
Downloaded from 
https://kar.kent.ac.uk/63969/ The University of Kent's Academic Repository KAR

The version of record is available from 
https://doi.org/10.1145/3144457.3144500

This document version
Author's Accepted Manuscript

DOI for this version

Licence for this version
UNSPECIFIED

Additional information

Versions of research works

Versions of Record
If this version is the version of record, it is the same as the published version available on the publisher's web site. Cite as the published version.

Author Accepted Manuscripts
If this document is identified as the Author Accepted Manuscript it is the version after peer review but before type setting, copy editing or publisher branding. Cite as Surname, Initial. (Year) 'Title of article'. To be published in *Title of Journal*, Volume and issue numbers [peer-reviewed accepted version]. Available at: DOI or URL (Accessed: date).

Enquiries
If you have questions about this document contact ResearchSupport@kent.ac.uk. Please include the URL of the record in KAR. If you believe that your, or a third party's rights have been compromised through this document please see our Take Down policy (available from https://www.kent.ac.uk/guides/kar-the-kent-academic-repository#policies).
Next2Me: Capturing Social Interactions through Smartphone Devices using WiFi and Audio signals

Jon Baker  
University of Kent  
School of Engineering and Digital Arts  
Canterbury, United Kingdom CT2 7NT  
j.baker@kent.ac.uk

Christos Efstratiou  
University of Kent  
School of Engineering and Digital Arts  
Canterbury, United Kingdom CT2 7NT  
c.efstratiou@kent.ac.uk

ABSTRACT

Typical approaches in detecting social interactions consider the use of co-location as a proxy for real-world interactions. Such approaches can underperform in challenging situations where multiple social interactions can occur in close proximity to each other. In this paper, we present a novel approach to detect co-located social interactions using smartphones. Next2Me relies on the use of WiFi signals and audio signals to accurately distinguish social groups interacting within a few meters from each other. Through a range of real-world experiments, we demonstrate a technique that utilises WiFi fingerprinting, along with sound fingerprinting to identify social groups. Experimental results show that Next2Me can achieve a precision of 88% within noisy environments, including smartphones that are placed in users’ pockets, whilst maintaining a very low energy footprint (<3% of battery capacity per day).

CCS CONCEPTS

- Human-centered computing → Ubiquitous and mobile computing systems and tools; Collaborative and social computing devices;

KEYWORDS

Smartphone sensing, social sensing, WiFi, audio

ACM Reference format:

https://doi.org/10.1145/nmnnnn.nnmmnn

1 INTRODUCTION

Social interactions represent a significant part of our daily lives. They are considered a significant aspect of the quality of people’s daily lives [4], as well as an important activity that enables collaboration and creativity [15]. In recent years there has been an increasing interest in developing technologies that can capture the social behaviour of people. Within working environments, analysis of the social behaviour of employees has been shown to reflect the performance and productivity of teams [16]. Within the health and well-being domain, long-term tracking of social behaviour has been used as indicator for changes in mental health and perceived quality of life [20].

People-centric sensing technologies have been employed in a number of scenarios to develop systems that can passively capture social interactions. Wearable and mobile devices (e.g. smartphones) have been used to infer social behaviour by analysing the mobility patterns of individuals [17]. Traditionally, most of the approaches that have been used in such scenarios assume that proximity between individuals is an indicator of social interaction. This may be a valid assumption for certain situations (e.g. people participating in a meeting), but the assumption may not hold when considering situations where social interactions take place within crowded environments, involving multiple social groups. Such scenarios are very common in the daily lives of people. Having a chat in a crowded café, or interacting with different people during a networking session in a conference, are common situations where proximity may not be sufficient to identify correctly the people involved in an interaction.

