Authors' Unofficial Copy of:
Hernisa Kacorri, Sergio Mascetti, Andrea Gerino, Dragan Ahmetovic, Hironobu Takagi, and Chieko Asakawa. 2016. Supporting Orientation of People with Visual Impairment: Analysis of Large Scale Usage Data. In Proceedings of the 18th Annual SIGACCESS Conference on Computers and Accessibility (ASSETS'16). Reno, NV, USA. New York: ACM Press.

Supporting Orientation of People with Visual Impairment: Analysis of Large Scale Usage Data

Hernisa Kacorri
Carnegie Mellon University
hkacorri@cmu.edu

Sergio Mascetti
Università degli Studi di Milano
and EveryWare Technologies
sergio.mascetti@unimi.it

Andrea Gerino
Università degli Studi di Milano
and EveryWare Technologies
andrea.gerino@unimi.it

Dragan Ahmetovic
Carnegie Mellon University
dragan1@cmu.edu

Hironobu Takagi
IBM
takagih@jp.ibm.com

Chieko Asakawa
Carnegie Mellon University
chiekoa@cs.cmu.edu

ABSTRACT

In the field of assistive technology, large scale user studies are hindered by the fact that potential participants are geographically sparse and longitudinal studies are often time consuming. In this contribution, we rely on remote usage data to perform large scale and long duration behavior analysis on users of *i*Move, a mobile app that supports the orientation of people with visual impairments.

Exploratory analysis highlights popular functions, common configuration settings, and usage patterns among *i*Move users. The study shows stark differences between users accessing the app through VoiceOver and other users, who tend to use the app more scarcely and sporadically. Analysis through clustering of VoiceOver *i*Move user interactions discovers four distinct user groups: 1) users interested in surrounding points of interest, 2) users keeping the app active for long sessions while in movement, 3) users interacting in short bursts to inquire about current location, and 4) users querying in bursts about surrounding points of interest and addresses.

Our analysis provides insights into *i*Move's user base and can inform decisions for tailoring the app to diverse user groups, developing future improvements of the software, or guiding the design process of similar assistive tools.

CCS Concepts

•Human-centered computing → Accessibility design and evaluation methods; •Computing methodologies → Cluster analysis; •Social and professional topics → People with disabilities;

1. INTRODUCTION

Nonvisual understanding of the environment is far more ineffective and inefficient as well as potentially dangerous than scanning the surroundings by sight [3]. In fact, orientation, a person's awareness of position and heading in the environment [10, 23], is a challenge for people with severe visual impairment and the main difficulties derive from the inability to efficiently obtain a mental map of the surrounding area while moving. To address this problem, many researchers and developers of assistive technology, surveyed in [10, 11], have explored technological approaches such as laser canes, sonar devices, and GPS navigation tools.

The design of these technological solutions is typically guided by supervised experiments with few participants, such as formative studies (e.g. [26]), Wizard-of-Oz experiments (e.g. [8]), and evaluation studies (e.g. [17]). These approaches may be attractive for the advantages they offer. Researchers can conduct experiments with prototype applications, or in some cases, even prototypes without working software. They can also conduct such experiments in controlled situations and with users whose characteristics (e.g., form of disability, age) are known in advance. However, these approaches are also limited in many ways. First, it is not possible to explore many real world scenarios. Second, these studies generally involve participants that live in close proximity to the physical location where the experiment is conducted, leading to the possibility of cultural bias. Third, these experiments are susceptible to the Hawthorne effect [1], where users may act differently when they know they are being watched. Finally, and most important, these approaches are not scalable both in terms of number of involved subjects and length of the study as stressed in [10].

We are interested in advancing state-of-the-art technologies for supporting orientation and mobility of people with visual impairment. For these applications, we want to study the following questions. Which are the most frequently used functionalities? What are the most common user interaction patterns? Can users be grouped based on their interaction patterns? How do users benefit from these applications? Being able to answer these questions makes it possible not only to improve existing applications but also to guide the design of similar applications supporting outdoor mobility (e.g., [15, 14]) as well as indoor navigation (e.g. [2] among others).

To answer these questions, we analyze large scale usage data remotely collected from *i*Move[1], a GPS-based mobile application that supports outdoor orientation of people with

[1]https://itunes.apple.com/us/app/imove/id593874954

 ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

visual impairments. The app provides information about nearby "landmarks" that help the user construct a mental map of their environment.

Our analysis is conducted both with inferential and exploratory methods using statistical tools. We also employ machine learning tools for unsupervised discovery of user clusters based on common interaction patterns. To the best of our knowledge, this is the first study adopting this methodology in the field of mobility of people with visual impairments.

Specifically this paper presents three main contributions:

- First, the analysis highlights a number of usage properties of iMove, including commonly used functions and preferences for applications settings. We examine the differences both in application use and preferred settings across screen-reader users and other users. We also discover clusters of users based on common interaction patterns and identify features that are primarily responsible for cluster formation. The proposed feature space is intuitive enough to interpret the meaning of the clusters.
- Second, we describe our collected dataset and release it for use by other researchers.
- Third, the analysis methodology proposed in this contribution may be adapted to the study of other applications in the field of assistive technologies.

2. RELATED WORK

Understanding user behavior during interactions with a software application is of paramount importance for evaluating the application's effectiveness, for guiding the iterative design process, and for informing the design of similar applications. However, there are inherent challenges in conducting behavioral studies both over long periods and with large samples of participants with disability. Thus, fewer contributions in the field of assistive technologies adopt methodologies involving analysis of collected real-world usage data and often, their participants' demographics are known a priori or collected through questionnaires. To name a few, [4] automatically collected user actions during web browsing to assess the accessibility of web pages by visually impaired users. Usage log analysis is also performed in [18] to evaluate the localization error of a navigation assistance tool using Video Light Communication (VLC) for guiding people with visual impairments. In [12], log data from real-world tasks over a long period were used to build predictive models in distinguishing users by pointing performance. Last, in [22], behavior anomalies perceived during user interaction with a sensor-enabled smart home environment act as a diagnostic tool for detecting mild cognitive impairments in senior patients.

