Insights on Assistive Orientation and Mobility of People with Visual Impairment Based on Large-Scale Longitudinal Data

Insights on Assistive Orientation and Mobility of People with Visual Impairment Based on Large-Scale Longitudinal Data

HERNISA KACORRI, University of Maryland – College Park
SERGIO MASCETTI and ANDREA GERINO, Università degli Studi di Milano, EveryWare Technologies
DRAGAN AHMETOVIC, Carnegie Mellon University
VALERIA ALAMPI, Università degli Studi di Milano
HIRONOBU TAKAGI, IBM Research
CHIEKO ASAKAWA, Carnegie Mellon University

Assistive applications for orientation and mobility promote independence for people with visual impairment (PVI). While typical design and evaluation of such applications involves small-sample iterative studies, we analyze large-scale longitudinal data from a geographically diverse population. Our publicly released dataset from *i*Move, a mobile app supporting orientation of PVI, contains millions of interactions by thousands of users over a year.

Our analysis: (i) examines common functionalities, settings, assistive features, and movement modalities in *i*Move dataset, and (ii) discovers user communities based on interaction patterns. We find that the most popular interaction mode is passive, where users receive more notifications, often verbose, while in motion and perform fewer actions. The use of built-in assistive features such as enlarged text indicate a high presence of users with residual sight. Users fall into three distinct groups: C1) users interested in surrounding points of interest, C2) users interacting in short bursts to inquire about current location, and C3) users with long active sessions while in motion. *i*Move was designed with C3 in mind and one strength of our contribution is providing meaningful semantics for unanticipated groups, C1 and C2. Our analysis reveals insights that can be generalized to other assistive orientation and mobility applications.

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

Additional Key Words and Phrases: behavior analysis, clustering, people with visual impairment, orientation, navigation

0:2 • Kacorri et al.

1 INTRODUCTION

To acquire information about their surroundings, people with visual impairment (PVI) rely to a larger extent on other senses to compensate for sight loss. Auditory, tactile, olfactory, thermal, vestibular, and other non-visual inputs can help PVI create a spatial mental model of the environment. However, non-visual exploration is characterized by a shorter sensory horizon [35] and lower information throughput [20]. Thus, acquiring spatial information and navigating based on non-visual sensing is slower and more cognitively demanding [7].

Different types of visual impairment affect access to spatial information differently. For example, while a sighted person can explore a great portion of the surroundings with a glance, individuals with tunnel vision can scan only a small portion of the surroundings at a time [17]. This sequential access to visual information can slow down the formation of a spatial mental model, leading to danger in the presence of fast moving vehicles that cannot be quickly identified and tracked. Instead, a reduced visual acuity provides concurrent sensing in all directions, but the quality of the formed image is limited. This diminishes access to distant or small visual cues, which need to be physically approached for exploration.

Assistive technology for navigation can address these challenges by supplementing or substituting the orientation and navigation capabilities of PVI. The solutions proposed in the literature, surveyed in [20, 21], adopt many different technological approaches, such as laser canes, sonar devices, and GPS localization. To design and evaluate their assistive technology, researchers typically rely on supervised user studies. Formative studies (e.g. [50]) are used to shape the design direction and gain insight of the user base needs and requirements. Wizard-of-Oz experiments (e.g. [13]), where the experimenter ("wizard") simulates the behavior of a system behind the scenes, allow investigation with prototype applications or, in some cases, even without working prototypes. Evaluation studies (e.g. [33]) can provide information on the efficacy and user satisfaction of the technology proposed. These experiments can be conducted in controlled conditions, and the participants' characteristics may be predefined or are known in advance.

One major limitation of supervised user studies is that they are constrained to specific scenarios or laboratory environments. The limited number and variety of settings that may be reproduced do not reflect the diversity of real-world situations. Additionally, user studies performed under observation may be affected by the Hawthorne effect [1], where participants behave differently due to their awareness of being observed. More importantly, in supervised studies, the participant pool is usually geographically constrained to the proximity of the physical location where the experiment is conducted. This can impair user representativeness and lead to cultural, gender or age related bias. In accessibility research, where the number of locally available users may be very limited, this issue is more severe, making it more challenging to scale the user studies in terms of number of involved subjects and study length [20].

In contrast to supervised studies, we perform extensive analysis of remotely collected usage data from an assistive application that supports orientation and mobility of PVI. This approach allows us to capture the behavior of a vast and diversified user pool in the wild without observation bias. Specifically, we investigate users' behavior through their interactions with *i*Move[1], a GPS-based mobile application that supports outdoor orientation of PVI. *i*Move provides information about a user's surroundings, such as nearby landmarks, current address, and their notes related to the location. This information assists the user during way-finding, and can help the user construct a mental map of their environment. It is provided both visually and through native system accessibility tools available on iOS, and therefore appeals to users having different visual impairments or blindness, as described in Section 3.1.

