiFi scan, because it takes much longer [33].

The idea of exploiting WiFi signals for this purpose is not new. However, this work still has important contributions to offer. **First**, our proposed model outperforms the models proposed earlier. **Second**, to the best of our knowledge, researchers have not yet tested this approach in practice, on a large population. Here, we use a longitudinal dataset to confirm the findings of the researchers who had work on the problem before, and improve on the methods they proposed. **Third**, major research on the topic was performed before both smartphones and WiFi routers became wide-spread. Moreover, currently sold phones cannot (and should not) remain Bluetooth discoverable. We show that it is now possible to use the WiFi based approach to measure proximity in various environments.

6.1 Privacy implications

There are two main privacy implications of this work.

First, the ability to track person-to-person proximity using WiFi can help us move away from relying on Bluetooth. By not requiring the participants' phones to remain Bluetooth discoverable we protect the privacy and security of the subjects. While currently most phones advertise their presence and identity by scanning for WiFi, this problem is being addressed. Both Android and iOS randomize the MAC address of the device every time it sends WiFi probe requests making it more difficult to identify the user.[3]

Second, our results indicate a potential erosion of privacy of Android users. As we have previously shown, WiFi can be efficiently used for high-resolution mobility tracking of entire populations [45, 46, 63]. Here we go a step further and infer who people are in close proximity with, not only where they are. Thus, results of

[3] The randomization can only happen when the device is not connected to any WiFi network. When it is, it announces its real MAC address in each probe request.

WiFi scans—collected by major manufacturers of mobile devices and available to majority of mobile application developers—constitute highly sensitive datasets. For example, a vast majority of the applications available in Google Play Store has access to WiFi information, including all the scan results requested by the system as often as every 15 seconds [46]. This problem is addressed in Android 6.0 and later. In the latest versions of the system an application has to hold a location permission to listen to WiFi scan results. However, the vast majority of handsets currently in use will not receive these crucial updates. Thus, WiFi signals remain a major privacy risk for years to come.

6.2 Limitations of Bluetooth as ground truth for proximity

In this work we assume that a person-to-person proximity event happens between two people when their devices detect each other using Bluetooth. This approach has been popularly used in various Computational Social Science deployments [12, 56, 57] but is not without limitations. **First**, human body has a high absorption coefficient for the frequencies at which Bluetooth operates (2.4 GHz). Therefore, the Bluetooth signal is attenuated more in dense environments, proximity events might yield low received signal strengths, and some may fail to be registered altogether [62]. **Second**, two proximate phones, both in discoverable mode, may fail to capture each other's presence during a Bluetooth scan. [3]

Due to lack of dependable ground truth we cannot claim that our proposed method fares better in crowded spaces than Bluetooth. We observe that the model's estimations diverge from the Bluetooth ground truth more in denser environments, but another study with an alternative ground truth would be necessary to determine which approach is more precise.

6.3 Limitations of the WiFi-based social inference

While our approach to inference of proximity using WiFi signals offers an important new method in computational social science, we want to recognize its limitations. The inference in the approach presented here depends on the WiFi routers being present in the environment. While today WiFi networks are nearly omnipresent, especially in densely-populated areas [46], we find that in our longitudinal and diverse dataset approximately 5% of the WiFi scans did not report any nearby networks, preventing inference of physical proximity.

In this study, all phones collecting data were of the same make and model. When considering a broader application of the method, differences in WiFi hardware transmitters and firmware and software of the phones may result in less consistent scan data, making it more difficult to devise a robust model as the one presented here.

Furthermore, due to the lack of ground truth data, we cannot prove that our model accurately estimates the distance between users. We show, that our model is more likely to recognize proximity with a higher Bluetooth RSSI, but this property does not trivially translate to distance estimation. It is also important to stress that while we can reliably detect person-to-person proximity, we are unable to claim that these proximity events result in actual social interactions. However, observed over longer periods these proximity events might hint at the social structure of the cohort [48].

Finally, we note that it is not our argument that the values of all model features for discovering proximity events are generally applicable to different populations. Depending on the specific population and social context under consideration, the weights in the model might be different or even entirely new features might be useful. Our results indicate, however, that physical proximity can be inferred in a feasible fashion using WiFi signals collected by smartphones, even in very densely-connected populations.