In this work, we attempt to develop a system that can accurately capture social interactions within challenging scenarios where multiple social groups interact within close proximity of each other. In order to distinguish the closely located social groups, we rely on analysis of audio captured by the smartphones of the participants and aim to identify which social group they participate in. Our hypothesis is that the sound patterns captured by the smartphones of the people participating in a conversation is sufficiently different from the sound patterns of people not involved in that interaction event, or participating in a different social interaction even if the groups are within a few meters from each other. Intuitively, we consider that people participating in a conversation that takes place in a noisy or crowded environment have the tendency to raise their voices enough to be heard by the people involved in the conversation. This natural behaviour is enough to produce distinct sound patterns that are very similar for the people participating in the conversation, and sufficiently different from the sound patterns captured by the smartphones of people in near-by social groups. We demonstrate the design of a system that relies on a combination of WiFi fingerprinting and Audio fingerprinting captured by smartphones that are in the participants’ pockets, or on tables in front of them. Through a range of controlled experiments and a real-world deployment in a noisy café, we demonstrate that the system can achieve an average precision of 88%, while maintain a power consumption of less than 3% of the battery life per day.
2 RELATED WORK

Traditional techniques in capturing social interactions passively involve the use of RF technologies as a means of capturing co-location of users. Examples include systems that use Bluetooth on smartphones [12, 14] or specialised wearable RF tags [1] to detect proximity. Although such techniques can offer an approximation of the social behaviour of users, ultimately co-location does not always imply social interaction.

WiFi fingerprinting has been widely used as a way of localising users within buildings [7, 8]. As such the technique has also been used to detect co-location between users in environments where sufficient WiFi infrastructure is present [9]. However, such techniques suffer from similar limitations to other proximity-based approaches. Indeed co-location estimation does not imply social interaction. Recently there has been significant work on the capture of social interaction passively through the collection of WiFi traces of users’ phones using the WiFi infrastructure of a building [6, 22]. These techniques allow the tracking of social interaction without the need for the users to install a particular application on their phone. They allow the passive tracking of large numbers of users, but require access to the WiFi infrastructure of a given environment. In practical terms, these techniques can only be employed in certain environments, and do not allow the capture of social interactions throughout the daily lives of participants. Moreover, the passive tracking of smartphones without the need for an app installation raises significant privacy issues. Smartphone OS such as iOS have recently employed MAC obfuscation techniques to avoid such passive tracking thus rendering these approaches infeasible.

Since most social interactions contain speech between participants, it makes sense to use audio recordings as a technique for conversational detection. Some work uses the on-board microphones to record audio and use it for speaker recognition or conversational turns [5, 11, 17, 23]. More work uses audio for indoor localisation [19] or proximity detection [21]. The work done in DSP.Ear [5] presents a smartphone system that extracts emotions from speech signals, estimates the number of speakers in a room [23], detects the identity of speakers and identifies common ambient sounds. This type of work relies heavily on the use of machine learning techniques, requiring training of the voice recognition component through appropriate sound datasets, often involving sound samples by the user. In many scenarios such approach would be infeasible for large scale deployment limiting the applicability of the approach.

3 MOTIVATION

Current approaches in capturing social interactions tend to rely on secondary signals such as co-location, as proxy for an actual social interaction. Indeed, when individuals are close to each other there is a high probability that they are interacting with each other. However, there are numerous scenarios where such approaches can lead to erroneous results. People working in shared office environments may be co-located but not interacting; interacting with people in busy places, such as a restaurant or a social event, co-location may involve more people than those somebody is interacting with. In order to enable a more accurate detection of social interaction, there is a need to move beyond co-location.

Audio has been shown to be a more accurate indicator of actual social interaction, as a means of capturing the actual conversations of people involved. However, relying on heavy-weight speech recognition or speaker recognition requires personalised voice training [17]. Such approaches do not scale well, as machine learning algorithms need to be trained with voice sample of participants.

In this work we aim to develop a system that combines co-location and audio sensing to accurately detect social interactions in challenging environments. Such environments involve close co-location of social groups, interacting within busy environments. In our approach we do not require prior training of the system with audio samples, neither from the users or the environment. Instead, we rely on the comparison between sound signals captured by the users’ phones, as indicators of close proximity. Our motivation is based on the assumption that sound signals will have unique patterns for the people participating in a conversation, and are different from those in nearby conversations. Even in a noisy environment, people tend to talk louder to make sure that their conversation is heard by the participants from within the same group. Intuitively, we expect that sound samples captured by smartphones within a social interaction will have similar sound patterns, containing primarily the voices of the people participating. In our overall system, we utilise co-location as a means to trigger audio sensing when there is a high probability of social interaction. Sparserly sensed audio samples are then used to formulate a “sound fingerprint”. Comparisons between sound fingerprints are then used to discover the social networks of co-located users. The proposed approach does not require any special infrastructure, and can be used in any environment where there are sufficient WiFi signals to facilitate co-location sensing.