In the broader field of human computer interaction, where the pool of participants tend to be much larger, it is more feasible for the researchers to perform behavior analysis on large scale datasets available to the research community (e.g. [7, 9, 13, 20]). These analyses often combine data-driven approaches from many fields such as classification, clustering, and time-series analysis from machine learning, sentiment analysis from natural language processing, and community detection from network analysis. The work of Wang et al., 2016 [25] is the closest to our analysis. The authors applied natural language processing techniques to detect similarity among Facebook social network users. Specifically, they analyzed "clickstreams", timed sequences of interactions with website, and performed hierarchical clustering on users' clickstream to identify common user profiles (e.g., those who like others' pages and those who update their status often).

Prior work in cognitive science related to spacial representation and navigation in people with visual impairments [24, 23] discuss limitations in user studies which compare orientation and mobility performance among sighted, early blind, and late blind participants. Their discussion on adopted and preferred navigation strategies among these users made us wonder whether similarities in these strategies also lead to similarities across user interaction with supportive orientation and navigation technologies. Motivated by this question, we investigate approaches, similar to Wang et al. [25], that automatically discover user clusters based on streams of interactions with iMove. However, the link between these clusters and underlying user-adopted navigation strategies is beyond the scope of this paper.

3. iMove APP AND DATASET

iMove is an iOS application that is accessible through VoiceOver screen reader and magnifier. The app informs users about outdoor geo-referenced information such as current address, nearby Points Of Interest (POIs), and *geo-notes* i.e., user-defined notes associated to a geographical location. Users can access this information either explicitly, e.g., ask for current address in the root screen (Fig. 1(a)) and list of nearby POIs (Fig. 1(b)), or periodically while in motion by turning on the "Notify me" toggle button in the root screen. The frequency of such periodic updates can be tuned both in terms of time and proximity (i.e., a minimum temporal/spatial distance between two readings). Geo-notes can be created and edited as audio recordings or text entries (Fig. 1(c)) and they are organized into "routes" (Fig. 1(d)).

iMove is designed to be highly customizable: users can specify the categories of POIs they are interested in, activate automatic readings of surrounding information, and modify settings related to system verbosity. Therefore, beyond user visited screens, actions, and received notification, we also collect data related to their settings modifications.

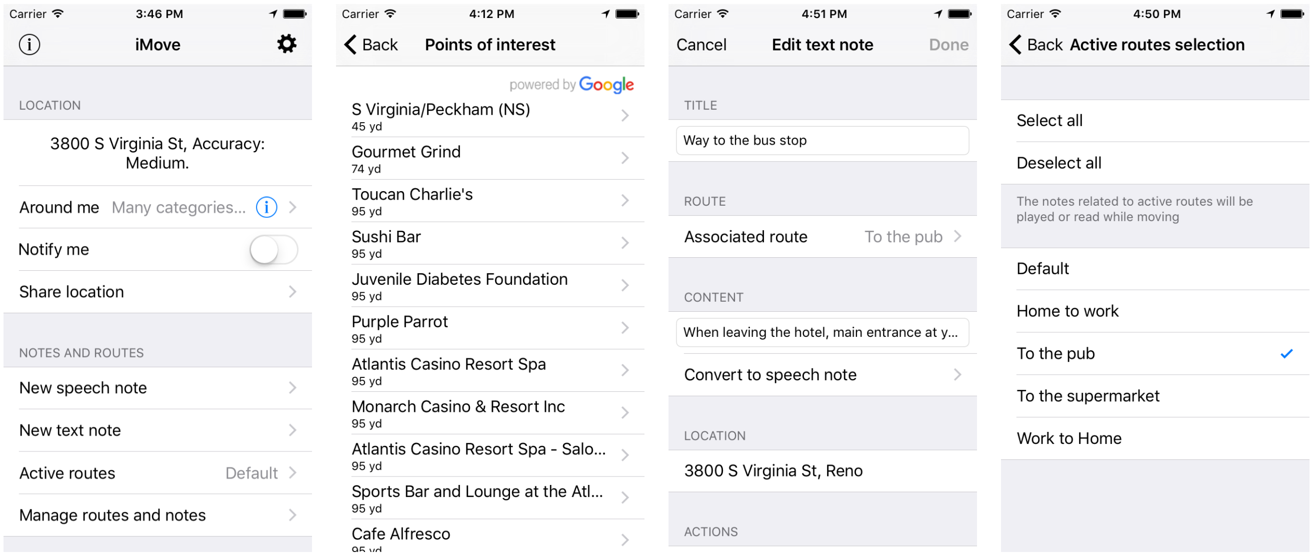
3.1 Remote Logging System

Since iMove version 2.0, released on December 8, 2015, the application implements a remote logging system that makes it possible to collect anonymous app usage information. Logging is supported by a client library within iMove communicating with a REST server and a non-relational database back-end.

A detailed description of the released dataset is available online[2]. Data was collected in compliance with European regulations[3] and user logs were recorded in anonymized form. Thus, the dataset does not include location-related information, e.g. POI, or user-generated content, e.g. geo-

[2]http://webmind.di.unimi.it/assetsim16/

[3]Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data, OJ L 281, 23.11.1995, 31-50.



(a) Root screen. (b) POI screen. (c) Edit text-note screen. (d) Route selection screen.

Figure 1: Main screens of the iMove application.

notes. To reconstruct user-interaction history, each log includes a unique pseudo-identifier associated with an anonymized user.

Each log record has two main components. The first component contains data about the user and the device on which *i*Move is running: the user's pseudo-identifier, the device model, the system language, whether VoiceOver is enabled or not, the application version (we collected data for build versions 31 and 32) and log creation timestamps in the user's time zone, UTC, and the server time.