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

Insights on Assistive Orientation and Mobility of People with Visual Impairment Based on Large-Scale
Longitudinal Data   •   0:3

1.1 Motivation for this Article

Our analysis primarily aims at providing a deeper understanding of prolonged real-world interactions, observed between users and an assistive technology such as *i*Move. Specifically, we are interested in the most frequently accessed functionalities, the assistive technology features activated on users' mobile devices while interacting with *i*Move, their preferred settings, and more importantly the discovery of prevalent interaction patterns across users. To this end, we perform automatic clustering of users into groups that adopt diverse usage strategies while interacting with *i*Move, and we highlight behaviors common among the members of the identified groups. We also investigate users' mobility status while using *i*Move, and we discover common behaviors that characterize our users. For example, we notice that users prefer to actively interact with the app while stationary or on a vehicle, while they prefer just receiving notifications while walking.

One motivation for this work is that knowledge of how users habitually interact with the application may be used to improve the existing system capabilities, adapt default settings to better accommodate different groups of end users, and examine data-driven personalization approaches similar to [26]. Moreover, we believe that these results also provide insights for future mobility assistance applications. For example, the identification of a major cluster in our initial analysis [27], where users sporadically interact with *i*Move just to confirm their location, was one of the inspirations behind the virtual navigation in [22]. We suspected that those users already possessed a mental representation of their environment and used *i*Move for confirmation purposes. The goal of [22] was to help users build these mental models a priori while virtually exploring a new environment.

Like other research groups (discussed in Section 2.3 and more extensively in [19, 20]), we are interested in supporting the orientation and mobility of PVI both in outdoor [31, 32] and indoor environments [4, 5]. While these environments may pose different challenges to PVI, e.g. finer-grained accuracy required in indoor areas [4], their user interfaces and interactions inherently share similarities and many researchers (e.g. [43, 52]) are working on integrated solutions that seamlessly support both. Therefore, another motivation for this article is to provide real-world longitudinal data, analysis methods, and insights from observations that can help formulate testable hypotheses as well as allow comparisons by future work in this area.

1.2 Overview of this Article

For the analysis of *i*Move usage data, we collected a dataset, described in Section 3, containing $3,784,700$ records of interaction between $61,715$ users and *i*Move over a period of 15 months. Using both inferential and exploratory methods, in Section 4 we examine commonly used functionalities, preferred settings, movement modalities, and the relationship between assistive iOS features with *i*Move-user interaction. We perform unsupervised discovery of user clusters based on common behavior patterns in Section 5. For this purpose we employ natural language processing and machine learning methods. A discussion on the findings, limitations, and future work is included in Section 6.

In this paper we extend our previous work [27] by introducing the following contributions:

- We expand the analysis of user preferences and user behavior with *i*Move to include a much longer time span. The initial *i*Move dataset comprised 5 months of the user interaction logs with the system, totalling $771,975$ records, while the new dataset contains $3,784,700$ records, collected for over 15 months.
- We refine the detected user clusters into more descriptive sub-cluster by performing hierarchical clustering and we explore dimensionality reduction approaches to further improve our clustering quality and preserve an interpretable feature space.
- We collect and analyze new data related to interface accessibility features activated by the user on the mobile device. Based on this data, we segment the users in different visual impairment categories. We also collect data about the user's speed and movement modality (e.g., stationary, walking, on a vehicle).

- We investigate the relations between the identified user clusters and the new movement modality information, as well as the new visual impairment classification. We also investigate the link between user preferences and the automatically detected clusters. The results could further 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.

2   RELATED WORK

2.1   Remote Evaluation of Assistive Technologies

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. For the evaluation of assistive technologies, conducting behavioral studies over long periods of time and with large samples of participants with disabilities is challenging. Instead, studies are often conducted with a small number of users in a controlled environment [4, 5, 31, 32, 36]. Thus, only a few 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.

Bigham et al. propose the WebinSitu system to automatically collect user actions during web browsing by PVI [10]. Authors argue for the importance of conducting remote evaluations by observing that it is hard to replicate in the lab the various assistive technologies and configurations normally used by blind users. This is in common with our approach but we also observe that remote evaluation is even more important with mobile applications, as in this case it is even harder to replicate the context of use such as a diverse outdoor environment.