4 PRELIMINARY STUDY

In order to develop a robust system for capturing social interactions using smartphones, we initially attempted to explore the extent to which WiFi based proximity detection can enable the identification of such interactions. Furthermore, we also tried to explore how audio signals can be analysed to further assist in identifying social interactions. Our aim was to explore whether the combination of these two modalities can lead to a robust social sensing system that does not require prior training.

For the preliminary study, we developed a data collection Android application. The application was running continuously capturing WiFi and audio data. Specifically, the application scanned for available WiFi access points every 10 seconds, and recorded the MAC addresses of the access points and the RSSI value of the signal strength received by each access point. At the same time, the application recorded audio continuously for the duration of the experiment.

Our aim was to target “challenging” scenarios where groups of people interact within close proximity of each other. We set up two experiments: “Experiment 1” representing a typical social interaction of a single group during a meeting, and “Experiment 2” where two groups were interacting within the same room in close proximity. Specifically, the first experiment involved 10 participants joining a meeting and sitting around a large table. The participants were asked to keep their phones on the table during the meeting.
where each access point \( ap \) is identified by its MAC address, \( rss_i \) is the received signal strength value for \( ap \). We generate a WiFi fingerprint for the smartphone of each participant, by aggregating multiple WiFi scans over a sliding window of \( w = 60 \text{ s} \) with 33% overlap. Consider \( SW_t = \{ S_i : i \in (t - w, t) \} \) to be the set of subsequent scans within the window \( w \). The WiFi fingerprint at time \( t \) is:

\[
F_t = \{ ap_1 : rss_{t, 1}, \ldots, ap_n : rss_{t, n} \},
\]

where \( ap_i \in SW_t \), \( \forall ap_i, rss_i' = \frac{\text{avg}(rss_i : rss_i \in SW_t)}{\text{rss}_i^\text{max} + 100} \). As it was shown in [9], the RSSI values captured by different smartphone models can vary significantly even when collected under identical conditions. In order to allow appropriate comparisons between WiFi fingerprints, the recorded RSSI values are normalised and converted to positive scale, by dividing them with the maximum RSSI within the fingerprint:

\[
rss_i^n = \frac{rss_i' + 100}{\text{rss}_i^\text{max} + 100}.
\]

where \( rss_i' \) is the RSSI value for access point \( i \) and \( rss_i^\text{max} \) is the maximum RSSI value for the entries of averaged fingerprints.

The generated fingerprints are then used to estimate the proximity between participants. Specifically, fingerprints generated by different participants are compared using a similarity function. We assume that the level of similarity is an indication of proximity between these participants. We applied the Manhattan distance as a similarity function, as it demonstrated the highest level of discrimination between different co-location distances when compared to Euclidean distance and Tanimoto similarity. Specifically, for any two fingerprints that were compared, each fingerprint was extended by adding access points that only appeared in the other fingerprint. The added access points were given an RSS value of 0dB. The similarity metric between the fingerprints was given by the Manhattan distance, with an additional division of common count to provide scaling [7]:

\[
distance = \frac{1}{n} \sum_{i=1}^{n} |rss_i^a - rss_i^b|,
\]

where \( n \) is the number of elements in the intersection between the two fingerprint sets: \( n = |A \cap B| \), and \( rss_i^a \) and \( rss_i^b \) are the normalised RSS values for the access point \( i \) captured by the two devices \( a \) and \( b \).

We used the WiFi scanning dataset to estimate the distance metric of the participants in the two controlled experiments. In Experiment 1, all participants joined a group meeting in the same room. The pair-wise distance metric for all participants over time clearly shows that the WiFi fingerprint can identify the participants joining the meeting (Figure 3a). Indeed the WiFi fingerprint comparison can clearly capture the sequence of arrival of the participants for example. However, when exploring the WiFi fingerprint similarities during the meeting we can see significant variations although the participants did not change their location during the meeting (Figure 3b).