The second component contains the application usage data. In *i*Move, we partition log entries into four different categories of usage data:

Screen logs capture user navigation between *i*Move screens. Each screen log records the screen name and an "enter" or "exit" label when a user enters or exits a screen.

Action logs record *i*Move function activation by a user such as recording a new speech note.

Notification logs are generated when the application automatically provides information to the user (e.g. when the user gets close to a POI).

Preference logs are generated every time a user changes *i*Move settings. A preference log lists the name of the parameter, its old value, and its new value.

3.2 *i*Move Dataset Overview

The *i*Move dataset was collected during the December 2015 - April 2016 period and contains a total of $771,975$ log records across $17,624$ unique user pseudo-identifiers ($\mu = 43.8, \sigma = 105.15$) log records per user with range 1 - 7,299. From the feedback we received by email and on the appstore, we realized that a number of users, who we call "incidental" users, installed the application without realizing its functionality and its intended use for people with visual impairments. For example, some users confused *i*Move with *iMovie*, a popular application for video editing.

To filter out these users, we introduce the concept of "interaction session" (or simply, session): a period of time during which a user frequently interacts with the application (e.g., navigates in the screens, performs actions or receives system notifications). A session is extracted from app usage data as a sequence of consecutive log entries such that: i) the sequence begins with a "screenRootEnter" record, which signals that the user opened the main screen of the application, and ii) there is at least a 5 minutes gap between the session starting log and the previous log. This constraint captures the intuition that the user might temporarily exit the app for a short time within an interaction session.

Based on the intuition that users who are uninterested in *i*Move would not use it for more than one session, we consider only users having two or more sessions. There are a total of $4,055$ such users generating a total of $255,004$ logs ($\mu = 62.89, \sigma = 211.51$ logs/user with range 2-7,296).

4. ANALYSIS

4.1 *i*Move Use Properties Across All Users

We analyze log records from all $4,055$ users with the goal of highlighting *i*Move use properties such as commonly used functions and user preferred values for interaction parameters. Using both inferential and exploratory methods we examine four categories of log records: preferences, screen activity, actions, and notifications.

One interesting aspect of *i*Move is the support of user-defined geo-notes, where users can either record a speech note associated with a location or type it as text. While both options are available, we expect that the former will be the one adopted by the users since the purpose of the app is to support mobility and it is has been observed that typing in mobility is particularly challenging for people with visual impairments [16]. Specifically, we formulate and examine the following hypothesis:

H1: *i*Move users will favor speech over text for input modality when creating geo-notes.

Results and Interpretation

Preference logs account for 3.41% of the total log records. Figure 2 reports, for each preference setting, its default value and how many times it has been set to a given value. We observe that the parameter "keepUserInformed", which toggles all notifications, was changed far more frequently. This interaction was expected by our intuition that users will frequently toggle off when they do not want to be disturbed by notifications. Anticipating such an interaction during the design of *i*Move, we position the toggle button in the root screen (see Figure 1(a)). Indeed, 22.2% of the users changed this value twice or more, while 20.9% of the users changed it more than once for at least one session.

We also explore log records for other parameters, whose semantics are detailed online[4], to assess the default values provided by *i*Move. This analysis cannot take into account only the values changed by the users. Since all logged changes necessarily involve modification of default values, the logged data does not inform us of how many users intentionally choose to stick with the default value for a given parameter. To estimate this, we compute, for each parameter, the percentage of users that changed the parameter value at least once, among the users that actually visited that parameter's settings screen (values are reported in Figure 2).

For example, only 4% of the users who entered the "Settings_location" screen actually changed the value of the "locationSpatialThreshold" parameter. On the other hand, 22% of the users who entered the System settings page changed the "prevent screen lock" option that by default is set to false. Similarly, 23% of the users changed the preference "sayCity" and more than 16% of the users changed the "saySpeed", "sayHeading" and "sayCourse". These are parameters whose default values are candidates for change in future versions of the app. More generally, we observe the four parameters above are all related to the type of information provided to the user when a location notification occurs. To avoid verbosity in the application, we limited location notifications to the name and number of the street by default. Apparently, many users prefer to have more detailed information.

Screen, Notification, and Action logs account for 66.23%, 29.55%, and 0.76% of the total 255,004 log records, respectively. Figure 3 illustrates the distribution of these records across the subsequent categories. We observe that "Location" is the most common notification followed by "POI" and the two geo-notes. Interestingly, the "NavigateToPOI" function, suggested by many users and introduced with app build 31, is the most frequent user action. Geo-notes notifications ("SpeechNote" and "TextNote") are less frequent than "Location" and "POI" notifications, accounting for 3% of the total notifications. This is due to the fact that 83% of the users never created a geo-note. Among users creating a geo-note, the percentage of geo-note notifications is 10% of the total notifications.

Figure 4 shows the distributions of per-user screen, action, and notification logs related to speech and text geo-notes (box indicates quartiles, center-line indicates median, square symbol indicates mean, whiskers indicate 1.5 inter-quartile ranges, and crosses indicate outliers). In *support of hypothesis H1*, there is a significant difference between the pairs of these graphs determined by Mann-Whitney U test. Specifically, users visit the "NewSpeechNote" screen significantly more times than the "NewTextNote" screen ($p < 0.001$) and perform significantly more "SavedNewSpeechNote" actions than "SavedNewTextNote" actions ($p < 0.05$). Not surprisingly, users receive significanlty more "SpeechNote" notifications than "TextNote" notifications ($p < 0.05$).

4.2 Voiceover-Based User Comparison

As mentioned in Section 3.1 for each log record we collect the VoiceOver field, which reports whether VoiceOver was active when the record was generated. This field is particularly relevant for our analysis as it allows us to distinguish users that are likely to have severe visual impairments. Therefore, we partitioned the *i*Move users into two groups: VO-group users (VO-users) have at least one VoiceOver-active record and NVO-group users (NVO-users), have no VoiceOver-active records.