6.4 Future directions

In this work we focus on detecting physical proximity of WiFi-enabled devices by comparing the lists of available routers and their measured signal strenghts (RSSI). Two limitations of this approach could potentially be resolved in the future if the firmware and public APIs of smartphones are extended. First, the method fails if there are no proximate WiFi routers. However, even in such environments smartphones search for routers by sending out probe requests. These requests can be sensed by other smartphones (in the same fashion they are normally detected by WiFi routers) and used to approximate distance. While an Android smartphone in the hotspot mode detects these probe requests, the data is not available to user applications. Second, RSSI-based estimation of distance is subject to significant error. On the other hand, WiFi-enabled devices also perform much more detailed mesurements of channel state information (CSI). CSI has been successfully employed for decimeter-level localization [65] as well as activity recognition [60, 61]. Again, the CSI data is not currently available through standard Android API.

7 CONCLUSION

In this work we showed how WiFi scan results can reveal our daily proximity with others and our social ties. By using behavioral traces, placed in context through meta information and our basic understanding of the inner working of social systems, we can transform a noisy data source to a strong social signal. Our findings have important privacy implications, especially given our previous work which shows that it is possible to use WiFi signals for tracking human mobility. On the other hand, WiFi scans also constitute a great opportunity for companies with access to such data on a global scale, to contribute *e.g.*, better epidemic models built on proximity data of billions of people. Finally, we hope that this method of social sensing will substitute Bluetooth sensing in future Computational Social Science deployments.

REFERENCES

[1] Lada A Adamic and Eytan Adar. 2003. Friends and neighbors on the web. *Social networks* 25, 3 (2003), 211–230.

[2] Nadav Aharony, Wei Pan, Cory Ip, Inas Khayal, and Alex Pentland. 2011. Social fMRI: Investigating and shaping social mechanisms in the real world. *Pervasive and Mobile Computing* 7, 6 (2011), 643–659.

[3] Giuseppe Anastasi, Renata Bandelloni, Marco Conti, Franca Delmastro, Enrico Gregori, and Giovanni Mainetto. 2003. Experimenting an indoor bluetooth-based positioning service. In *Distributed Computing Systems Workshops, 2003. Proceedings. 23rd International Conference on.* IEEE, 480–483.

[4] Paramvir Bahl and Venkata N Padmanabhan. 2000. RADAR: An in-building RF-based user location and tracking system. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, Vol. 2. Ieee, 775–784.

[5] Darcy Bullock, Ross Haseman, Jason Wasson, and Robert Spitler. 2010. Automated measurement of wait times at airport security: deployment at Indianapolis international airport, Indiana. *Transportation Research Record: Journal of the Transportation Research Board* 2177 (2010), 60–68.

[6] Alessandro Carlotto, Matteo Parodi, Carlo Bonamico, Fabio Lavagetto, and Massimo Valla. 2008. Proximity Classification for Mobile Devices Using Wi-fi Environment Similarity. In *Proceedings of the First ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments (MELT '08)*. ACM, New York, NY, USA, 43–48. DOI:http://dx.doi.org/10.1145/1410012.1410023

[7] Iacopo Carreras, Aleksandar Matic, Piret Saar, and Venet Osmani. 2012. Comm2Sense: Detecting proximity through smartphones. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on.* IEEE, 253–258.

[8] Yu-Chung Cheng, Yatin Chawathe, Anthony LaMarca, and John Krumm. 2005. Accuracy Characterization for Metropolitan-scale Wi-Fi Localization. In *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services (MobiSys '05).* ACM, New York, NY, USA, 233–245. DOI:http://dx.doi.org/10.1145/1067170.1067195

[9] Eunjoon Cho, Seth A Myers, and Jure Leskovec. 2011. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 1082–1090.

[10] Siraj Datoo. 2013. High street shops are studying shopper behaviour by tracking their smartphones or movement. http://goo.gl/vGg8k8. (2013). Accessed: 2017-01-13.

[11] Romain Dillet. 2014. Happn Is A Dating App Powered By Real Life Interactions. http://goo.gl/0nHyIr. (2014). Accessed: 2017-01-13.

[12] Nathan Eagle and Alex Pentland. 2006. Reality mining: sensing complex social systems. *Personal and ubiquitous computing* 10, 4 (2006), 255–268.

[13] Nathan Eagle, Alex Sandy Pentland, and David Lazer. 2009. Inferring friendship network structure by using mobile phone data. *Proceedings of the National Academy of Sciences* 106, 36 (2009), 15274–15278.

[14] Alan Eustace. 2010. WiFi data collection: An update. http://goo.gl/VFJ9mM. (2010).

