Local Business Ambience Characterization Through Mobile Audio Sensing

He Wang
University of Illinois at Urbana-Champaign
Urbana Champaign, IL, USA
hewang5@illinois.edu

Dimitrios Lymberopoulos, Jie Liu
Microsoft Research
Redmond, WA, USA
{dlymper,liuj}@microsoft.com

ABSTRACT

Local search users today decide what business to visit solely based on distance information, and business ratings that can be sparse or stale. We believe that when users search for local businesses, such as bars or restaurants, they need to know more about the ambience of each business, such as how crowded it is, how loud and of what type the music it plays is, as well as how loud the human chatter in the business is. Unfortunately, this information doesn't exist today. In this paper, we propose to automatically crowdsource such rich, local business ambience metadata through real user check-in events. Every time a user checks into a business, the phone is in user's hands, and the phone's sensors can sense the business environment. We leverage the phone's microphone during this time to infer the occupancy and human chatter levels, the music type, as well as the music and noise levels in the business. As people check-in to businesses throughout the day, business metadata can be automatically updated over time, enabling a new generation of local search experience. Using approximately 150 audio traces collected from real businesses of various types over a period of 3 months, we show that by properly extracting the temporal and frequency signatures of the audio signal, it is feasible to train models that can simultaneously infer occupancy, human chatter, music, and noise levels in a business, with higher than 79% accuracy.

Categories and Subject Descriptors

C.3 [**Special-Purpose and Application-Based Systems**]: Miscellaneous

Keywords

Local Search; Business Ambience; Audio Sensing

1. INTRODUCTION

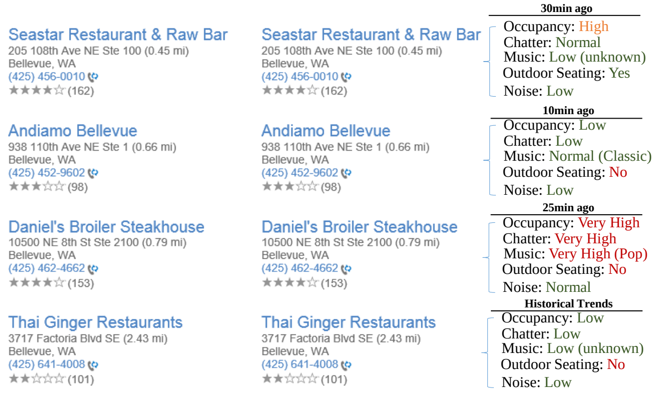
Users increasingly rely on their devices to search for local entities, typically businesses, while either on the go, looking for the next business to immediately visit, or at the comfort of their home planning their future activity. Despite the rapid growth and wide adoption of local search services, the current user experience is mainly inherited from traditional web search, and fails to meet users' needs. As Figure 1(a) shows, a typical local search result page consists of a list of businesses along with usually static business metadata, such as ratings, number of reviews, pricing and phone information. No information is provided about the ambience of the business, such as how crowded the business is, what type of music it plays, how loud the music is, or if outdoor seating is available at the business. This type of business ambience metadata has the potential to transform the local search experience by changing the way users select businesses to visit, and by changing the way local search algorithms rank businesses.

For instance, consider the search results shown in Figure 1(b) where rich ambience information is provided for each business. Different users can now evaluate the same results much more effectively according to their current context. A young professional that wants to go out for a happy hour with his colleagues can prioritize the crowded places with loud pop music. A father that looks for a restaurant to enjoy dinner with his family can prioritize the quiet Italian restaurant. At the same time, search engines can index this metadata to enable users to query local entities based on physical attributes. For instance, users could submit queries such as “Crowded bar playing loud pop music", or “Quiet Italian restaurants with outdoor seating". None of the commercially available search engines today have the intelligence to understand what a “crowded bar" or a “bar playing loud pop music" actually is.

The key to enabling this new generation of local search experience is the extraction of accurate business ambience metadata. Surprisingly, this is not a data mining or ranking problem, but rather a systems and data inferencing problem. Current search engines collect and index data that is available on the web through powerful web crawlers and data mining tools designed to leverage user-generated content, such as reviews [17, 9, 10, 5, 24, 20, 16, 22]. This approach fails when it comes to rich business metadata for three reasons. First, user reviews are sparse covering only a small subset of the existing businesses. Second, important information such as how crowded a business is, or how loud the music or the human chatter in the business is, might not be available in online user reviews. Third, this type of business metadata can change during the day or across days, making crawling this information even more difficult.

In this paper, we present the design, implementation, and evaluation of a scalable system for crowdsourcing business ambience metadata. In particular, we propose to leverage the mobile phones of users visiting businesses to extract the necessary ambience metadata in real time. As shown in Figure 2, every time a customer checks-in to a business on his mobile device (i.e., Foursquare, Facebook etc.), the device is placed in his hands for several tens of seconds. This is enough time for the phone to leverage its on-board

WWW'14, April 7–11, 2014, Seoul, Korea.
ACM 978-1-4503-2744-2/14/04.
http://dx.doi.org/10.1145/2566486.2568027.



Seastar Restaurant & Raw Bar
205 108th Ave NE Ste 100 (0.45 mi)
Bellevue, WA
(425) 456-0010
★★★★☆ (162)

Andiamo Bellevue
938 110th Ave NE Ste 1 (0.66 mi)
Bellevue, WA
(425) 452-9602
★★★☆☆ (98)

Daniel's Broiler Steakhouse
10500 NE 8th St Ste 2100 (0.79 mi)
Bellevue, WA
(425) 462-4662
★★★★☆ (153)

Thai Ginger Restaurants
3717 Factoria Blvd SE (2.43 mi)
Bellevue, WA
(425) 641-4008
★★☆☆☆ (101)

(a)

Seastar Restaurant & Raw Bar
205 108th Ave NE Ste 100 (0.45 mi)
Bellevue, WA
(425) 456-0010
★★★★☆ (162)

30min ago
Occupancy: High
Chatter: Normal
Music: Low (unknown)
Outdoor Seating: Yes
Noise: Low

Andiamo Bellevue
938 110th Ave NE Ste 1 (0.66 mi)
Bellevue, WA
(425) 452-9602
★★★☆☆ (98)

10min ago
Occupancy: Low
Chatter: Low
Music: Normal (Classic)
Outdoor Seating: No
Noise: Low

Daniel's Broiler Steakhouse
10500 NE 8th St Ste 2100 (0.79 mi)
Bellevue, WA
(425) 462-4662
★★★★☆ (153)

25min ago
Occupancy: Very High
Chatter: Very High
Music: Very High (Pop)
Outdoor Seating: No
Noise: Normal

Thai Ginger Restaurants
3717 Factoria Blvd SE (2.43 mi)
Bellevue, WA
(425) 641-4008
★★☆☆☆ (101)

Historical Trends
Occupancy: Low
Chatter: Low
Music: Low (unknown)
Outdoor Seating: No
Noise: Low

(b)

Figure 1: (a) Bing local search results (b) Local search results enhanced with rich business metadata.

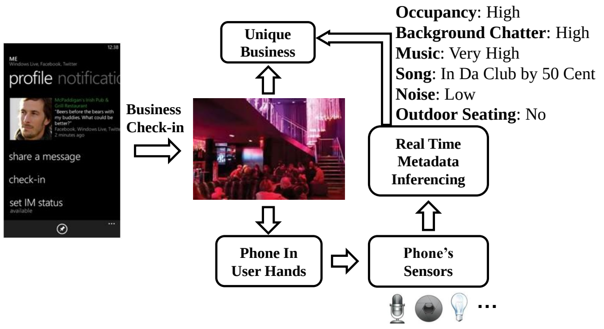


Figure 2: Automatically crowdsourcing business metadata through real user business check-ins.

sensors to sense the environment, and infer information about the current ambience of the business. The inferred information can be leveraged in two ways. Given a local search query, the most recent business metadata available can be surfaced in the search results (Figure 1(b)), enabling users to get a near real-time peek into the ambience of the business. When recent information about a business is not available, historical metadata can be accumulated and analyzed over time to enable forecasting of the business ambience at any given day and time. In essence, this rich business information can become for local search what traffic has become to maps. As people use near real-time or forecasted traffic information to navigate the road network, people can now use this rich business ambience metadata to navigate the local search results more efficiently and effectively.