We formulate and examine the following hypotheses:

H2: VO-users will have different settings preferences than NVO-users.

H3: VO-users will make more intense use of *i*Move as measured by the number of actions and notifications as well as the span of days using the app.

Results and Interpretation

VO-group consists of 1,025 users whereas NVO-group includes the rest 3,030 users. We observe that while VO-group includes a smaller percentage of the overall *i*Move users (25.28%), the number of records generated by this group accounts for more than half of the logs (56.34%) along with a higher mean records per user ($\mu = 140.16, \sigma = 403.91$) than the NVO-group ($\mu = 36.74, \sigma = 45.05$). We also observe a small positive correlation in our dataset between the number of records for a user and the percentage of records with activated VoiceOver for the same user.

Users in VO-group generated logs with a high mean percentage of active-VoiceOver records ($1\%-100\%, \mu = 95.26\%$, $\sigma = 16.6\%$). This suggests that, while by definition a user in VO-group can only have one record with VoiceOver-active, in practice users in VO-group have VoiceOver activated almost all the time during use of *i*Move. We suspect that users in VO-group are mostly people with severe visual impairments and a few users with low vision that sporadically activate VoiceOver while users in NVO-group either use magnifier in their interaction with the app or are non visually impaired ("incidental" users, see Section 3.2).

Figure 5 illustrates side-by-side the distribution of threshold preference from both groups. In *support of hypothesis H2*, we find that users in VO-group set smaller temporal and spatial threshold values determined by Mann-Whitney U test (p<0.05). Even though different threshold parameters have different semantics, smaller temporal values result in more frequent notifications, while smaller spatial values for "PoiProximity" and "GeoNoteProximity" indicate preference for notification only in close proximity to the target place (POI or geo-note). These findings suggest that users in VO-group prefer to receive information more frequently than users in NVO-group and only in close proximity to the target.

To examine hypothesis H3, we consider the number of notifications and actions, as well as the period of *i*Move use

[4] *i*Move parameter semantics is detailed in http://webmind.di.unimi.it/assetsim16/#param_semantics.

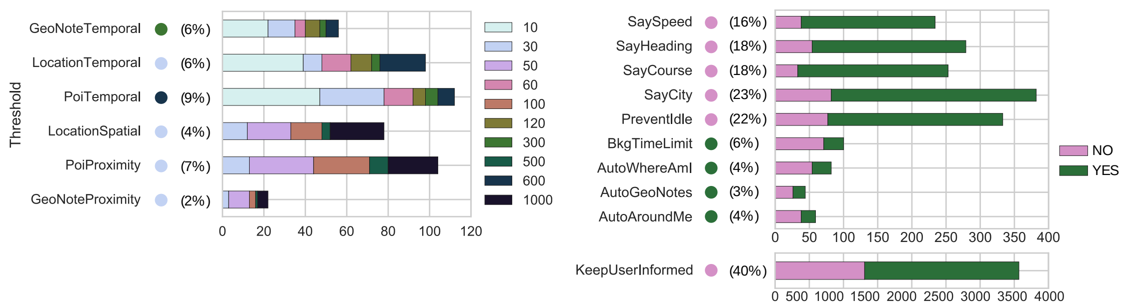


Figure 2: Number of preference records generated from the subset of users that modified the default values. Colored circles indicate preferences' default values, while the percentages represent how many users changed the value at least once, among those who visited the corresponding settings screen.

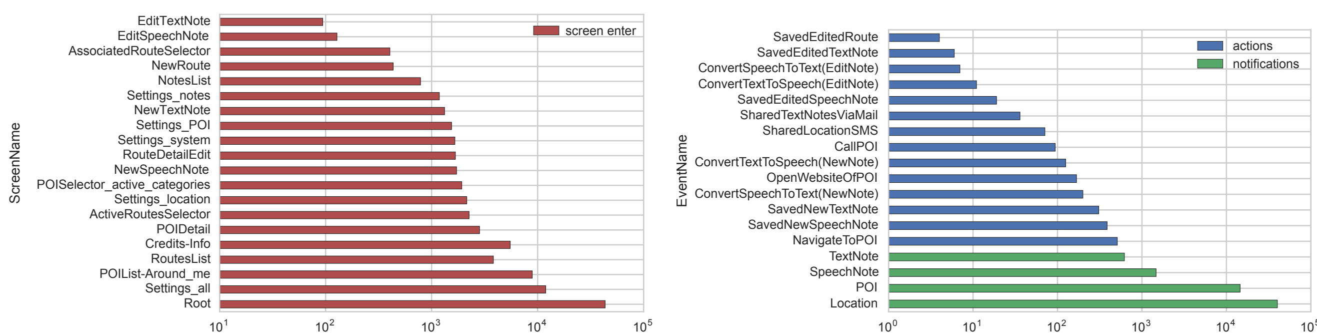


Figure 3: User log records distributed across the screens, actions, and notifications logs.

per user in each group and compare their mean ranks with Mann-Whitney U test. In *support of hypothesis H3*, we find that users in VO-group receive significantly more notifications ($p < 0.001$) such as the "Location" notifications shown in Figure 6(a)). Similarly, users in VO-group perform significantly more actions ($p < 0.001$), for example Figure 6(b) shows how the number of times a VO-user asks for directions to navigate to a POI is significantly higher than for a NVO-user. Users in VO-group use the application for a significantly longer period than the NVO-users ($p < 0.0001$), where the period of use is measured as the span of days between the first and last time a user enters the iMove root screen. On average, this duration is of 53.95 days for users in VO-group and of 20.45 days for users in NVO-group (as shown in Figure 6(c)).