In our second experiment we attempt to explore how WiFi fingerprinting could be applied in case of co-located social interactions. In that experiment we have two groups of 3 people each, interacting in close proximity to each other (less 1 meter distance between the groups). Looking at the similarity of the WiFi fingerprints (Figure 3c), there is no obvious pattern that helps identify the two interacting groups. Furthermore, we explored the overall distribution of the similarity measurement (as Manhattan distance) between the pairs of participants that were interacting with each other, and compared it with the distribution of measurements between pairs
As amplitude alone was not considered a sufficient feature to identify social groups, we attempted to explore if sound signals can reveal distinctive patterns that can help differentiate between people participating in the same conversation. In Experiment 2
participants were asked to place their phones on the table in front of them (tables a and b), while 2 participants had phones placed in their pockets (Figure 2). We analysed the sound signals captured by these devices, extracted samples of audio of 2 seconds and applied a Fourier transformation (FFT) to look at the sound patterns in the frequency domain. We selected the frequencies between 300Hz and 3,400Hz, which is considered the speech range. This filtering allowed us to eliminate some of the noisy data that was captured by the phones in participants’ pockets. Figure 6 shows the frequency patterns from two devices participating in the same conversation (Figures 6a and 6b), and a device on a different conversation near by (Figure 6c). The patterns that we observe in this case show that devices that participate in the same conversation have high energies around the same frequencies, while the devices on different conversations show a significantly different pattern. Based on these observations, it is possible to devise a technique to extract a "sound fingerprint" that is based on the most significant frequencies of captured audio data, that can help distinguish between users participating in different conversations. Although the experiment involved participants in very close location, the difference in sound patterns can be explained by the natural tendency of people to speak loud enough so that all their conversation participants can hear them. This in practice ensures that the sound captured by the phones in close proximity to the conversation is dominated by the speakers participating in that particular group.

Based on the finding of these preliminary studies, we aimed to design a system to detect social interactions using a combination of WiFi signals, as early indicator that users are in close proximity, followed by audio sensing to identify smaller groups within the same area.

5 SYSTEM OVERVIEW

Next2Me is a mobile sensing system that can identify social interactions using WiFi and audio signals. The overall architecture can be seen in Figure 7. The system consists of sensing components running on a smartphone device, and a cloud service responsible for comparing datasets from multiple users. The system relies on WiFi fingerprinting to discover when a user is co-located with other users of the system. When co-location is detected, the participant’s smartphones are triggered to perform sound sensing. The sound sensing subsystem is responsible for discovering similarities in the sound patterns captured by the participating smartphones. The sound similarities are then used to identify a social network, as it is formed by the similarities of the sound signals. Finally, applying a community detection algorithm helps identify the sub-groups of people interacting within close proximity to each other. The following sections describe the system in more detail.

5.1 WiFi and co-location

The system relies on WiFi fingerprinting to detect co-location between users. Each smartphone device scans every 10s for near by WiFi access points transmitting at 2.4 GHz. Using the signals strength information gathered from near-by access points, we construct a WiFi fingerprint as it was described in Section 4.1. The aggregated WiFi fingerprints, containing the normalised average signal strength values of access points over a window of 60s, are uploaded to the cloud. A cloud service is then responsible for estimating if two devices are co-located. Specifically, an adjusted Manhattan distance metric (as shown in Section 4.1) is calculated over the WiFi fingerprints of smartphones that are potentially co-located (i.e. contain at least one common access point in their set). Subsequent WiFi fingerprints are generated every 2.5 mins.

Deciding if two users are potentially participating in a social interaction according to proximity depends on two parameters: the
estimated distance between them, and the duration that they are co-located. The selection of these parameters depends on the particular types of interactions that are targeted by the system. In this system, our objective is to capture significant social interactions that last for more than a few minutes. Specifically, we consider two devices to be co-located if the Manhattan distance is below a threshold of 0.8. Based on our preliminary experiments, that threshold was considered sufficient to discover co-location within less than 5m. Furthermore, if two users are co-located for more than a period of 5 mins, we consider this a potentially significant interaction. If these conditions are met, the cloud service triggers the co-located phones to initiate their sound sensing tasks.