To enable this experience, we leverage the phone's microphone to record the audio signature of the business. The on-board microphone is able to simultaneously capture music, noise and human chatter information, providing a rich base for ambience metadata extraction. We carefully analyze the magnitude and smoothness of audio signals at the temporal and frequency domain to extract features that capture the unique characteristics of the different audio sources, and then use these features to properly train individual models for classifying occupancy, background chatter, music, and noise levels across businesses. Intuitively, occupancy, even though not directly measured by audio, can be inferred by properly exploiting the audio characteristics of background human chatter to estimate its power, and then use this as a proxy to infer occupancy in the business. In general, the higher the background chatter, the higher the occupancy of the business. Furthermore, we train a model to accurately detect when people are talking near the phone, and use it to improve the resilience of the business metadata extraction process under this type of noise. Using more than 150 real business audio traces, collected over a period of 3 months through a variety of Windows Phone, Android, and iOS devices, we demonstrate that the proposed system can classify occupancy, background chattter, music, and noise levels with higher than 79% accuracy.

2. GOALS, CHALLENGES AND CONTRIBUTIONS

In this work, we consider the following types of business metadata:

Occupancy Level: an indication of how crowded the business is.

Music Level: an indication of how loud the music playing in the business is.

Background Human Chatter: an indication of how loud the background human chatter in the business is.

Noise Level: an indication of how noisy the business is in terms of noise coming from hardware equipment or the environment (e.g., fridge, nearby traffic etc.).

Outdoor Seating Option: an indication of the business offering outdoor seating or not.

Music Type/Song: the exact song or music type currently playing.

Note that detecting the music type or the exact song currently playing is something that can already been done accurately with freely available services such as Shazam [27], and SoundHound [28]. Furthermore, Zhou et al. [34] recently implemented a mobile service that leverages various sensors on the mobile device, including light, magnetometers, and cell reception, to reliably identify when the mobile device is indoors or outdoors. In light of this work, we focus on presenting only the results on accurately classifying the first four types of metadata. In particular, each type of metadata is classified into four different classes: Very High, High, Normal, and Low, according to the specification in Table 1.

2.1 Motivation - User Study

To better understand the value of the local business ambience metadata we propose to extract to real users, we conducted an online survey. 65 users with different backgrounds (technical people, housewives, teachers etc.), that were not familiar with this work, filled out the survey. The age and search profile of these 65 people are shown in Figure 3(a), and Figure 3(b) respectively. Even though our user study covered multiple aspects of local search, in the interest of space, we only show the results around rich business metadata extraction. Figure 3(c) shows the types of business metadata proposed in this work that users would find useful to see in their local search results (a user could select more than one metadata types). More than 90% of the users indicated occupancy, while approximately 50% of the users thought that outdoor seating and noise levels would also be useful. Figure 3(d), shows user responses when they were asked to choose the single most useful metadata type that would like to know at the time of their queries. Approximately 75% of the users chose business occupancy, while 6% thought that all of the proposed metadata would be useful. Only 10% of the users expressed no interest in knowing any of the proposed metadata. Figure 3 verifies our intuition and motivation behind this work. It shows that local search users highly value rich ambience metadata about local businesses, with occupancy information clearly being the most important.

2.2 Challenges and Contributions

Classifying business ambience metadata using audio data from the phone's microphone is a natural choice, but a challenging one.

Business Metadata	Class			
	Very High	High	Normal	Low
Occupancy	>80%	60%-80%	30%-60%	0%-30%
Background Chatter	Need to yell to be heard	Need to talk loud to be heard	Normal talking, clearly hear other people	Barely hear other people
Music Level	Need to yell to be heard	Need to talk loud to be heard	Normal talking, clearly listen to music	Barely hear music or no music
Noise Level	Loud noise	Loud enough to distract you	Typical indoor environmental noise	Barely hear any noise or no noise

Table 1: The different classes into which business metadata types are classified along with their descriptions.

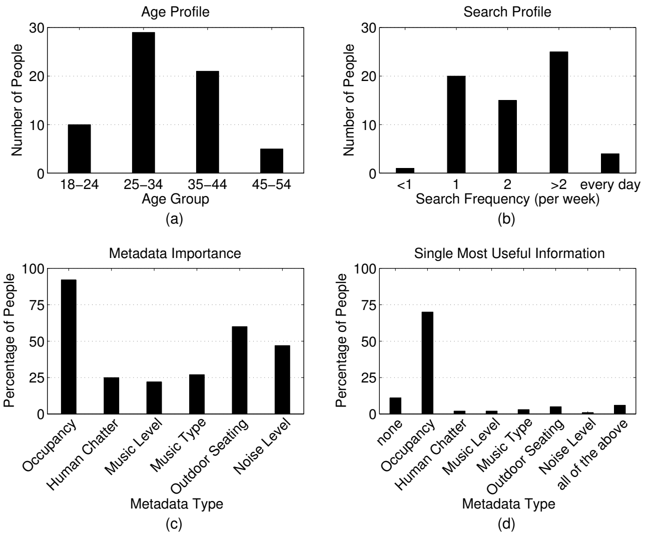


Figure 3: Online survey results from 65 participants.

First, the phone's microphone simultaneously samples all 3 different acoustic sources in the environment (human chatter, music, and noise), making it hard to distinguish and separately process each type of sound. Even in the case where the phone is equipped with dual microphones for noise cancellation purposes, sound source separation techniques [2, 21, 30] could only provide coarse grain separation such as background vs. foreground audio.

To address this problem, we carefully analyze the smoothness and amplitude of the audio signal in both the temporal and the frequency domain, to extract features that capture the unique characteristics of human chatter and music. By leveraging these distinctive features in the model training phase, we show that it is feasible to simultaneously, and accurately, classify occupancy, background chat, music, and noise levels based on the audio stream of a single microphone.

Another source of unreliability stems from the fact that people nearby the mobile device (i.e., the phone's owner) might be actively generating human chatter while audio is being recorded. Because of the proximity of the human chattering to the device, near-phone talking can overshadow background chatter in the business and could lead to erroneous classification.

To address this problem, we use the recorded audio data to train an additional model for identifying near-phone talking, and leverage it in two distinct ways. First, whenever near-phone talking is detected, the audio recording can be simply invalidated and not used for extracting business metadata to avoid erroneous classification. Second, we encode near-phone phone talking as an additional binary feature in the business metadata models we build, to enable high recognition rates even when near-phone talking is taking place. We demonstrate that we can detect near-phone talking with higher than 95% accuracy, and leverage it to improve recognition accuracy of business metadata models by up to 7.6%.

We evaluate our approach using approximately 150 audio traces, recorded in more than 50 unique businesses over a 3-month period. We leverage this dataset to exhaustively train more than 4000 decision tree models, and show that occupancy, human chatter, music, and noise levels can be successfully classified with higher than 79% accuracy. Furthermore, we collect additional real business audio traces using various Android, and iOS devices, and demonstrate that the proposed models can provide high detection accuracy across multiple hardware vendors and OS platforms.

3. ARCHITECTURE

The audio-based metadata extraction takes place in three steps. First, the raw audio signal is mapped into a limited number of discrete features that best describe its temporal and frequency structure. Then labeled audio traces recorded in multiple real businesses are leveraged to train a *single* model for each type of business metadata we need to infer. At run time, when a user checks into a business, audio is recorded on the user's mobile device, transformed into a feature vector, and through the trained models it is automatically labeled for each type of metadata. Next, we describe each of these steps in more detail.

3.1 Feature Extraction

Figure 4 provides an overview of the audio data processing pipeline. Initially, an audio stream of duration T seconds is recorded at a sampling rate of 16KHz on the user's mobile device.

The recorded audio stream is first split into T sequential segments of 1 second duration each. From the modeling perspective each of these 1 second segments is treated independently, in the sense that feature extraction and model training takes place for each 1 second segment separately. This segmentation of the audio stream is necessary to ensure high classification accuracy. Since the overall recording time can be in the orders of tens of seconds, the characteristics of the different sound sources can easily change multiple times during the recording, resulting into major variations in the recorded audio signals. For instance, initially people could be talking next to the phone while music is playing, and then for a few seconds only the background human chatter might be recorded. As a result, using lengthy audio segments that span multiple seconds can pollute the feature extraction process, which in turn can lead to erroneous classification. Segmentation of the audio trace helps remove these variations, and allows us to make more robust inferences over multiple shorter time windows during the recording.

During feature extraction, we examine the smoothness and amplitude of each 1-second segment at both the temporal and frequency domain. As Figure 4 shows, this is a two-step process. First, a large number of short-term features is computed over tiny sub-second audio windows. The duration of these windows varies depending on the exact feature that is being extracted. Second, a small number of long-term features is generated by examining the statistics of the short-term features over all sub-second windows within the 1 second segment. In particular, for every feature, we record its mean, minimum, and maximum values over all different sub-second windows, as well as its overall variation across these windows. This set of long-term features forms the actual feature vector that describes each 1 second segment.