4.3 User Clustering Based on $i$streams

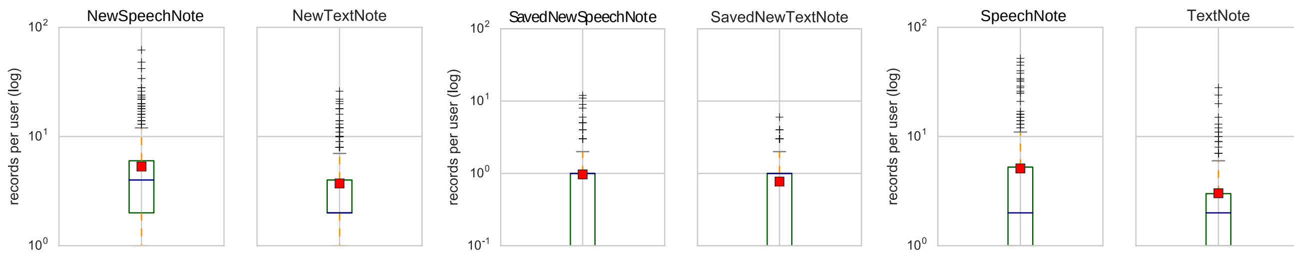
While the exploratory and inferential analyses in the previous sections reveal interesting patterns, they do not take into account the sequential relationship between the log entries. In order to learn richer patterns of interaction, we use unsupervised learning techniques on record streams, which preserve the temporal structure of the data. We anticipate that users naturally fall into clusters based on common interaction patterns with iMove. The automatic discovery of these clusters can help us identify: what are the major interaction categories; which is the most prevalent interaction; and what is the relationship between different types of interactions. This clustering is performed on the $1,025$ users residing in VO-group, who are likely to have severe visual impairments and, as shown above, make intensive use of the application.

Clustering Methodology

As discussed in related work, HCI researchers have adopted prior work in machine learning, natural language processing and network analysis, to better understand user behavior, with the social network analysis in [25] being the closest to our work. Our methodology builds upon previous methods to understand and support assistive orientation of people with visual impairment. One of the inherent challenges in analyzing our data is that users can interact with the app either by actively navigating the screens and using their functions, captured by screen and action logs, or by physically changing their location thus generating notifications logs. We introduce the notion of sessions (defined in Section 3.2) into our feature engineering (described below) to yield more intuitive and high level descriptions for the discovered clusters.

Specifically, we represent each user by the stream of interactions ($i$stream) with the app. We map users to a feature space extracted from these streams, construct a similarity graph by comparing users in this feature space, and identify clusters of similar users by graph partitioning. Finally, we interpret the meaning of the clusters by isolating primary features that are responsible for forming the clusters. To assist future researchers in adopting this methodology for analysis of their data, we describe the above steps, implementation, assumptions, and the hyper-parameters used in our clustering.

Obtaining user $i$stream. We define an $i$stream as a sequence of interactions between the user and iMove, extracted from user's log records ordered by timestamp. It captures both the type of the log entry (i.e. screen, action,



(a) Screens. (b) Actions. (c) Notifications.

Figure 4: Distribution of log records highlighting differences between speech and text geo-notes logs.

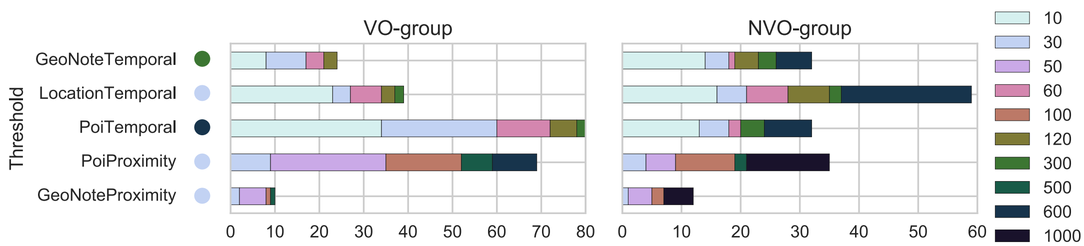


Figure 5: Preference log records across users in VO-group and NVO-group.

or notification) and the magnitude of time gaps between two consecutive log entries. Precise time gap values are omitted if the log entries belong to the same session (defined in Section 3.2) and are represented by the symbol "|" if they denote session boundaries. Figure 7 illustrates an example of this approach for obtaining a discrete user $i$stream.

Mapping users to an intuitive feature space. We treat $i$streams as text sentences and adopt n-gram-based text representation, a common practice in natural language processing. We consider three classes of records: screen enters, actions and notifications. Each of these three classes is defined as a set of atomic strings, which are dented by A_s (screen enters), A_a (actions), and A_n (notifications). For example, the string "s-Root" $\in A_s$ represents an entrance in the root screen; "a-navigateToPOI" $\in A_a$ represents the action of getting the navigation instructions to a POI; and "n-Location" $\in A_n$ represents the location notification. We define an $i$stream as a sequence $S = (s_1 s_2 ... s_m)$, where $s \in A_s \cup A_a \cup A_n \cup \{|\}$ and m is the total length of the $i$stream. We define F_n as the set of all possible n-grams (n consecutive elements) from all the users' $i$stream sequences: $F_n = n\text{-gram}(S_1) \cup n\text{-gram}(S_2) \cup ... \cup n\text{-gram}(S_{\#users})$. For each user $i$stream we calculate the normalized frequencies of the n-grams in F_n. We experimented with different values of n in the n-gram and chose 5-grams for our analysis, though 4-grams and 3-grams reveal similar clusters. As discussed in [25], intuitively, a larger value of n for the n-gram captures longer subsequences that are unlikely to repeat as a pattern in the $i$stream. For the above calculations we use the NLTK platform [5].

Constructing a similarity graph. We create a fully connected graph where each node represents a user and each edge between a pair of users represents the weight based on their pairwise similarity score. To calculate the similarity score between two users, we compute the cosine similarity of their n-gram feature vectors using scikit-learn [21].