5.2 Sound fingerprint

The preliminary analysis of audio signals showed that smartphones that are close to a social interaction can capture distinctive frequency patterns that can help distinguish the nearby social groups. In order to capture such patterns, we designed a technique that can capture a "sound fingerprint" that can represent the speech patterns detected over a time window of a few seconds. Our aim was to represent the sequence of sounds over that window as a fingerprint vector that can be easily compared with other fingerprints captured by nearby smartphones.

The sound sensing subsystem of the Next2Me system captures audio at a sampling rate of 16KHz. This allows a Fast Fourier transform (FFT) resolution of 8 kHz, but also provides a balance of higher quality signals. We use a window size of 2 secs, with a hamming window for calculating the FFT. The window of 2 secs was considered sufficiently large to allow more lenience with audio synchronisation across smartphones, considering that the on-board real-time clocks may not be perfectly synchronised. We extract the frequency bands between 300Hz and 3,400Hz which is the typical spectrum for human speech. Considering the results from the preliminary study, we can observe that frequency spectrums from devices around the same social interactions demonstrate high magnitudes around the same frequencies. Our aim is to use the significant frequencies in each sound sample as a way of comparing the sound patterns captured by different devices. As the sound capturing sensitivity varies across devices, we first need to reduce the variance on the sound spectrum that is produced. We apply a linearly weighted sliding average across the spectrum to smooth the results. Next we sub-sample the smoothed spectrum to reduce the granularity. Specifically, we use a 30Hz spectral window and calculate the average frequency magnitude for each window. The whole process produces a smoother frequency spectrum with 30Hz granularity. From this spectrum, we define as partial fingerprint the set of the top n frequencies with the highest magnitude.

In order to improve the robustness of the fingerprint against ambient noise, we produce the sound fingerprint for part of a social interaction by combining multiple partial fingerprints as a time series of sets with the top n frequencies (see Figure 8):

\[ S = \{P_1, P_2, \ldots, P_k\}, \text{ where } P_i = \{f_1, f_2, \ldots, f_n\} \]

Sound fingerprints captured by different devices can be compared by using the Jaccard index over their partial fingerprints. The Jaccard index \( \frac{\mid A \cap B \mid}{\mid A \cup B \mid} \) measures the similarity of two sets by estimating the number of common frequencies over the total number of unique frequencies in the two sets. We define the similarity function for two sound fingerprints as the average Jaccard index of their partial fingerprints:

\[
\text{sim}(S_a, S_b) = \frac{1}{k} \sum_{i=1}^{k} \frac{|P_i^a \cap P_i^b|}{|P_i^a \cup P_i^b|}
\]

The output of the comparison of sound fingerprints gives us a metric that represents the proximity of people according to the sounds captured by their phones.

5.3 Community detection

The sound fingerprints captured by the smartphones are uploaded to the cloud. A cloud service estimates the similarity metric between sound fingerprints of all the co-located devices. This similarity metric is then used to produce a weighted graph that represents the social network of all the co-located devices. It is expected that smartphones of users participating in the same social group will have a higher similarity (i.e. weight) in the graph. Using the social graph, we extract communities by applying the Louvain community detection algorithm [2]. In order to overcome the resolution limit issue experienced in modularity based community detection, we use the resolution limit technique described in [10]. We experimentally chose a limit of 0.8 to allow smaller communities to be identified. The output of the community detection represents the output of the system, identifying the different groups interacting within close proximity of each other (Figure 9).

5.4 Fine-tuning parameters

In order to analyse and fine-tune the parameters of the system we needed a scenario that involves a more complex setting than the preliminary studies. We conducted an additional study involving a social networking event (Experiment 3). We invited 7 participants (2 female, 5 male) to join a large meeting room and engage in a typical networking situation where they were asked to form smaller groups and freely discuss about their work (Figure 10). The participants...
Figure 9: Community network graph rendered with ForceAtlas 2, with data over time from 10, 20, 30, and 80 seconds. Taken from the interaction "Experiment 2" from the conference scenario in Table 1, nodes coloured by a modularity of 0.8.