[15] R. Friedman, A. Kogan, and Y. Krivolapov. 2013. On Power and Throughput Tradeoffs of WiFi and Bluetooth in Smartphones. *Mobile Computing, IEEE Transactions on* 12, 7 (July 2013), 1363–1376. DOI: http://dx.doi.org/10.1109/TMC.2012.117

[16] R. C. Gatej. 2013. *An adaptive approach to mobile sampling.* Master's thesis. Technical University of Denmark.

[17] Marta C Gonzalez, Cesar A Hidalgo, and Albert-Laszlo Barabasi. 2008. Understanding individual human mobility patterns. *Nature* 453, 7196 (2008), 779–782.

[18] Google. 2017. Enabling discoverability. goo.gl/ExuuVW. (2017). Accessed: 2017-01-13.

[19] Przemyslaw A Grabowicz, José J Ramasco, Bruno Gonçalves, and Víctor M Eguíluz. 2014. Entangling mobility and interactions in social media. *PLoS One* 9, 3 (2014), e92196.

[20] Andreas Haeberlen, Eliot Flannery, Andrew M. Ladd, Algis Rudys, Dan S. Wallach, and Lydia E. Kavraki. 2004. Practical Robust Localization over Large-scale 802.11 Wireless Networks. In *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking (MobiCom '04)*. ACM, New York, NY, USA, 70–84. DOI: http://dx.doi.org/10.1145/1023720.1023728

[21] Edward Twitchell Hall. 1966. The hidden dimension . (1966).

[22] Dongsu Han, David G Andersen, Michael Kaminsky, Konstantina Papagiannaki, and Srinivasan Seshan. 2009. Access point localization using local signal strength gradient. In *Passive and Active Network Measurement*. Springer, 99–108.

[23] Ross Haseman, J Wasson, and D Bullock. 2010. Real time measurement of work zone travel time delay and evaluation metrics using bluetooth probe tracking. *Journal of the Transportation Research Board* (2010).

[24] Pan Hui, Augustin Chaintreau, James Scott, Richard Gass, Jon Crowcroft, and Christophe Diot. 2005. Pocket switched networks and human mobility in conference environments. In *Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. ACM, 244–251.

[25] Lorenzo Isella, Juliette Stehlé, Alain Barrat, Ciro Cattuto, Jean-François Pinton, and Wouter Van den Broeck. 2011. What's in a crowd? Analysis of face-to-face behavioral networks. *Journal of theoretical biology* 271, 1 (2011), 166–180.

[26] Jong Hee Kang, William Welbourne, Benjamin Stewart, and Gaetano Borriello. 2004. Extracting places from traces of locations. In *Proceedings of the 2nd ACM international workshop on Wireless mobile applications and services on WLAN hotspots*. ACM, 110–118.

[27] Mikkel Baun Kjærgaard and Petteri Nurmi. 2012. Challenges for social sensing using wifi signals. In *Proceedings of the 1st ACM workshop on Mobile systems for computational social science*. ACM, 17–21.

[28] Vassilis Kostakos, Eamonn O'Neill, Alan Penn, George Roussos, and Dikaios Papadongonas. 2010. Brief encounters: Sensing, modeling and visualizing urban mobility and copresence networks. *ACM Transactions on Computer-Human Interaction (TOCHI)* 17, 1 (2010), 2.

[29] John Krumm and Ken Hinckley. 2004. The nearme wireless proximity server. In *UbiComp 2004: Ubiquitous Computing*. Springer, 283–300.

[30] Jakob Eg Larsen, Piotr Sapiezynski, Arkadiusz Stopczynski, Morten Mørup, and Rasmus Theodorsen. 2013. Crowds, Bluetooth, and Rock'N'Roll: Understanding Music Festival Participant Behavior. In *Proceedings of the 1st ACM International Workshop on Personal Data Meets Distributed Multimedia (PDM '13)*. ACM, New York, NY, USA, 11–18. DOI: http://dx.doi.org/10.1145/2509352.2509399

[31] David Lazer, Alex Sandy Pentland, Lada Adamic, Sinan Aral, Albert Laszlo Barabasi, Devon Brewer, Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann, and others. 2009. Life in the network: the coming age of computational social science. *Science (New York, NY)* 323, 5915 (2009), 721.