Clustering and identifying primary features. We partition the graph into clusters of similar users with community detection using the Louvain method[5] described in [6]. To interpret cluster meaning, we isolate the primary features responsible for a cluster by performing feature selection based on Chi-square statistics (χ^2) [27]. For each cluster, we build a classifier that distinguishes users belonging to that cluster from the remaining users. Then we select the top k features with the highest discriminating power in separating the two classes using the "SelectKBest" method from scikit-learn [21].

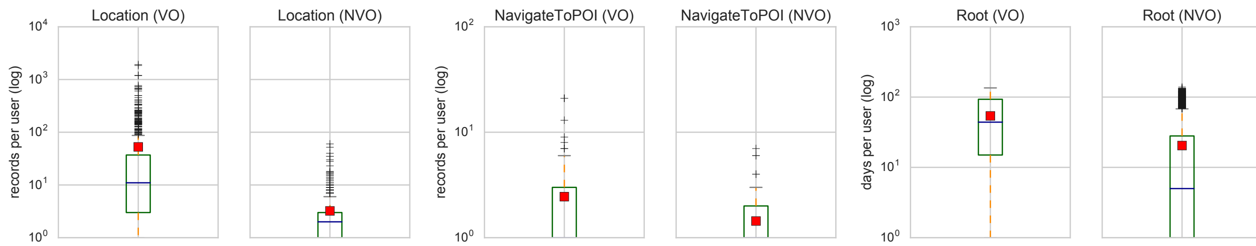
Results and Interpretation

The clustering procedure generates 9 clusters with a modularity of 0.47, where modularity [19] is a widely-used metric to assess the quality of a graph's partition into communities. Loosely speaking, it measures the density of edges inside clusters to edges outside clusters with values in the $[-1, 1]$ range, where a higher value indicates better clustering. Five of the detected clusters contain a total of 6 outlier users which we omit from the following discussion, hence focusing on four clusters with many users. Figure 8 visualizes the resulting clusters and the top 3 features with the highest discriminating power per cluster.

The first cluster (C1) contains 370 users. From the 5 primary features: two indicate that short sessions, in which the user simply opens the application without further interaction, appear with lower normalized frequency for users in C1 than those outside C1; one indicates that long sessions with many consecutive location notifications appear with low frequency as well; last, the remaining two primary features indicate that sessions in which the user navigates iMove screens with the list of POIs and their details have higher frequency for users in C1 than the rest. We can infer that users in this cluster often open the application to check the list of nearby POIs and their details.

The second clusters (C2) contains 247 users. From the 5 top features characterizing this cluster, three indicate high

[5] Library: http://perso.crans.org/aynaud/communities



(a) Location notifications. (b) Navigate to POI action. (c) Period of use.

Figure 6: Differences between users in VO-group and NVO-group.

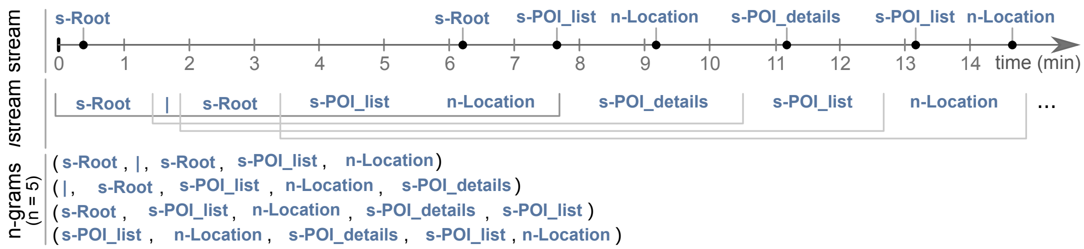


Figure 7: Mapping interaction streams to n-grams.

frequency of location and POI notification sequences in a single session for users in C2; and the remaining two primary features indicate low frequency of "empty" sessions, e.g., "| screenRootEnter | screenRootEnter |". These features suggest that C2 is a set of users running the application for long sessions during which they frequently receive many location and POI notifications.

The third cluster (C3) contains 198 users. In this case four of the 5 primary feature denote high frequencies of short "empty" sessions; and feature points to lower frequency of consecutive location notifications within the same sessions for users in C3 than outside C3. These features suggest that C3 contains users that starts the application, do not wait for any notifications, and then close the application. We speculate C3 users often open iMove simply to read (though VoiceOver) the current address.

The fourth cluster (C4) contains 154 users. All 5 primary features have high frequencies of short sessions with some location and POI notifications. Our interpretation is that these users start iMove and listen to one or two notifications without any further interactions.

To get a confirmation of the semantics we associate to each cluster, and to further study these clusters, we analyze user characteristics across clusters. We consider the average session length per user, computed as the distance between timestamps of the last and first records in each session. As shown in Figure 9(a), users in C2 have longer sessions that the other users. This supports our earlier interpretation based on the primary features. Figure 9(b) shows that users in C2 also have a higher number of sessions, followed by users in C3 and C4. We can interpret this observation in two ways. First, given the particular use of the app (keeping iMove active while moving), users in C2 tend to use it more frequently (e.g., every day, commuting to work). A second interpretation is that more experienced users of iMove tend to use it for longer sessions and hence belong to C2. Distinguishing these two cases requires additional analysis that we leave as future work. Last, Figure 9(c) shows that C1 users have a higher rate of records corresponding to POI details screen enters. This is in support of the primary features extracted for this cluster, identifying C1 as a user group with higher frequency of sessions that explore POI-related screens.

5. CONCLUSIONS AND FUTURE WORK

This paper presents an analysis of users interactions with iMove, a mobile app that supports the orientation of people with visual impairment. The initial dataset contains more than $17,000$ users, many of which are "incidental" users, not really interested in the functions of the app. To filter these users out, we adopted a session-based heuristic that eliminates 77% of the users and 67% of the log records.