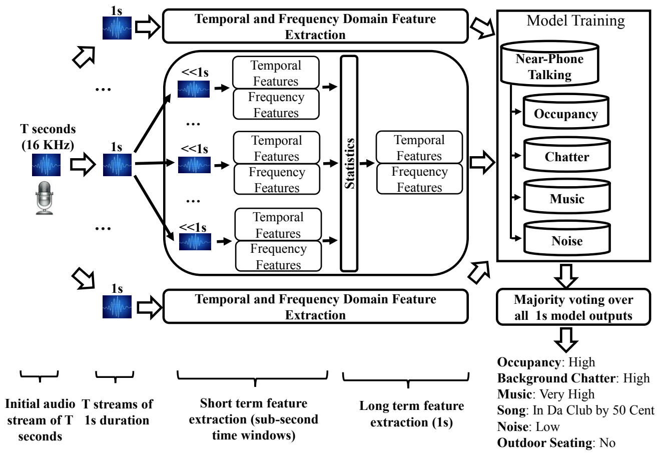


Figure 4: Processing pipeline for automatically inferring occupancy, background chatter, music, and noise levels from the audio stream of a mobile device. A single model for each of the four types of business metadata is trained using labeled data recorded at multiple businesses.

Section 4 presents the feature design process and the intuition behind it in more detail.

3.2 Metadata Model Training

Given the ability to map audio traces to feature vectors, we leverage manually collected, labeled data to train models for the different types of metadata we need to infer. This data is only needed to bootstrap the modeling process, and, as shown later in the paper, even a small number of traces (around 100) is sufficient to train accurate models. Each audio trace, and for each type of metadata we infer, is associated to one of 4 different labels: Low, Normal, High, and Very High (Table 1). For training and evaluation purposes, labels for occupancy, human chatter, music, and noise levels are provided for each audio recording. All labels are provided by human subjects during the time of the recording. At run-time, during actual business check-ins, there is no requirement for users to provide these labels. Note that labeled audio traces across multiple businesses are used to train a single model for each of the four types of metadata.

For each of the four models trained, the exact same process is followed. The only difference between models lies on the labels used each time. First, every audio trace is divided to multiple 1 second segments as shown in Figure 4. The labels associated to each of these 1 second segments are identical, and inherited from the initial audio recording. For instance, if an audio trace was labeled as "Lo Occupancy", then all 1 second segments derived from this trace are labeled as "Low Occupancy". Next, the temporal and frequency domain features are extracted for each 1 second segment. In that way, a collection of labeled feature vectors is generated. At this stage, various machine learning approaches to multi-class classification [33, 23, 1] can be applied to learn a mapping between feature values and actual labels representing the level of a specific business metadata type (Low, Normal, High or Very High). In this work, we leverage the WEKA learning toolkit [32] to train $C4.5$ decision tree models [25].

By leveraging real business audio traces, properly labeled with information of when people were talking near the phone or not, we also build an additional model for inferring near-phone talking. Audio trace segmentation, feature extraction and training for the near-phone talking model are identical to the rest of the models. The only difference are the labels (near-phone talking or not) assigned to the feature vectors during the training process. The output of the near-phone talking model is then directly fed as a binary feature to each of the four metadata models. In that way, we provide enough information in the training phase to enable models to adjust to near-phone talking audio traces, and maintain high recognition rates. Alternatively, the audio traces where near-phone talking is detected could be ignored to avoid erroneous classification.

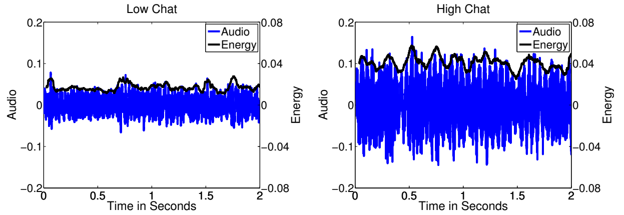


Figure 5: Two instances of the recorded raw audio signal and the computed energy corresponding to low and high background human chatter in a typical restaurant business.

3.3 Run-time Metadata Extraction

Having trained decision tree models for each type of business metadata, we can infer the level of occupancy, human chatter, music, and noise for any business audio trace. In particular, we first split the audio trace into 1 second segments, and compute all the temporal and frequency domain features for each of these segments as in the model training phase. The trained models can take the computed feature vector for each 1 second segment as input, and probabilistically map it to one of the pre-specified labels (Low, Normal, High or Very High). The label with the highest probability is assigned to each 1 second segment, and then majority voting across all of the 1 second segments is applied to infer the level of each business metadata type for the specific audio recording (Figure 4).

4. FEATURE DESIGN

The feature design process exploits the temporal and frequency signature of the audio stream to extract unique information of the different acoustic sources that are simultaneously recorded.

4.1 Temporal Domain Features

At the temporal domain, the recorded audio stream describes the amplitude of the audio signal over time (Figure 5). *Absolute* amplitude information is key for estimating the loudness of the audio signal, either this is music, human chatter or people talking right next to the phone. For instance, Figure 5 shows the recorded audio signal from a real business when there is low and high background human chatter. The amplitude difference in the two cases is clear. To capture this effect, we calculate the energy E of the recorded audio signal as the root-mean-square of the audio samples s_i:

$$E = \sqrt{\frac{\sum_{i=1}^{N} sign(s_i)^2}{N}} \qquad (1)$$

where N is the total number of samples. Energy is calculated over sliding windows of 50ms duration. Given the 16KHz microphone sampling rate, an energy value is computed a total of 15201 times within each 1 second audio segment (Figure 4). As shown in Figure 5, Equation 1 can accurately capture the absolute amplitude of the recorded audio signal over time.

Relative amplitude information, and in particular the smoothness of the amplitude over time, can also provide deep insights about the recorded audio signal. For instance, consider Figure 6 that shows the recorded audio when a person says the word “SO" in front of the phone, and when music only is playing in the background. Along

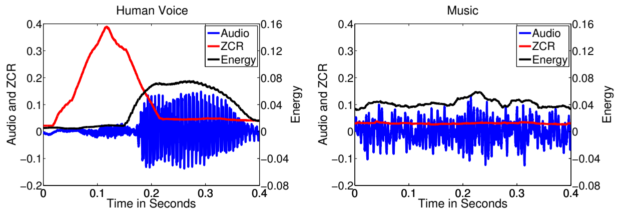


Figure 6: Raw audio, energy and zero cross rate (ZCR) when a human says the word “SO" and when only music is playing.

with the raw audio signal, the energy and zeros cross rate (ZCR) are shown. In the former case, pronunciation of the constituent “S" produces a low amplitude signal and a high ZCR, while pronunciation of the vowel “O" produces a high amplitude signal and a low ZCR. As a result, the combination of the two in one word generates large variations in both the energy and the zero cross rate of the recorded audio signal. In general, human talking consists of continuous repetitions of such constituents and vowels resulting into audio signals with high energy and ZCR variations in short time windows. On the other hand, as Figure 6 shows, background music recording corresponds into a far smoother audio signal. The energy and ZCR variations in the signal are almost negligible compared to the ones when people are actually talking. Note that this difference holds even when comparing a person talking and singing the exact same word or sentence. The energy variance during normal human speech is significantly smoothed out during singing because of the different pronunciation of constituents and vowels (i.e., prolonging vowels etc.).