[32] Jinyang Li, John Jannotti, Douglas S. J. De Couto, David R. Karger, and Robert Morris. 2000. A Scalable Location Service for Geographic Ad Hoc Routing. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom '00)*. ACM, New York, NY, USA, 120–130. DOI: http://dx.doi.org/10.1145/345910.345931

[33] Kaisen Lin, Aman Kansal, Dimitrios Lymberopoulos, and Feng Zhao. 2010. Energy-accuracy Trade-off for Continuous Mobile Device Location. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services (MobiSys '10)*. ACM, New York, NY, USA, 285–298. DOI: http://dx.doi.org/10.1145/1814433.1814462

[34] Gilles Louppe. 2013. How are feature importances determined in Random Forest Classifier? http://stackoverflow.com/a/15821880. (2013). Accessed: 2015-10-17.

[35] Xin Lu, Linus Bengtsson, and Petter Holme. 2012. Predictability of population displacement after the 2010 Haiti earthquake. *Proceedings of the National Academy of Sciences* (2012). DOI: http://dx.doi.org/10.1073/pnas.1203882109

[36] Marvin McNett and Geoffrey M. Voelker. 2005. Access and Mobility of Wireless PDA Users. *SIGMOBILE Mob. Comput. Commun. Rev.* 9, 2 (April 2005), 40–55. DOI: http://dx.doi.org/10.1145/1072989.1072995

[37] Jean-Luc Meunier. 2004. Peer-to-peer determination of proximity using wireless network data. (2004).

[38] Bruce Meyerson. 2007. AOL introduces location plug-in for instant messaging so users can see where buddies are. http://goo.gl/2W1uYh. (2007).

[39] Daniel Olguín Olguín, Benjamin N Waber, Taemie Kim, Akshay Mohan, Koji Ara, and Alex Pentland. 2009. Sensible organizations: Technology and methodology for automatically measuring organizational behavior. *Systems, Man, and Cybernetics, Part B: Cybernetics,*

IEEE Transactions on 39, 1 (2009), 43–55.

[40] Venet Osmani, Iacopo Carreras, Aleksandar Matic, and Piret Saar. 2014. An analysis of distance estimation to detect proximity in social interactions. *Journal of Ambient Intelligence and Humanized Computing* 5, 3 (2014), 297–306.

[41] Eamonn OâĂŹNeill, Vassilis Kostakos, Tim Kindberg, Alan Penn, Danaë Stanton Fraser, Tim Jones, and others. 2006. Instrumenting the city: Developing methods for observing and understanding the digital cityscape. In *UbiComp 2006: Ubiquitous Computing*. Springer, 315–332.

[42] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[43] Nissanka B Priyantha, Anit Chakraborty, and Hari Balakrishnan. 2000. The cricket location-support system. In *Proceedings of the 6th annual international conference on Mobile computing and networking*. ACM, 32–43.

[44] Marcel Salathé, Maria Kazandjieva, Jung Woo Lee, Philip Levis, Marcus W Feldman, and James H Jones. 2010. A high-resolution human contact network for infectious disease transmission. *Proceedings of the National Academy of Sciences* 107, 51 (2010), 22020–22025.

[45] Piotr Sapiezynski, Radu Gatej, Alan Mislove, and Sune Lehmann. 2015a. Oportunities and Challenges in Crowdsourced Wardriving. In *Proceedings of the 15th ACM SIGCOMM conference on Internet measurement*. ACM.

[46] Piotr Sapiezynski, Arkadiusz Stopczynski, Radu Gatej, and Sune Lehmann. 2015b. Tracking Human Mobility Using WiFi Signals. *PLoS ONE* 10, 7 (07 2015), e0130824. DOI: http://dx.doi.org/10.1371/journal.pone.0130824

[47] Karen Scarfone and John Padgette. 2008. Guide to bluetooth security. *NIST Special Publication* 800 (2008), 121.

[48] Vedran Sekara and Sune Lehmann. 2014. The strength of friendship ties in proximity sensor data. *PloS one* 9, 7 (2014), e100915.

[49] Vedran Sekara, Arkadiusz Stopczynski, and Sune Lehmann. 2016. Fundamental structures of dynamic social networks. *Proceedings of the National Academy of Sciences* 113, 36 (2016), 9977–9982. DOI: http://dx.doi.org/10.1073/pnas.1602803113

[50] Chaoming Song, Zehui Qu, Nicholas Blumm, and Albert-László Barabási. 2010. Limits of predictability in human mobility. *Science* 327, 5968 (2010), 1018–1021.

[51] Hendrik Stange, Thomas Liebig, Dirk Hecker, Gennady Andrienko, and Natalia Andrienko. 2011. Analytical workflow of monitoring human mobility in big event settings using bluetooth. In *Proceedings of the 3rd ACM SIGSPATIAL international workshop on indoor spatial awareness*. ACM, 51–58.