Figure 10: Example layout for the conference scenario during experiment 1

Table 1: The groupings of participants during the networking experiment. Participants changed the formed groups three times during the experiment. All participants had their phones in their pockets.

<table>
<thead>
<tr>
<th>Stage</th>
<th>Groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(P1, P2, P3)</td>
</tr>
<tr>
<td>2</td>
<td>(P1, P2, P6, P7)</td>
</tr>
<tr>
<td>3</td>
<td>(P2, P6)</td>
</tr>
<tr>
<td></td>
<td>(P1, P7)</td>
</tr>
<tr>
<td></td>
<td>(P3, P4, P5)</td>
</tr>
</tbody>
</table>

Table: The groupings of participants during the networking experiment. Participants changed the formed groups three times during the experiment. All participants had their phones in their pockets.

In order to estimate the best parameters for the sound fingerprint, we selected a small sample of audio data captured in this experiment where there was definitely speech. Using a combination of numbers for the top frequencies, and the total duration of the fingerprint, we achieved the best results when selecting the top 6 frequencies for each partial fingerprint and maintaining a fingerprint window of 10 secs.

6 EVALUATION

In a real scenario, the Next2Me detection of social interactions does not necessarily need to run continuously through a co-location event. Instead, audio sensing can be used sparsely during a period to identify social groups. In our evaluation, we firstly attempted to estimate the average precision that can be achieved if sound fingerprinting is used only once during a social interaction using a 10 secs fingerprint. As it is shown in Figure 9, applying the sound fingerprinting technique at different intervals can have varying results. In order to estimate the average performance of the system, for each experiment we calculated the average performance for every 10 secs time window of the social interaction, using a 10 secs sliding window with 9 secs overlap.

For Experiment 2, two tables were positioned no more than 1m apart, phones were placed on the table, and two participants had additional phones in their pockets. We first estimated the average precision by including only the smartphones that were placed the desks, which resulted in 100% success rate (Table 2). This is a good result but somehow expected, considering that the experiment was performed in a quite environment, and the smartphones were at the centre of conversations that were taking place. When combining the system with the smartphones placed in pockets, the average precision dropped to 88.3%. Exploring the results, we could see that the location of one pocket smartphones was quite close to the second group, occasionally picking up stronger sound signals from the other table. Furthermore, the table itself acted as a barrier, blocking sound signals from the conversations reaching the pocket smartphones affecting the precision of the overall system.

In Experiment 3, where participants socialised within the same room, all smartphones were placed in participants’ pockets. We run our evaluation over the different stages where different groups were formed. The overall precision ranged from 74% to 91% (Table 2). Although these were relatively encouraging results, they all relied on capturing a single sound fingerprint during a social interaction.
We anticipate that precision on social interaction detection using whole experiment and estimated the average precision of the re-
with the duty-cycled scheme being applied at any time during the
W

a

b

1

k

k

i

b

k

i

a

b

time points, offers more chances to discover the correct social
group mapping. By applying a duty cycling scheme, there are ways
to potentially improve the overall precision of the system while
keeping the energy cost relatively low. Specifically, we explored the
effects of a fixed-length duty cycled sensing, where a fixed num-
ber of sound fingerprints can be captured during a potential social
interaction. When combining multiple fingerprints we wanted to
explore what is the number of consecutive sound fingerprints that
we should use to improve precision, and how the length of the
sleeping windows between them would affect the overall result.

Combining multiple fingerprints for the detection of social groups
would involve modifications in the way that the weights in the social
network graph are calculated. Specifically, when the social graphs
are formed, the weight between two nodes includes the average
fingerprint similarity over the number of sound fingerprints:

\[ W_{a,b} = \frac{1}{k} \sum_{i=1}^{k} \text{sim}(S_{i,a}, S_{i,b}) \]

where, \( W_{a,b} \) is the weight between participants \( a \) and \( b \), \( k \) is the number of sound fingerprints involved, and \( S_{i,a} \) is the \( i \)-th sound
fingerprint for participant \( a \). After weights are estimated combining multiple fingerprints, the same community detection algorithm is used to estimate the social groups that are formed.