To capture this fundamental difference between human chatting and music in the audio signal, we compute ZCR, as well as the variation of both ZCR and energy (ZCR_{var}, E_{var}):

$$ZCR = \frac{\sum_{i=2}^{N} |sign(s_i) - sign(s_{i-1})|}{2} \qquad (2)$$

Similar to energy, ZCR is computed over 50ms sliding time windows resulting into the computation of 15201 ZCR values over each 1 second audio segment. On the other hand, ZCR_{var} and E_{var} are computed using $500ms$ overlapping time windows with a step size of $100ms$, resulting into the computation of 5 ZCR_{var} and 5 E_{var} values. As a result, a total of 30412 features are computed for each 1 second segment in Figure 4:

$$\begin{aligned} F_{\text{s.t.}}^{temp.} \;=\; \{ & E_{\text{s.t.}} = [E^1, \ldots, E^{15201}], \\ & ZCR_{\text{s.t.}} = [ZCR^1, \ldots, ZCR^{15201}], \\ & E_{\text{s.t.}}^{var} = [E_{VAR}^1, \ldots, E_{VAR}^5], \\ & ZCR_{\text{s.t.}}^{var} = [ZCR_{VAR}^1, \ldots, ZCR_{VAR}^5]\} \end{aligned} \qquad (3)$$

$F_{\text{s.t.}}^{temp.}$ represents all the short-term features generated from the sub-second audio segment processing (Figure 4). These features are not directly used as input to the model training stage. Instead, statistics for each of the 4 different types of short-term temporal features are computed. More specifically, the minimum, maximum, mean, and variation values of energy, ZCR, ZCR_{var} and E_{var} are computed over all values in Equation 3:

$$\begin{aligned} F_{\text{l.t.}}^{temp.} \;=\; & \{\{min, max, mean, var\}(E_{\text{s.t.}}), \\ & \{min, max, mean, var\}(ZCR_{\text{s.t.}}), \\ & \{min, max, mean, var\}(E_{\text{s.t.}}^{var}), \\ & \{min, max, mean, var\}(ZCR_{\text{s.t.}}^{var})\} \end{aligned} \qquad (4)$$

$F_{\text{l.t.}}^{temp.}$ contains exactly 16 long-term features for each 1 second segment. This set of temporal features represents the temporal signature of each 1 second segment, and is used as input during the model training phase.

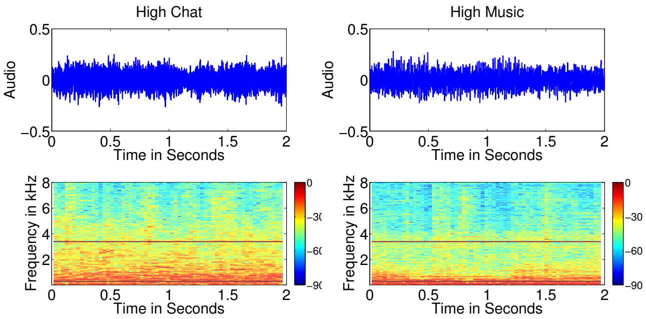


Figure 7: Spectrogram of two audio traces labeled as "High Chat" and "High Music". The magnitude of the audio signal across frequencies and over time is represented by different colors. The two solid horizontal lines at $300Hz$ and $3.4KHz$ indicate the voice frequency range used in telephony.

4.2 Frequency Domain Features

Similar processing of the audio signal can be applied in the frequency domain to analyze the magnitude of the audio signal across frequencies and its smoothness over time. These features can capture parts of the underlying structure of the audio stream that temporal features might not be able to accurately capture.

At the frequency domain, we first calculate the spectrogram of the recorded audio stream. In essence, the spectrogram describes the magnitude of the audio signal at different frequencies over time, and forms the basis for feature extraction. To better illustrate the rich information that spectrogram encodes, consider Figure 7 where the spectrograms for two audio traces recorded at different businesses are shown.

Note that even though the labels on the two traces are different ("High Chat" vs. "High Music"), the recorded raw audio signals appear to be very similar. However, when considering the spectrogram of the two traces, their differences become immediately clear. Human chatter translates to significantly higher magnitude signals in frequencies higher than $300Hz$. In addition, when looking over time, the spectrogram in the case of human chatter exhibits significantly higher spectral density between $300Hz$ and $3.4KHz$, compared to the High Music trace in Figure 7. Note that this frequency range, indicated by the two solid horizontal lines in Figure 7, is the actual voice frequency range used in telephony. This is an example of how the frequency signature of the audio signal can provide rich information about the signal's underlying structure, that the temporal signature might not be able to accurately capture.

Directly encoding spectrogram as a feature is not a scalable approach as a large number of features would be generated, posing stringent restrictions on data collection and model training. Instead, we leverage spectrogram's building components to extract a small, yet powerful feature set. In particular, for each 1 second segment in Figure 4, we calculate a 512-point FFT of the audio signal (32ms time window given the 16KHz microphone sampling rate) in 31 non-overlapping windows. For each of the 31 FFTs, we delete the DC component and normalize the remaining frequency bins such that the sum of squares equals to one. We use $p_t(i)$ to denote the magnitude of ith frequency bin of the normalized FFT at time t. We then summarize spectrogram by computing Spectral Centroid (SC), Bandwidth (BW), and Spectral Flux (SF) as follows:

$$SC = \frac{\sum_{i=1}^{N} i * p(i)^2}{\sum_{i=1}^{N} p(i)^2}, N = 256 \qquad (5)$$

$$BW = \frac{\sum_{i=1}^{N} (i - SC)^2 * p(i)^2}{\sum_{i=1}^{N} p(i)^2}, N = 256 \qquad (6)$$

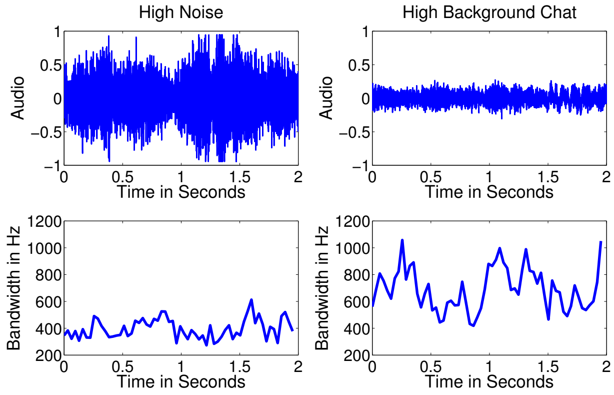


Figure 8: Raw audio signal and computed bandwidth when high noise and high background human chatter dominate the recording.

$$SF_t = \sum\nolimits_{i=1}^{N}(p_t(i) - p_{t-1}(i))^2, N = 256 \tag{7}$$

Both spectral centroid and bandwidth are computed for each one of the 31 FFTs over a single 1 second segment, resulting to 31 SC and 31 BW features. On the other hand, spectral flux is computed for every consecutive pair of FFTs resulting into 30 SF features for each 1 second segment.

Intuitively, spectral centroid represents the center frequency of the computed FFT and is calculated as the weighted mean of the frequencies present in the normalized FFT, with the magnitude of these frequencies as the weights. Bandwidth is a measure of the width/range of frequencies in the computed FFT. Finally, spectral flux represents the spectrum difference in adjacent FFTs and is an indication of the variation of spectral density over time.

To better understand the power of these features in the context of business metadata extraction, consider Figure 8, and Figure 9 showing raw audio signals recorded in real businesses under different conditions, and the corresponding frequency domain features. Figure 8 shows the raw audio signal from two businesses where high noise from equipment, such as ice-cream fridge, and high background human chatter are dominating each of the recordings. The calculated bandwidth information is able to clearly differentiate these two cases. In the case of background chatter, absolute bandwidth values and their variation over time are significantly higher than in the case of high noise from equipment.

Figure 9 shows the raw audio signal and the corresponding spectral flux in the cases where near-phone talking, high background chatter, and high background music are dominating the recording. Spectral flux values can consistently differentiate the various acoustic sources. High background music corresponds to noticeably lower spectral flux values as compared to high background human chatter. On the other hand, the mean of spectral flux values appears to be similar between near phone talking and high background human chatter, something that is expected given that in both cases human chatter is dominating the recording. However, the variation of spectral flux is considerably higher in the case of near phone talking when compared to background human chatter.

Besides spectral centroid, bandwidth, and spectral flux, we also compute the Mel-Frequency Cepstrum Coefficients (MFCC) [18]. MFCCs are coefficients that collectively make up an MFC which is a representation of the short-term power spectrum of a sound. MFC coefficients have been widely used in speech recognition [8, 31] and speaker identification [7, 11], and are considered high quality descriptors of human speech.

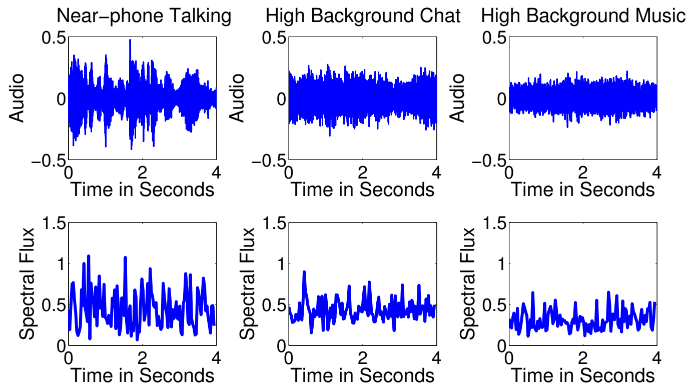


Figure 9: Raw audio signal and computed spectral flux when near phone talking, high background human chatter, and high background music dominate the recording.