In [23], log data from real-world tasks are collected to assess the pointing problems of older adults and individuals with motor impairment. Authors argue that laboratory data may not be representative of natural behaviour for a number of reasons, including the fact that subjects are observed, task can be unrealistic, and tools are unfamiliar to the subjects. Authors hence recognize the benefits of remote evaluations, but they also identify two main challenges: to interpret user intent and to segment real-world data. These challenges are not present in laboratory studies, in which the users are generally assigned tasks with a clear objective that can be separated. This is not the case when data is collected "in the wild"; these challenges also arose in our research and are further discussed in Section 6.2.

Another example of remote data acquisition is presented by Riboni et al. [44] with the aim of detecting mild cognitive impairment. The paper describes the installation of a number of sensors in a smart home where the activities of an elderly woman were monitored for 55 days to detect behavioural anomalies. Authors observe that, due to privacy concerns, it was not possible to directly observe the execution of activities. Similarly, privacy issues arose while designing data collection in *i*Move and for this reason any re-identifying information (including location and user notes) was not logged.

Similar to this prior work, our contribution adopts a remote data-acquisition technique that makes it possible to record natural interactions of *i*Move users. One important difference to prior work is that our contribution presents tests conducted on a different scale. For example, the number of subjects involved in the remote tests in the three papers presented above is 20, 6 and 1, respectively; our work, as discussed in the next section, considers thousands of subjects. Furthermore, WebinSitu considers a total of 325 hours of use, while we consider a total of about 5,000. This difference in scale resides in subjects' recruitment and motivations. Indeed, an explicit recruitment process was required in the three studies above and, in the case of WebinSitu, subjects were paid to participate. Instead, we rely on the fact that subjects use *i*Move because it provides an useful service and, incidentally (from the user's point of view), it also logs usage data. While this facilitates collection of usage data, it poses certain challenges, such as collecting user demographics.

2.2   Behavior Analysis on Large-Scale Datasets

For human-computer interaction studies that involve broader participant pools, behavior analysis on large-scale data is adopted more frequently (e.g. [12, 16, 18]). These analyses often combine data-driven approaches from many fields such as classification, clustering, and time-series analysis from machine learning [28, 34, 48], sentiment analysis from natural language processing [38, 48], and community detection from network analysis [42].

For example, remotely collected large-scale smartphone usage data reveal user interactions with mobile applications across the day span [12, 18]. Similar to our analysis, the interactions were considered in sequence. However, these sequences were more coarse-grained and captured transitions between application categories. Inspired by their preliminary results on the impact of location and movement modality we extended the *i*Move dataset with motion sensing data such as walking, automotive, running and analyzed user behavior in this context (Section 4.3). However, due to privacy concerns *i*Move data did not include information on user location, a popular feature to cluster users ([28], [42]).

The work of Wang et al., 2016 [48] is the closest to our analysis, where similarity among social network users is detected with natural language processing techniques similar to document clustering. Specifically, users are clustered based on their sequences of clicks. Then the most common patterns, short subsequences of those clicks among users within the same cluster compared to users outside the cluster, are used to interpret cluster formation. We extend this approach by incorporating the notion of a session based on an inactivity threshold (as in Meier et al. [34]). While alternative techniques that are tolerant to specific forms of permutable, redundant and omittable user-interaction patterns within sequences have been recently proposed [16], for comparison purposes we follow a similar approach to our initial analysis [27].

2.3   Supporting Orientation and Navigation of PVI

PVI often learn, through O&M training [29, 49], to adopt sophisticated navigation strategies to safely sense and traverse the surrounding environment. Prior literature [45, 47] in cognitive sciences related to spatial representation and navigation by PVI highlights that the way-finding capabilities among sighted, early blind, and late blind individuals are similar, but individuals often rely on profoundly different preferred navigation strategies. Thus, Shinazi et al. [45] argue that orientation and mobility performance in diverse individuals should not be evaluated using some predefined navigation strategies. Instead, for a fair comparison, each participant should be able to rely on the most appropriate and familiar set of navigation strategies.

In addition to O&M training, many diverse mobility assistive tools have been proposed to support the orientation and navigation capabilities of PVI. Environment augmentations, such as tactile paving [24] or audio cues near pedestrian crossings [40], provide sensory substitution mechanisms for visual cues within the environment and help PVI localize themselves and maintain orientation [29]. Carried sensing instruments expand the sensory horizon of the user by detecting cues outside of their haptic proximity and therefore help them to learn the structure of their surroundings [29]. This category includes a white cane, ultrasonic sensing of obstacles [41], computer vision based detection of visual cues [2, 6, 31, 32], and GPS location based services such as *i*Move (See Section 3.1). Mixed approaches couple environment augmentations with carried sensing devices. For example, the NavCog [3–5] system relies on Bluetooth beacons installed in the environment, sensed by a smartphone carried by the user.