Using the datasets from Experiment 3 (networking event), we
tested the performance of the system when using 2 or 3 sound fingerprints, with a varying sleeping windows between them; ranging from continuous (no sleeping) to fingerprints captured with a 60sec gap between them. We calculated the performance of the system, with the duty-cycled scheme being applied at any time during the whole experiment and estimated the average precision of the re-
results (Figure 11). The results show that using more than one sound
fingerprint improves the overall precision, while the combination of three fingerprints reduces the variance that we observed in preci-
sion. Generally, we see that combining multiple fingerprints with a sleeping window of 40sec offers the best results. Specifically, a duty
cycling scheme of 3 sound fingerprints with 40sec sleeping shows
an average precision of 0.92% and a combination of 2 sound finger-
prints with 40sec shows an average precision of 0.89%. Following
this, we conclude that for a setup of 3 samples/40sec sleeping is
appropriate for the high precision, while a 2 samples/40sec sleeping
scheme offers a good balance of energy cost and precision.

### 6.2 Coffee Shop scenario

As a final step in the evaluation of the Next2Me system, we performed a “real-world” deployment where a number of participants
where involved in social interaction within a busy coffee shop. Six
participants were invited to install the Next2Me application on
their phones. They were invited to meet in a busy coffee shop and
socialise, forming two separate social groups and sitting at nearby
tables (Figure 12). The event took place during a busy time where a
number of other people were in the coffee shop, engaged in con-
vosations. The setup was selected to ensure that the environment
involved ambient noise involving other people talking to each other.
During the event, the participants placed their phones on the table,
while two participants had a smartphone placed in their pocket.
Note that the table in this scenario had a relatively lower height
than the tables involved in previous scenarios.

The system was configured to perform WiFi scanning to detect
co-location, and trigger sound fingerprinting when participants
were co-located for more than 5mins. The overall experiment lasted
for 20 mins. We analysed the performance of the system using 3
sound fingerprints captured with a 40sec sleeping window between
them. Using all the devices involved in the scenario the system
achieves an average precision of 88%. When the pocket smartphones
are not included in the estimation, the precision raises to 99.1%. This
shows that smartphones situated without physical obstructions and
in an open environment will perform well, and smartphones in a
pocket will be clustered into communities with less precision due
to the frequency-filtering effect of the pocket material.

### 7 ENERGY CONSIDERATIONS

The design of the Next2Me system relies heavily on sensing modal-
ities that can have a significant impact on the battery life of the
participants’ smartphones. In this section we analyse the energy
cost implications of using Next2Me. In our analysis we attempt to
establish the average cost in the form of electric charge (measured
in mAh) consumed by the Next2Me during a day. This estimation
will allow us to consider the impact that the system would have on
the battery life of common smartphones, with battery capacities in
the rage of 2,800mAh (Samsung S5) to 3,220mAh (Nexus 6).

The WiFi fingerprinting subsystem relies on the periodic WiFi
scanning to discover near-by WiFi access points. If we consider
that the electric charge consumed during a WiFi scan is \( E_w \), a WiFi
fingerprint is generated using 6 scans, and a fingerprint is produced
every \( s_w \) seconds, the overall cost of continuously running the WiFi
scanning subsystem for a whole day is:

\[
W_{\text{total}} = \frac{86,400}{s_w}(6 \cdot E_w + N_w)
\]

where 86,400 is the total number of seconds in a day and \( N_w \) is the
average energy cost of uploading data to the cloud.

When a co-location incident is captured by WiFi scanning, and
it lasts for more than 5mins, the sound fingerprinting subsystem

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Precision</th>
<th>Correct</th>
<th>Incorrect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exp 2 - On table</td>
<td>1.00</td>
<td>66</td>
<td>0</td>
</tr>
<tr>
<td>Exp 2 - In pocket</td>
<td>0.88</td>
<td>99</td>
<td>13</td>
</tr>
<tr>
<td>Exp 3 - Stage 1</td>
<td>0.74</td>
<td>149</td>
<td>50</td>
</tr>
<tr>
<td>Exp 3 - Stage 2</td>
<td>0.91</td>
<td>238</td>
<td>21</td>
</tr>
<tr>
<td>Exp 3 - Stage 3</td>
<td>0.80</td>
<td>186</td>
<td>44</td>
</tr>
</tbody>
</table>