To compute the MFC coefficients we use 256-sample sliding windows with a step size of 128 samples (given the 16KHz microphone sampling rate, this corresponds to 16ms windows with an 8ms step size). This results into 124 windows for each 1 second segment. For each window, we leverage the first 12 MFCC coefficients. We denote the ith MFCC coefficient at window t as $MFCC^t(i)$.

As a result, the set of short-term frequency domain features extracted over each 1 second segment contains exactly 1580 features:

$$\begin{aligned} F^{freq.}_{\text{s.t.}} \quad = \quad & \{SC_{\text{s.t.}} = [SC^1, \ldots, SC^{31}], \\ & BW_{\text{s.t.}} = [BW^1, \ldots, BW^{31}], \\ & SF_{\text{s.t.}} = [SF^1, \ldots, SF^{30}]\}, \\ & MFCC_{\text{s.t.}(i)} = [MFCC^1(i), \ldots, MFCC^{124}(i)], \\ & i = 1, \ldots, 12\} \end{aligned} \tag{8}$$

The long-term features that are eventually used during model training are computed directly from $F^{freq.}_{\text{s.t.}}$. Similarly to the temporal domain feature extraction, the minimum, maximum, mean, and variation values of SC, BW, and SF, as well as the mean values for each of the 12 MFCC coefficients are computed over all the short-term feature values in Equation 8:

$$\begin{aligned} F^{freq.}_{\text{l.t.}} \quad = \quad & \{\{min, max, mean, var\}(SC_{\text{s.t.}}), \\ & \{min, max, mean, var\}(BW_{\text{s.t.}}), \\ & \{min, max, mean, var\}(SF_{\text{s.t.}}), \\ & mean(MFCC_{\text{s.t.}}(i)), i = 1, \ldots, 12 \end{aligned} \tag{9}$$

$F^{freq.}_{\text{l.t.}}$ contains exactly 24 features for each 1 second segment. This set of long-term features represents the frequency signature of each 1 second segment, and is used as input during the model training phase.

4.3 Computation and Power Overhead

The long-term features extracted at the temporal ($F^{temp.}_{\text{l.t.}}$) and frequency ($F^{freq.}_{\text{l.t.}}$) domains form the feature set leveraged in model training. In total, 40 features are used for each 1 second segment: 16 temporal and 24 frequency domain features. Figure 10 provides an overview of the processing pipeline for generating all the short-term and long-term features.

Overall, 31992 short-term features need to be computed for each 1 second audio segment. Assuming an audio trace of 15 seconds, a total number of 479880 short-term features need to be computed. Even though this might seem as a computation stretch for a mobile

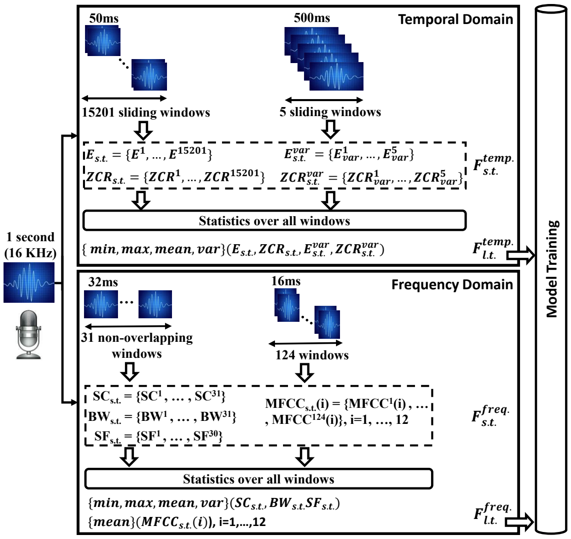

50ms
500ms
Temporal Domain
15201 sliding windows
5 sliding windows
$E_{s.t.} = \{E^1, \ldots, E^{15201}\}$
$E^{var}_{s.t.} = \{E^1_{var}, \ldots, E^5_{var}\}$
$ZCR_{s.t.} = \{ZCR^1, \ldots, ZCR^{15201}\}$
$ZCR^{var}_{s.t.} = \{ZCR^1_{var}, \ldots, ZCR^5_{var}\}$
$F^{temp.}_{s.t.}$
Statistics over all windows
$\{min, max, mean, var\}(E_{s.t.}, ZCR_{s.t.}, E^{var}_{s.t.}, ZCR^{var}_{s.t.})$
$F^{temp.}_{l.t.}$
1 second
(16 KHz)
32ms
16ms
Frequency Domain
31 non-overlapping windows
124 windows
$SC_{s.t.} = \{SC^1, \ldots, SC^{31}\}$
$BW_{s.t.} = \{BW^1, \ldots, BW^{31}\}$
$SF_{s.t.} = \{SF^1, \ldots, SF^{30}\}$
$MFCC_{s.t.}(i) = \{MFCC^1(i), \ldots, MFCC^{124}(i)\}$, i=1, …, 12
$F^{freq.}_{s.t.}$
Statistics over all windows
$\{min, max, mean, var\}(SC_{s.t.}, BW_{s.t.} SF_{s.t.})$
$\{mean\}(MFCC_{s.t.}(i))$, i=1,…,12
$F^{freq.}_{l.t.}$
Model Training

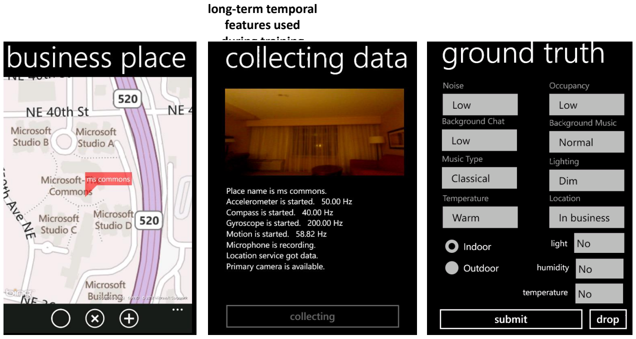
Figure 10: Detailed view of feature extraction for each 1 second segment. First, short-term features ($F^{temp.}_{\text{s.t.}}$,$F^{freq.}_{\text{s.t.}}$) are computed over different windowing schemes based on the feature type. Then, long-term features ($F^{temp.}_{\text{l.t.}}$,$F^{freq.}_{\text{l.t.}}$) are computed as statistics over all the short-term feature values. Only the long-term features are used for model training.

device, it can be easily and timely carried out by modern smartphones. We have implemented the whole feature extraction pipeline as a managed application in Windows Phone 7.5 running on a Nokia Lumia 900 device. Feature extraction for each 1-second segment takes approximately 1 second to complete. However, by pipelining microphone sampling and feature computation, features for the immediately previous 1 second segment can be extracted while the next 1 second segment is being sampled. Thus, independently of the length of the audio trace, feature extraction is completed one second after the microphone sampling is completed. Given that the model on the phone runs in milliseconds, the raw audio data can be mapped to high level ambience metadata in real-time.

Note that audio is being recorded only while the mobile device is in user hands, and while the user is attempting to check-in to a given business. In that way we ensure that the sensors on the phone are not obstructed, and that the power overhead due to the microphone sensing is minimized. Because the user is already using the phone, all major power drawing components (i.e., processor, screen, radio) are already powered up. As a result, the power overhead of sensing, and feature computation becomes negligible.

5. EVALUATION

The evaluation of the proposed approach is based on audio traces recorded in real businesses. To streamline the collection of human labeled audio traces from real businesses we developed BusinessProfiler, a Windows Phone 7.5 application that records all the available sensors on the mobile phone during a check-in event (Figure 11). BusinessProfiler exposes a map interface through which users can explore nearby businesses, and select one to check-in. As soon as a business is selected by the user, BusinessProfiler starts to automatically record data from all the available sensors on the mobile device (camera, microphone, accelerometer, magnetometer, gyro, and location) for 15 consecutive seconds. In this work, we only leverage the audio data recorded through the microphone at the sampling rate of 16KHz. When data collection on the mobile device is completed, BusinessProfiler exposes a data input screen where the ground truth labels about the occupancy, background human chatter, music, and noise levels are provided by the user, according to the information shown in Table 1.



Place name is ms commons.
Accelerometer is started. 50.00 Hz
Compass is started. 40.00 Hz
Gyroscope is started. 200.00 Hz
Motion is started. 58.82 Hz
Microphone is recording.
Location service got data.
Primary camera is available.