The discussion on adopted and preferred navigation strategies among PVI raises the question 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. [48], that automatically discover user clusters based on streams of interactions with *i*Move. Moreover, we examine how well these clusters capture users' settings preferences in an assistive orientation application when compared to their inferred visual impairment (e.g., blind versus low vision). However, the link between these clusters and underlying user-adopted

navigation strategies is beyond the scope of this article since the estimation of these navigation strategies can not be captured at large without raising privacy concerns.

3 *i*MOVE APP AND DATASET

We present the iMove app (Section 3.1), its remote logging system (Section 3.2), and the collected data with descriptive statistics (Section 3.3).

3.1 *i*MoveApp

*i*Move is an iOS application designed to support orientation of PVI. 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., by selecting the "around me" function in the main screen that shows the list of nearby POIs (Fig. 1(b)), or periodically by turning on the "Notify me" toggle button. 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)).

The app is accessible visually, and also through built-in accessibility tools (ATs) available on iOS devices. As shown in Section 4.4, the most common of these tools activated by *i*Move users is VoiceOver, which provides access to built-in and other compatible applications through audio feedback and can be used by users who are blind or with low vision. The other ATs are designed to improve accessibility for users with low vision, for example presenting enlarged fonts.

*i*Move is designed to be highly customizable: users can specify the categories of POIs they are interested in, activate the automatic reading of surrounding information and modify settings related to the system verbosity. This latter aspect (verbosity) needs to balance two contrasting needs: on one side users would like to receive frequent updates, each one with detailed orientation information. However, on the other side, since this information is provided through audio, a verbose system can divert users' attention from ambient noise. *i*Move allows the user to tune a number of parameters related to this aspect, including, for example, how often (in terms of both space and time) the updated address should be read aloud.

3.2 Remote Logging System

Since *i*Move 2.0, released on December 8, 2015, the application implements a remote logging system that makes it possible to collect anonymous app usage information[2]. Logging is supported by a client library in *i*Move that communicates with a REST server to store data on a non-relational database.

The collected data are made available online, together with a detailed description of their format[3]. Data are collected in anonymized form. Thus, data do not include location (e.g., address), location-related information (e.g. nearby POIs) or user-generated content, e.g. geo-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 one 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 *i*Move build version (we collected data for build versions 31, 32, 33, 34, 38) 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 also described in Appendix A:

> **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.

---

[2]Users are informed of the data logging process at first app run, when they accept EULA (End User Licence Agreement) and privacy policy.
[3] https://ewserver.di.unimi.it/taccesssim17/

(a) Root screen.   (b) POI screen.   (c) Edit text-note screen.   (d) Route selection screen.
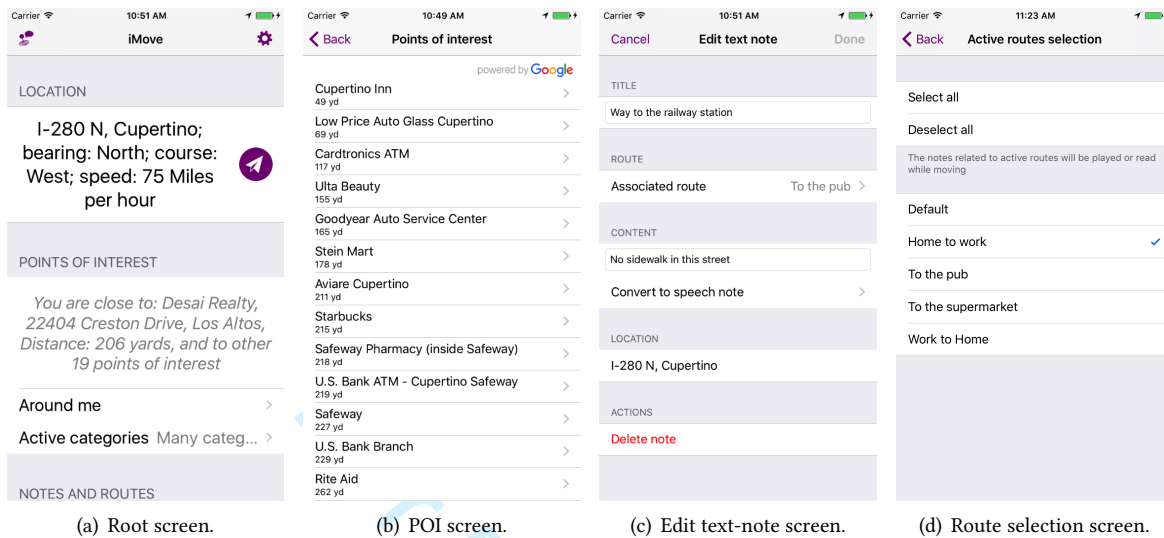
Fig. 1. Main screens of the iMove application.