The data analysis performed on about $4,000$ remaining users, highlights a number of iMove use properties, including commonly used functions and users' preferred values for settings parameters. In summary:

- While initial iMove settings favored sporadic and brief notifications, we observed that users, in particular those with severe visual impairments, prefer to have frequent and detailed information about the current location, which should include city, speed, heading and course.

- Applications similar to iMove are recommended to activate the "prevent screen lock" option by default.

- iMove users favored speech over text for input when creating notes associated to geographical locations.

- We observed that points of interest (POIs) were important in iMove functionality. Many users checked the list of nearby POIs (the third most visited screen) and the most popular action was navigating to a POI.

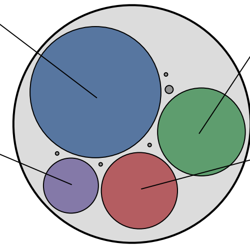
- VoiceOver users (VO-users) received more notifications, made intensive use of core iMove functions, and used the app for longer periods than other users. While

c1 (370 👤): Check list and details of nearby POIs

feature space	normalized frequency in cluster	normalized frequency not in cluster
(s-POI_list, n-Location, s-POI_details, s-POI_list, n-Location)	0.013078	0.001167
(s-Root, |, s-Root, |, s-Root)	0.002553	0.073674
(n-Location, n-Location, n-Location, n-Location, n-Location)	0.000192	0.046369

c4 (154 👤): Short sessions with few notifications

feature space	normalized frequency in cluster	normalized frequency not in cluster
(|, s-Root, n-Location, |, s-Root)	0.024224	0.001708
(n-Location, |, s-Root, n-Location, |)	0.036877	0.001331
(|, s-Root, n-POI, n-Location, |)	0.023818	0.001058



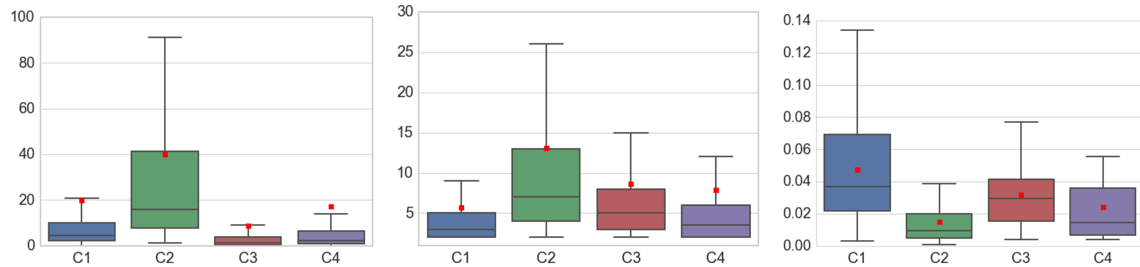
c2 (247 👤): Long sessions of location and POI notifications

feature space	normalized frequency in cluster	normalized frequency not in cluster
(n-Location, n-POI, n-Location, n-Location, n-Location)	0.021846	0.000411
(n-POI, n-Location, n-Location, n-Location, n-Location)	0.016155	0.000411
(n-Location, n-Location, n-Location, n-Location, n-Location)	0.175919	0.000615

c3 (215 👤): Very short sessions of only Root screen entrances

feature space	normalized frequency in cluster	normalized frequency not in cluster
(s-Root, |, s-Root, |, s-Root)	0.222907	0.001778
(|, s-Root, |, s-Root, |)	0.186638	0.000567
(n-Location, n-Location, n-Location, n-Location, n-Location)	0.000361	0.056407

Figure 8: Clustering results.



(a) Sessions length (minutes). (b) Number of sessions. (c) Ratio of s-POIdetails records.

Figure 9: Analysis of the four clusters.

> iMove was designed with blind users in mind, the observed differences with non-VoiceOver users, possibly including people with low vision, raises concerns about the app design in support of this population.

iMove was designed with a main user target in mind: people with visual impairment that would keep the app active along a route to get notifications. By clustering about $1,000$ iMove VO-users based on common interaction patterns, our user target base was successfully identified from one of the major clusters (C2), which contained 25% of the VO-users. In addition, our clustering method was able to capture and provide semantics for the remaining 75% of the VO-users with three more clusters; indicating those users who interact with the app in short sessions. We speculate that users in those clusters avoid interacting with the app while moving, because they do not want to be distracted or do not feel comfortable walking while holding their smartphone. Hence, they use the app in short bursts when they feel comfortable.

The identification of additional user clusters, other than C2, can help improve iMove by designing new interaction patterns and functions that support these usage patterns. For example, since many users (those in C1) often open the app to check nearby POIs, it may be possible to optionally show the list of POIs in the first app screen. Similarly, we speculate that users in C1 often open the app to check current address and close it. To support these operations, researchers could investigate different interaction modalities like an accelerometer-based interface to determine when the user wants to read the current address while the device is in the user's pocket.

This contribution highlights a number of possible future works. First, the analysis was conducted from data collected in a period of four months during which iMove has been downloaded on average more than $4,000$ times each month. We expect the number of users to grow linearly with time so that in few months it will be possible to conduct the same analysis on a larger set of users and adopt hierarchical clustering that can potentially refine our higher-level clusters into more descriptive sub-clusters. On the other hand, collecting data for a longer period will enable better analysis of a user's learning curve and evolution of interactions over time, possibly characterizing the behavior of novice users with respect to experienced ones.

In the future it will also be possible to collect additional types of log data. For example, while it is not possible to collect users' location or user-defined geo-notes due to privacy concerns, it may be possible to collect additional context-related information, like users' speed and whether users are walking or traveling on a bus/car.