[52] Juliette Stehlé, Nicolas Voirin, Alain Barrat, Ciro Cattuto, Lorenzo Isella, Jean-François Pinton, Marco Quaggiotto, Wouter Van den Broeck, Corinne Régis, Bruno Lina, and others. 2011. High-resolution measurements of face-to-face contact patterns in a primary school. *PloS one* 6, 8 (2011), e23176.

[53] Arkadiusz Stopczynski, Alex Sandy Pentland, and Sune Lehmann. 2015. Physical Proximity and Spreading in Dynamic Social Networks. *arXiv preprint arXiv:1509.06530* (2015).

[54] Arkadiusz Stopczynski, Riccardo Pietri, Alex Pentland, David Lazer, and Sune Lehmann. 2014. Privacy in Sensor-Driven Human Data Collection: A Guide for Practitioners. *CoRR* abs/1403.5299 (2014). http://arxiv.org/abs/1403.5299

[55] Arkadiusz Stopczynski, Piotr Sapiezynski, Sune Lehmann, and others. 2015. Temporal fidelity in dynamic social networks. *The European Physical Journal B* 88, 10 (2015), 1–6.

[56] Arkadiusz Stopczynski, Vedran Sekara, Piotr Sapiezynski, Andrea Cuttone, Mette My Madsen, Jakob Eg Larsen, and Sune Lehmann. 2014. Measuring Large-Scale Social Networks with High Resolution. *PLoS ONE* 9, 4 (04 2014), e95978. DOI: http://dx.doi.org/10.1371/journal.pone.0095978

[57] Aaron Striegel, Shu Liu, Lei Meng, Christian Poellabauer, David Hachen, and Omar Lizardo. 2013. Lessons Learned from the Netsense Smartphone Study. In *Proceedings of the 5th ACM Workshop on HotPlanet (HotPlanet '13)*. ACM, New York, NY, USA, 51–56. DOI: http://dx.doi.org/10.1145/2491159.2491171

[58] Jameson L Toole, Carlos Herrera-Yaqüe, Christian M Schneider, and Marta C González. 2015. Coupling human mobility and social ties. *Journal of The Royal Society Interface* 12, 105 (2015), 20141128.

[59] Olga Troyanskaya, Michael Cantor, Gavin Sherlock, Pat Brown, Trevor Hastie, Robert Tibshirani, David Botstein, and Russ B Altman. 2001. Missing value estimation methods for DNA microarrays. *Bioinformatics* 17, 6 (2001), 520–525.

[60] Hao Wang, Daqing Zhang, Junyi Ma, Yasha Wang, Yuxiang Wang, Dan Wu, Tao Gu, and Bing Xie. 2016b. Human Respiration Detection with Commodity Wifi Devices: Do User Location and Body Orientation Matter?. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '16)*. ACM, New York, NY, USA, 25–36. DOI: http://dx.doi.org/10.1145/2971648.2971744

[61] Wei Wang, Alex X. Liu, and Muhammad Shahzad. 2016a. Gait Recognition Using Wifi Signals. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '16)*. ACM, New York, NY, USA, 363–373. DOI: http://dx.doi.org/10.1145/2971648.2971670

[62] Jens Weppner and Paul Lukowicz. 2013. Bluetooth based collaborative crowd density estimation with mobile phones. In *Pervasive computing and communications (PerCom), 2013 IEEE international conference on*. IEEE, 193–200.

[63] David Kofoed Wind, Piotr Sapiezynski, Magdalena Anna Furman, and Sune Lehmann. 2016. Inferring Stop-Locations from WiFi. *PloS*

one 11, 2 (2016), e0149105.

[64] Ford-Long Wong and Frank Stajano. 2005. Location Privacy in Bluetooth. In *Security and Privacy in Ad-hoc and Sensor Networks*, Refik Molva, Gene Tsudik, and Dirk Westhoff (Eds.). Lecture Notes in Computer Science, Vol. 3813. Springer Berlin Heidelberg, 176–188. DOI: http://dx.doi.org/10.1007/11601494_15

[65] Zheng Yang, Zimu Zhou, and Yunhao Liu. 2013. From RSSI to CSI: Indoor Localization via Channel Response. *ACM Comput. Surv.* 46, 2, Article 25 (Dec. 2013), 32 pages. DOI: http://dx.doi.org/10.1145/2543581.2543592

Received December 2016; revised February 2017; revised April 2017; accepted April 2017