Noise: Low
Occupancy: Low
Background Chat: Low
Background Music: Normal
Music Type: Classical
Lighting: Dim
Temperature: Warm
Location: In business
● Indoor
○ Outdoor
light: No
humidity: No
temperature: No

Figure 11: The BusinessProfiler data collection application running on Windows Phone 7.5.

Model Type	Classification Label			
	Low		High	
	Low	Normal	High	Very High
Occupancy	36	56	37	21
Chatter	29	53	47	21
Music	65	40	24	21
Noise	34	76	37	3

Table 2: Distribution of traces across the different classification labels. For 2-level classification, Normal and Low labels are unified to a single Low label, and High and Very High labels are unified to a single High label.

Multiple people used BusinessProfiler on Nokia Lumia 900 devices to record audio traces of real businesses over a period of three months in the US. The audio traces were recorded at the table or bar area where the person was seated, or even at the entrance of the business as the person was waiting for a table. During this time, 150 labeled audio traces were recorded spanning more than 50 unique businesses of multiple types including restaurants, bars, coffee and ice-cream shops. For some businesses multiple traces were recorded, but all of them were recorded at different dates, across different locations within the business, and with different conditions (i.e., outdoor vs. indoor area, high vs. low occupancy etc.). The set of profiled businesses included a wide range of business layouts varying from tiny coffee shops to large establishments with multiple rooms, as well as indoor and outdoor seating areas. Table 2 shows the distribution of the collected audio traces across the different labels for all the different types of models.

This dataset forms the basis for our model evaluation. For each of the four business metadata types we apply 3-fold validation on the labeled audio traces to train a *single* model across all businesses, and evaluate the model's accuracy. We initially train models assuming the 4-level labeling scheme (Table 1), and then re-train the same models assuming a 2-level labeling scheme where Low and Normal labels are combined to a single Low label, and High and Very High labels are combined to a single High label. In both cases, we perform feature sensitivity analysis by exhaustively training more than 4000 models to cover all possible combinations of features for all four business metadata models. Unless otherwise noted, all results were acquired according to the setup shown in Figure 4, where the total duration T of the audio trace is 15 seconds.

The modeling infrastructure used in this work is based on the WEKA toolkit [32]. WEKA provides a platform for training and evaluating a vast collection of state-of-the-art machine learning algorithms. In this work, we use $J48$ decision trees as our modeling approach, which is an implementation of the widely used $C4.5$ decision trees [25].

		Class of Features Leveraged				
		Energy	ZCR	Spectrogram	Naive	**All**
4-Level	Occupancy	52.0	40.7	42.7	37.3	**75.3%**
	Chatter	56.7	44.0	56.7	35.3	**72.0%**
	Music	55.3	42.7	58.7	43.3	**78.0%**
	Noise	52.7	53.3	60.7	50.7	**72.0%**
2-level	Occupancy	62.0	68.0	67.3	61.3	**81.3%**
	Chatter	72.0	67.3	72.7	54.7	**81.3%**
	Music	84.7	70.0	74.7	70.0	**88.7%**
	Noise	73.3	74.7	74.7	73.3	**80.0%**

Table 3: Recognition accuracy when different classes of features are leveraged across all types of models. The "Naive" column shows the accuracy when the largest class for each model (Table 2) is always predicted.

5.1 Model Recognition Accuracy

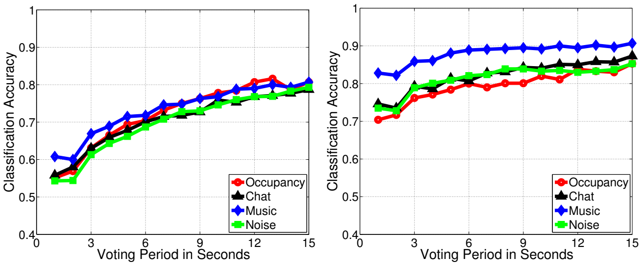
Table 3 shows the recognition accuracy achieved by the four models when 4-level and 2-level classification is used. Results are shown when all, as well as subsets of features are used (e.g., only features related to energy, ZCR, etc.). In the case of 4-level classification, the occupancy of the business can be predicted with 75% accuracy, while background chatter, mucic, and noise levels can be predicted with 72%, 78%, and 72% accuracy respectively. Note that when individual classes of features are used, the recognition accuracy is significantly lower. Individual temporal or frequency-based features cannot accurately capture the underlying structure of the recorded audio signals, resulting into low recognition rates. However, their combination increases model recognition accuracy anywhere from 4% to 35% depending on the class of feature and the model type, demonstrating the complementary nature of temporal and frequency features. Furthermore, when compared to the naive approach of always predicting the largest class for each model, the accuracy increases anywhere between 21% and 38%.

In the case of 2-level classification, the recognition rates are significantly higher. As Table 3 shows, all models achieve higher than 80% accuracy when all features are leveraged. This increase in accuracy is expected when the granularity of the labels is reduced from four to two. In general, the higher the granularity of the labels, the harder it is to maintain high accuracy.

5.1.1 Feature Sensitivity Analysis

To better understand the importance of the different features in the modeling process, we exhaustively trained models with all possible combinations of features. Table 4 shows the feature combinations that produce the highest recognition accuracy for each model type, and for both 4-level and 2-level classification. With proper feature selection, recognition rates increase to approximately 80%, and 85% in the case of 4-level and 2-level classification respectively. More importantly, the best feature set for every model includes at least one feature from every class of features (Energy, ZCR, spectrogram, and MFCC), highlighting the complementary nature of the extracted features.

The fact that subsets of features in Table 4 achieve consistently higher recognition accuracy compared to when all features are used, might be counter-intuitive at a first glance. In general, decision trees are able to train over a large number features while maintaining the best possible accuracy. However, in practice, this depends on the volume of available training data. As the number of features increases, more training data is required to efficiently train the decision tree model. With 150 traces, during our 3-fold validation only 100 traces are used for training. We believe that this amount of training data might not be enough to shape the most accurate decision tree model during training, and indicates that the recognition accuracies shown in Table 3 can be further improved.



(a) 4-level

(b) 2-level

Figure 12: Model recognition accuracy for different voting periods (duration T of the audio trace in Figure 4).

5.2 Audio Trace Duration

So far, all of the results were acquired by assuming a 15 second audio recording. However, the time the user interacts with the mobile device during the check-in event might vary, reducing the total duration of the audio recording, and thus the total number of 1 second segments available in the majority voting step. To study the impact of the initial audio recording's duration on the recognition accuracy, we re-run the majority voting step for each model type assuming different audio recording durations. Figure 12 shows the model recognition accuracy as a function of the initial audio recording's duration. In the case of 4-level classification, 6 and 14 seconds of audio recordings are required to achieve accuracy higher than 70% and 80% respectively. When 2-level classification is applied, 6 seconds of microphone samples are sufficient to achieve higher than 80% accuracy across all model types. In general, the time it takes a mobile user to launch the check-in application, interact with the UI, download the list of nearby businesses, and eventually upload the right business, will always be longer than 6 seconds, and many times longer than 10 seconds, allowing for enough time to sample the microphone, and ensure high recognition rates.

5.3 Audio Segmentation Strategies

So far, the initial 15-second audio recording is split to 1-second segments, and 1-second models are trained (Figure 4). In this section, we experiment with different segmentation strategies. In particular, we experiment with leveraging the whole audio recording of 15 seconds as a single segment, as well as with using 3-second segments. The feature extraction process is identical with the one described in Section 4. The only difference is that in the case of 15-second segments, we train 15-second models, instead of 1-second models, and there is no voting step as there is only one segment. In the case of 3-second segments, 3-second models are trained and voting is applied over all five 3-second segments.

Table 5 shows the recognition accuracy achieved from the different segmentation strategies when all features are leveraged. Verifying our initial intuition (Section 3), 1-second segments with voting achieves the highest accuracy across all models. In contrast, the 15-second segments approach achieves the lowest accuracy, and in the case of occupancy and background human chatter models, the accuracy is prohibitively low.

5.4 Near Phone talking

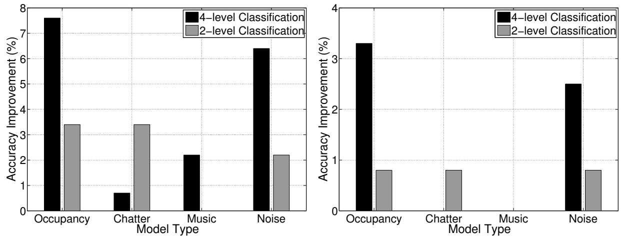
To evaluate the performance of the near phone talking model, we apply exhaustive 3-fold validation on all the recorded audio traces. When all features are leveraged, higher than 95% recognition accuracy is achieved. The accuracy increases to 96.7% when only energy, ZCR, SF, BW, and MFCC features are leveraged.