From the point of view of users' clustering, there are three directions along which we intend to extend this contribution. First, we want to explore hierarchical clusters and dimensionality reduction approaches that can further improve our clustering quality and preserve an interpretable feature space. Second, we intend to investigate the link between preferences for user settings and the automatically detected user clusters. Third, we intend to experiment with clustering techniques for effectively identifying "incidental users" so that it is possible to remove them more reliably.

We see the results, methods, and data provided in this paper to improve existing applications, provide guidance, and advance the state of art in the field of assistive orientation and navigation – ultimately leading to a better experience of independent mobility for people with visual impairment.

6. ACKNOWLEDGMENTS

Chieko Asakawa acknowledges support from Shimizu Corporation. Sergio Mascetti was partially supported by grant "Fondo Supporto alla Ricerca 2015" under the project "Assistive Technologies on Mobile Devices". Authors are grateful to Valeria Alampi who assisted in the iMove development.

7. REFERENCES

[1] J. G. Adair. The hawthorne effect: A reconsideration of the methodological artifact. *Journal of applied psychology*, 69(2):334, 1984.

[2] D. Ahmetovic, C. Gleason, K. M. Kitani, H. Takagi, and C. Asakawa. Navcog: turn-by-turn smartphone navigation assistant for people with visual impairments or blindness. In *Proceedings of the 13th Web for All Conference*. ACM, 2016.

[3] A. Arditi, J. D. Holtzman, and S. M. Kosslyn. Mental imagery and sensory experience in congenital blindness. *Neuropsychologia*, 26(1):1–12, 1988.

[4] J. P. Bigham, A. C. Cavender, J. T. Brudvik, J. O. Wobbrock, and R. E. Ladner. Webinsitu: a comparative analysis of blind and sighted browsing behavior. In *Proceedings of the 9th international ACM SIGACCESS conference on Computers and accessibility*, pages 51–58. ACM, 2007.

[5] S. Bird, E. Klein, and E. Loper. *Natural language processing with Python*. " O'Reilly Media, Inc.", 2009.

[6] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[7] M. Böhmer, B. Hecht, J. Schöning, A. Krüger, and G. Bauer. Falling asleep with angry birds, facebook and kindle: a large scale study on mobile application usage. In *Proceedings of the 13th international conference on Human computer interaction with mobile devices and services*, pages 47–56. ACM, 2011.

[8] E. L. Brady, D. Sato, C. Ruan, H. Takagi, and C. Asakawa. Exploring interface design for independent navigation by people with visual impairments. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility*, ASSETS '15, pages 387–388, New York, NY, USA, 2015. ACM.

[9] S. Chennuru, P.-W. Chen, J. Zhu, and J. Y. Zhang. Mobile lifelogger–recording, indexing, and understanding a mobile userâĂŹs life. In *Mobile Computing, Applications, and Services*, pages 263–281. Springer, 2010.

[10] N. A. Giudice and G. E. Legge. Blind navigation and the role of technology. *Engineering handbook of smart technology for aging, disability, and independence*, pages 479–500, 2008.

[11] M. Goldberg. *Assisting Visually Impaired People with Mobility through Technology in the Age of Context*. PhD thesis, City University of New York, 2015.

[12] A. Hurst, S. E. Hudson, J. Mankoff, and S. Trewin. Distinguishing users by pointing performance in laboratory and real-world tasks. *ACM Transactions on Accessible Computing (TACCESS)*, 5(2):5, 2013.

[13] Q. Li, Y. Zheng, X. Xie, Y. Chen, W. Liu, and W.-Y. Ma. Mining user similarity based on location history. In *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*, page 34. ACM, 2008.

[14] S. Mascetti, D. Ahmetovic, A. Gerino, and C. Bernareggi. Zebrarecognizer: Pedestrian crossing recognition for people with visual impairment or blindness. *Pattern Recognition*, 2016.

[15] S. Mascetti, D. Ahmetovic, A. Gerino, C. Bernareggi, M. Busso, and A. Rizzi. Robust traffic lights detection on mobile devices for pedestrians with visual impairment. *Computer Vision and Image Understanding*, 148:123–135, 2016.

[16] S. Mascetti, C. Bernareggi, and M. Belotti. *TypeInBraille: quick eyes-free typing on smartphones*. Springer, 2012.

[17] S. Mascetti, L. Picinali, A. Gerino, D. Ahmetovic, and C. Bernareggi. Sonification of guidance data during road crossing for people with visual impairments or blindness. *International Journal of Human-Computer Studies*, 85:16–26, 2016.

[18] M. Nakajima and S. Haruyama. New indoor navigation system for visually impaired people using visible light communication. *EURASIP Journal on Wireless Communications and Networking*, 2013(1):1–10, 2013.

[19] M. E. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical review E*, 69(2):026113, 2004.

[20] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135, 2008.

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

[22] D. Riboni, C. Bettini, G. Civitarese, Z. H. Janjua, and R. Helaoui. Smartfaber: Recognizing fine-grained abnormal behaviors for early detection of mild cognitive impairment. *Artificial Intelligence in Medicine*, 2016.

[23] V. R. Schinazi, T. Thrash, and D.-R. Chebat. Spatial navigation by congenitally blind individuals. *Wiley Interdisciplinary Reviews: Cognitive Science*, 7(1):37–58, 2016.

[24] C. Thinus-Blanc and F. Gaunet. Representation of space in blind persons: vision as a spatial sense? *Psychological bulletin*, 121(1):20, 1997.

[25] G. Wang, X. Zhang, S. Tang, H. Zheng, and B. Y. Zhao. Unsupervised clickstream clustering for user behavior analysis. In *SIGCHI Conference on Human Factors in Computing Systems*, 2016.

[26] M. A. Williams, A. Hurst, and S. K. Kane. Pray before you step out: describing personal and situational blind navigation behaviors. In *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility*, page 28. ACM, 2013.

[27] Y. Yang and J. O. Pedersen. A comparative study on feature selection in text categorization. In *ICML*, volume 97, pages 412–420, 1997.