Table 2: Results for the two experiments involving social in-
teractions of groups in close proximity.
we performed a number of lab measurements to estimate the energy we can estimate the average energy for an individual who has on would typically be uploaded to the cloud, incurring an additional consumption. We used the Monsoon Power Meter setup to intercept performing the FFT respectively. The data from 3 sound /fingerprints is triggered. Each sound fingerprinting will involve 10sec of audio recording, and involves a CPU processing cost to perform an FFT over the sample. Subsequent sound fingerprints will then be uploaded to the cloud for comparison. We can model the additional energy for the sound /fingerprinting subsystem caused by a single social interaction as:

\[ S_{\text{int}} = 3 \cdot (E_{\text{sense}} + E_{\text{FFT}}) + N_s \]  

(2)

where \( S_{\text{int}} \) is the cost for a single social interaction, \( E_{\text{sense}} \) and \( E_{\text{FFT}} \) are the costs for capturing audio for a sound fingerprint, and performing the FFT respectively. The data from 3 sound fingerprints would typically be uploaded to the cloud, incurring an additional \( N_s \) cost for network communication. From Equations (1) and (2) we can estimate the average energy for an individual who has on average \( k \) significant social interactions during their day:

\[ E_{\text{total}} = W_{\text{total}} + k \cdot S_{\text{int}} \]  

(3)

In order to estimate the average energy of the Next2Me system, we performed a number of lab measurements to estimate the energy consumption. We used the Monsoon Power Meter setup to intercept the current drawn from the battery of a phone. We run experiments using the Samsung Galaxy J3 smartphone. A base line current when a phone is not performing any activities was estimated to be 9.16mA (2.27mA in airplane mode). When the phone was set to perform WiFi scanning, the average current during the scanning, without the baseline, was estimated to be 93.24mA. Each scan lasted for approximately 0.78s which results in an electric charge of \( E_{w} = 72.73\text{mAs} \). The average cost of data upload can vary significantly depending on the network infrastructure and external conditions. In order to estimate the impact of data upload using WiFi we use the energy cost per KB of 5mJ as it is estimated in [18] which results in consumed energy charge of \( N_w = 1.3\text{mAs} \). In the final deployment of the system we set the WiFi scanning subsystem to perform a scan once very 2.5 mins (which would enable the detection of 5min colocation instances). From equation 1 we can then estimate that in case of a user who does not have any significant interactions during the day, the overall energy cost is:

\[ W_{\text{total}} = \frac{86.400}{150} (436.38 + 1.3) = 252,138\text{mAs} = 70.03\text{mAh} \]

For a phone with a battery of 2,800mAh this would be 2.5% of the battery’s capacity.

In order to estimate the impact of sound fingerprinting, we calculated the average energy cost of audio sampling, and performing an FFT over a sound sample. The average current for audio sampling without the baseline was estimated to be 32.84mA. From equation 2 we can estimate the additional cost of detecting a single social interaction as:

\[ S_{\text{int}} = 3 \cdot (328.41 + 29.47) + 1.3 = 1074.94\text{mAs} = 0.29\text{mAh} \]

Assuming a case of a user who has about 20 significant interactions during the day, the additional energy capacity consumed by the system would be \( E_{\text{total}} = 70.03 + 20 \cdot 0.29 = 75.83\text{mAh} \). This results to 2.7% of the battery’s capacity. These results demonstrate that Next2Me has a very small impact on the smartphone’s battery life and would be appropriate for continuous sensing.


8 CONCLUSION

We developed a system that can use WiFi and Audio signals captured by smartphone devices, and requires no supervision or training. The proposed system detects the social interactions between people in various environments by capturing their co-location using WiFi, and then providing further confirmation by analysing the top magnitudes of speech frequencies. To estimate the allocation of people within different social groups, we use a community detection algorithm. Our technique achieved a high precision at low energy overhead, regardless of sound blocking material such as pockets, and can be robust to background noise.

REFERENCES