		All Features Accuracy	Best Feature Set Accuracy	Best Feature Set
4-Level	Occupancy	75.3%	80.7%	{ Energy, ZCR_{var}, SC, BW, MFCC }
	Chatter	72.0%	78.7%	{ Energy, E_{var}, ZCR_{var}, SF, MFCC }
	Music	78.0%	80.0%	{ Energy, ZCR, ZCR_{var}, SF, BW, MFCC }
	Noise	72.0%	79.0%	{ Energy, ZCR_{var}, SF, SC, BW, MFCC }
2-Level	Occupancy	81.3%	85.3%	{ Energy, ZCR, ZCR_{var}, SF, SC, BW, MFCC }
	Chatter	81.3%	87.3%	{ Energy, E_{var}, SF, SC, BW, MFCC }
	Music	88.7%	90.7%	{ Energy, E_{var}, ZCR, SF, MFCC }
	Noise	80.0%	85.3%	{ Energy, E_{var}, ZCR_{var}, SC, MFCC }

Table 4: Combination of features achieving the highest possible accuracy in our exhaustive model training evaluation.

		Audio Segment Size		
		15-second	3-second	1-second
4-Level	Occupancy	37.3	58.7	**75.3%**
	Chatter	36.0	63.3	**72.0%**
	Music	47.3	67.3	**78.0%**
	Noise	47.3	70.7	**72.0%**
2-Level	Occupancy	60.7	77.3	**81.3%**
	Chatter	72.0	77.3	**81.3%**
	Music	74.7	86.0	**88.7%**
	Noise	66.7	75.3	**80.0%**

Table 5: Recognition accuracy when different segmentation strategies of the initial audio trace are used.



(a) (b)

Figure 13: Model recognition accuracy improvement when: (a) all traces indicated as near phone talking are removed. (b) near phone talking is added as a feature.

Accurate near phone talking detection allows us to further optimize the performance of the rest of the models. Figure 13(a) shows the model recognition accuracy improvement when all audio traces indicated as near phone talking by the model are excluded from the evaluation. Consistently across all models, and for both 4-level and 2-level classification, the exclusion of near phone talking audio segments increases accuracy anywhere from 0.7% to 7.6%.

Figure 13(b) shows the model recognition accuracy when the output of the near phone talking model is used as an input binary feature in all other models. The near phone talking feature improves the recognition accuracy up to 3% with the exception of music, where the accuracy remains the same. Occupancy and noise models seem to exhibit the highest accuracy improvement when near phone talking is encoded as a feature.

5.5 Cross-device Performance

So far, all of the 150 audio traces were recorded using Nokia Lumia 900 devices. However, in practice, audio recordings from multiple devices will be generated. Differences in the hardware and the operating system of the device can lead to differences in the recorded audio signals, jeopardizing the models' accuracy.

To evaluate the sensitivity of the modeling approach to hardware variations, we collected 20 additional audio recordings from multiple businesses. During each of these recordings, 3 different devices (Nokia Lumia 900 with Windows Phone 7.5, iPad2 with iOS, and Samsung Nexus S with Android) were simultaneously recording their microphones at the same sampling rate for 15 seconds.

	No Calibration - **Calibration**		
	Nokia Lumia 900	iPad2	Nexus S
Occupancy	75% - **75%**	80% - **90%**	80% - **80%**
Chatter	90% - **90%**	65% - **85%**	70% - **85%**
Music	100% - **100%**	90% - **100%**	100% - **100%**
Noise	80% - **80%**	75% - **75%**	75% - **75%**
N.P.T.	100% - **100%**	100% - **100%**	100% - **100%**

Table 6: Model recognition accuracy across devices with and without calibration of the raw audio signal.

Figure 14(a) and Figure 15(a) show the raw audio signal along with the computed ZCR and Energy features for an example business audio recording across all three devices. At a higher level, even though similar, the raw audio signal recorded on iPad2 and Nexus S devices exhibits differences with respect to the audio signal recorded on Nokia Lumia 900. These differences can be better seen when comparing the ZCR and energy features across the 3 devices. Overall, iPad2 and Nexus S consistently produce higher energy and ZCR values compared to Nokia Lumia 900.

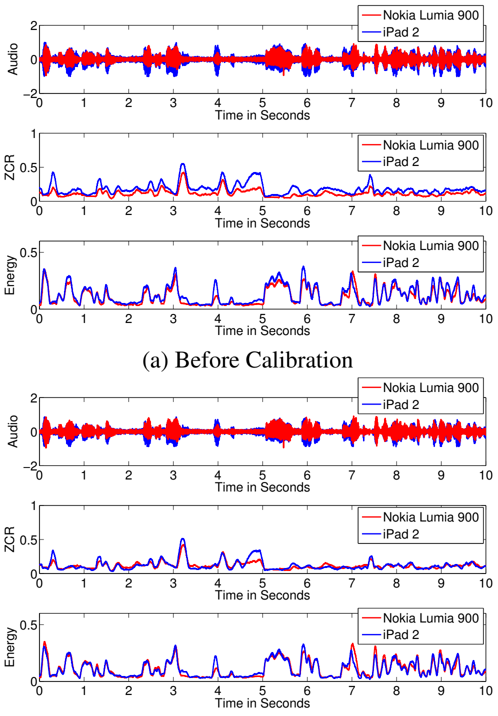
These differences, however, tend to be consistent over time. By closely examining the raw audio signals recorded on all three devices, it became apparent that these constant differences could be easily offset using small scaling parameters. In particular, the raw audio signal amplitude/energy on the iPad2 can perfectly match the one recorded on Nokia Lumia 900, by simply multiplying the audio signal by a constant factor. Similarly, by properly adjusting the audio zero level (shifting logical zero into a value higher than absolute zero), similar ZCR values are recorded across all three devices.

Figure 14(b) and Figure 15(b) show the raw audio signals along with the computed ZCR and Energy features after scaling the amplitude of the audio signal, and properly adjusting the zero level in the iPad2 and Nexus S devices. Both, raw audio and feature values now exhibit very small differences.

To evaluate the impact of these device variations, we apply the models trained in the previous section, where only data from Nokia Lumia 900 was used, on the 20 additional traces recorded on Nokia, iPad2, and Nexus S devices. Table 6 shows the model recognition accuracy with and without calibration for all devices. Even when no calibration is applied, the recognition accuracy for both iPad2 and Nexus S devices is high. Only background chatter recognition is significantly lower for both iPad2 and Nexus S. However, by calibrating their raw audio signals as described above, the background chatter recognition accuracy for these devices increases by 20%, to become as high as 85%. Accuracy can be further increased if labeled data from multiple platforms is used during model training.

6. RELATED WORK

The area of automated business metadata extraction is new, and highly unexplored. To the best of our knowledge, the startup com-



(a) Before Calibration

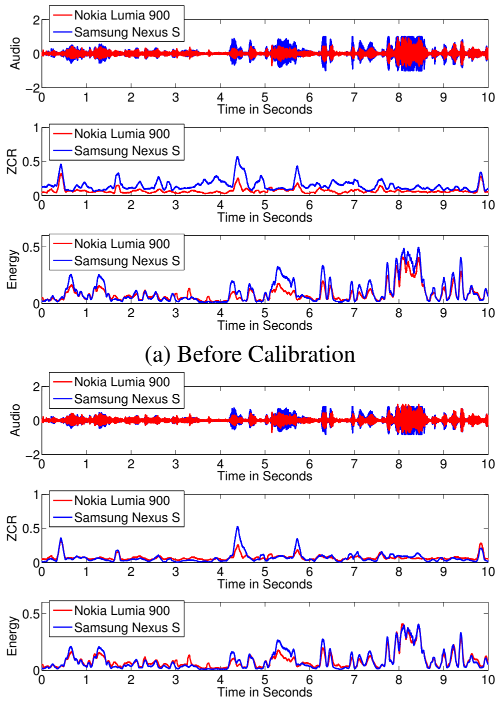
(b) After Calibration

Figure 14: Example raw audio and feature extraction before and after calibration for an iPad2.

pany SceneTap [26] provides the only alternative approach to the problem. SceneTap deploys custom cameras in businesses, that can accurately count the number of people entering/exiting a business. Even though such an approach can produce very accurate occupancy information, it doesn't scale. Instrumenting every business with custom hardware is not a scalable solution, and in many cases, employing camera sensors for this purpose can be privacy intrusive for business customers. In contrast, our approach uses audio data recorded on real users' mobile devices at the time of a business check-in. Since we don't require any hardware instrumentation of businesses, the proposed approach can easily scale to cover any business real users actually visit.

Recently, Liu et al. [13] proposed a geo-sensitive question answering architecture that enables users to submit location-related questions, and have random users, familiar with these locations, answer them through social media. Even though such a system could be used to obtain similar ambience metadata about businesses, it has two major drawbacks. First, question answering is not real-time, in the sense that users asking questions might have to wait for an arbitrary amount of time to get a response. Second, there are not always users that are willing to actively provide this information. Conversely to this approach, our work proposes a system architecture to automatically collect such rich business metadata in a passive, unobtrusive way that takes mobile users out of the loop.

Previous work has already leveraged audio recordings in businesses, but in a different context, and with simpler recognition tasks at hand. Chon et al. used audio recordings to classify the type of the business [4] (i.e., mall vs coffee shop), and Azizyan et al. to differentiate nearby businesses for localization purposes [3]. Even though both of these works leverage similar audio processing, our work differs in two fundamental ways. First, in contrast to previous work, we demonstrate the feasibility of identifying fine-grained information about the ambience of the business, that requires to individually model the different acoustic sources (human chatter, music, noise) in the audio recording. This is particularly challenging because we are not trying to recognize specific repeatable sound signatures (i.e., Azizyan et al. [3]) as human chatter, music, and occupancy varies continuously within a business and across businesses. Second, we explicitly address the challenge of people talking right next to the phone, and show that we can reliably detect the ambience of a business even in the presence of near phone talking.



(a) Before Calibration

(b) After Calibration

Figure 15: Example raw audio and feature extraction before and after calibration for a Nexus S.

At a higher level, Micro-blog [6], a platform where users can upload location annotated multimedia blogs (e.g., videos, audio, pictures), could be seen as similar to the work described here. Microblog encourages users to upload multimedia blogs, and allows them to browse these blogs on a map. Instead of making available location-annotated data, we design the necessary intelligence for automatically transforming audio data to useful information for users and search engines to consume.

Even though mobile audio data have not been used before for business metadata extraction, they have been exploited to enable various tasks on mobile devices ranging from understanding life events, and automatic diary creation [15], to indoor localization [29], context minin [12, 19], and speaker identification [14]. Several of the low-level audio signal processing techniques leveraged in these works have inspired parts of our feature extraction pipeline.

7. CONCLUSIONS

We have presented a robust and scalable approach for extracting local business ambience metadata that has the potential to transform local search. By carefully analyzing audio signals recorded on the phones of real users as they check-in to businesses, information about the occupancy, human chatter, music, and noise levels in a business can be inferred with higher than 79% accuracy. This type of metadata can enable a new local search user experience. Users can get a peek into the ambience of any business at the time of the query, and use it to navigate the physical world of local businesses in the same way they use traffic data to navigate the road network.

8. REFERENCES

[1] M. Aly. Survey on multiclass classification methods. Technical report, California Institude of Technology, 2005.

[2] F. Asano, M. Goto, K. Itou, and H. Asoh. Real-time sound source localization and separation system and its application to automatic speech recognition. In *EuroSpeech 2001*, pages 1013–1016.

[3] M. Azizyan, I. Constandache, and R. Roy Choudhury. Surroundsense: Mobile phone localization via ambience fingerprinting. In *ACM MobiCom'09*, 2009.

[4] Y. Chon, N. D. Lane, F. Li, H. Cha, and F. Zhao. Automatically characterizing places with opportunistic crowdsensing using smartphones. In *ACM UbiComp'12*, 2012.

[5] M. Gamon, A. Aue, S. Corston-Oliver, and E. Ringger. Pulse: Mining customer opinions from free text. In *IDA 05*, 2005.

[6] S. Gaonkar, J. Li, R. Roy Choudhury, L. Cox, and A. Schmidt. Micro-blog: Sharing and querying content through mobile phones and social participation. In *ACM MobiSys'08*, 2008.

[7] M. R. Hasan, M. Jamil, and M. G. R. M. S. Rahman. Speaker identification using mel frequency cepstral coefficients. In *ICECE 2004*, 2004.

[8] H. G. Hirsch and D. Pearce. The aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions. In *ISCA ITRW ASR2000*, 2000.

[9] M. Hu and B. Liu. Mining and summarizing customer reviews. In *KDD 04*, 2004.

[10] M. Hu and B. Liu. Mining opinion features in customer reviews. In *American Association for Artificial Intelligence*, 2004.

[11] T. Kinnunen, E. Karpov, and P. Franti. Real-time speaker identification and verification. *IEEE Trans. Audio Speech, and Landguage Process*, 14(1):277–288, 2006.

[12] P. Korpipaa, J. Mantyjarvi, J. Kela, H. Keranen, and E. J. Malm. Managing context information in mobile devices. *IEEE Pervasive Computing*, 2:42 – 51, 2003.

[13] Y. Liu, T. Alexandrova, and T. Nakajima. Using stranger as sensors: Temporal and geo-sensitive question answering via social media. In *WWW 13*, 2013.

[14] H. Lu, A. J. B. Brush, B. Priyantha, A. K. Karlson, and J. Liu. Speakersense: energy efficient unobtrusive speaker identification on mobile phones. In *Proceedings of the 9th international conference on Pervasive computing*, Pervasive'11, pages 188–205, 2011.

[15] H. Lu, W. Pan, N. D. Lane, T. Choudhury, and A. T. Campbell. Soundsense: Scalable sound sensing for people-centric applications on mobile phones. In *ACM MobiSys'09*, 2009.

[16] J. McAuley and J. Leskovec. From amateurs to connoisseurs: Modeling the evolution of user expertise through online reviews. In *WWW 13*, 2013.

[17] X. Meng and H. Wang. Mining user reviews: from speciïňAcation to summarization. In *ACL-IJCNLP 09*, 2009.

[18] P. Mermelstein. Distance measures for speech recognition, psychological and instrumental. In R. Chen, editor, *Pattern Recognition and Artificial Intelligence*, pages 374–388. Academic Press, New York, 1976.

[19] E. Miluzzo, C. T. Cornelius, A. Ramaswamy, T. Choudhury, Z. Liu, and A. T. Campbell. Darwin phones: The evolution of sensing and inference on mobile phones. In *ACM MobiSys'10*, 2010.

[20] S. Moghaddam and M. Ester. The flda model for aspect-based opinion mining: Addressing the cold start problem. In *WWW 13*, 2013.

[21] K. Nakadai, H. G. Okuno, and H. Kitano. Real-time sound source localization and separation for robot audition. In *IEEE ICSLP 2002*, pages 193–196.

[22] J. Pasternack and D. Roth. Latent credibility analysis. In *WWW 13*, 2013.

[23] J. C. Platt, N. Cristianini, and J. Shawe-Taylor. Large margin dags for multiclass classification. In *NIPS 00*, 2000.

[24] A. M. Popescu and O. Etzioni. Extracting product features and opinions from reviews. In *EMNLP 05*, 2005.

[25] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[26] SceneTap. `http://scenetap.com/`.

[27] Shazam. `http://www.shazam.com/`.

[28] SoundHound. `http://www.soundhound.com/`.

[29] S. P. Tarzia, P. A. Dinda, R. P. Dick, and G. Memik. Indoor localization without infrastructure using the acoustic background spectrum. In *ACM MobiSys'11*, 2011.

[30] J.-M. Valin, F. Michaud, J. Rouat, and D. Letourneau. Robust sound source localization using a microphone array on a mobile roboot. In *IEEE/RSJ IROS 2003*, 2003.

[31] O. Viikki and K. Laurila. Cepstral domain segmental feature vector normalization for noise robust speech recognition. *Speech Communication*, 25:133–147, 1998.

[32] WEKA. `http://www.cs.waikato.ac.nz/ml/weka/`.

[33] T. Wu, C. Lin, and R. C. Weng. Probability estimates for multi-class classification by pairwise coupling. In *Journal of Machine Learning Research*, 2004.

[34] P. Zhou, Y. Zheng, Z. Li, M. Li, and G. Shen. Iodetector: A generic service for indoor outdoor detection. In *ACM SenSys'12*, 2012.