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 either when a user changes the *i*Move settings or when the app is started and a new value is detected for a system accessibility-related option. A preference log lists the name of the modified parameter, its old value, and its new value.

The logging system evolved with the different versions of *i*Move. Changes were introduced to support new app functions and parameters[4] as well as an updated logging system on user profiling. In particular there are two major updates. Starting from version 34 *i*Move registers users' mobility context, in the form of users' speed and current activity ("stationary", "walking", "running", "automotive", "cycling", "unknown"), together with a confidence level. With version 38 *i*Move also logs which ATs are in use.

3.3 *i*Move Dataset Overview

The *i*Move dataset (*DS*1) was collected during the December 2015 - March 2017 period with descriptive statistics presented in Table 1. 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 PVI. For example, some users confused *i*Move with *iMovie*, a popular application for video editing.

To filter out these "incidental" 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 records such that the time gap between each pair is less than 10 minutes. This constraint captures the intuition that the user might temporarily exit the app for a short time within an interaction session. The choice of using a 10 minutes threshold is driven by the fact that this is the maximum value that can be set as the

---

[4]Only minor changes in functions were introduced in the various versions considered for the analysis while several default values for parameters were changed, as discussed in Section 4

temporal distance between two consecutive location notifications in *i*Move. Based on the intuition that users who are uninterested in *i*Move would not use it for more than one session, we only consider logs from a subset of *DS*1, we call *DS*2, that includes users having two or more sessions.

Table 1. Descriptive statistics on the *i*Move dataset and its subsets considered in the analysis.

Dataset	Users	Records	μ records/user	σ records/user	Range records/user
DS1: all *i*Move users	61,715	3,784,700	61.33	235.63	$1 - 35,237$
DS2: $DS1_{users>2sessions}$	14,948	1,683,737	112.64	472.21	$2 - 35,237$
DS2-VO: $DS2_{\geqslant 1VoiceOver}$	2,560	749,235	292.67	1,111.95	$2 - 35,237$
DS2-NVO: $DS2_{0VoiceOver}$	12,388	934,502	75.44	73.98	$2 - 2,527$
DS2-B34: $DS2_{inst.\ version<34}$	8,506	952,416	111.97	605.53	$2 - 35,237$
DS2-A34: $DS2_{inst.\ version\geqslant 34}$	6,442	731,321	113.52	182.42	$2 - 6,157$
DS3: $DS2_{inst./updat.\ version\geqslant 38}$	3,456	742,378	214.81	930.67	$2 - 35,237$
DS3-AT: $DS3_{\geqslant 1access.\ tools}$	1,517	557,420	367.45	1,387.39	$4 - 35,237$
DS3-NAT: $DS3_{0access.\ tools}$	1,939	184,958	95.39	73.13	$2 - 1,422$

As discussed in our initial analysis [27], the VoiceOver field in the logs was the only indicator for distinguishing users that were likely to have severe visual impairment. To allow comparison with that prior study, we also partition *DS*2 based on the presence of VoiceOver logs, as shown in Table 1. Specifically, *DS*2-VO contains data from users with at least one VoiceOver-active record and *DS*2-NVO contains the rest of the users with no VoiceOver-active records.

Since the logging of users' mobility context started with *i*Move build version 34, we also partition *DS*2 users along a different dimension, based on the presence of users' motion status. In particular, *DS*2-B34 contains records from users that started using *i*Move before version 34 and *DS*2-A34 records of all the other installing the app after version 34.

As mentioned in Section 3.2, starting with *i*Move build version 38 we also log any user-activated system accessibility tools. Therefore, for all users that installed or updated *i*Move to version 38 we logged, at least once, their system accessibility preferences. The records from these users form dataset *DS*3 with descriptive statistics shown in Table 1. As with *DS*2, we further partition *DS*3 based on the presence of built-in accessibility tools. In particular, *DS*3-AT includes all records from users that had activated one or more accessibility tools at least once and *DS*3-NAT all the rest.

4 ANALYSIS OF *I*MOVE USE

In this section we analyze log records from *DS*2 and *DS*3 datasets to highlight commonly used *i*Move functionalities (Section 4.1), preferred settings (Section 4.2), and typical movement modalities during interactions (Section 4.3). In addition, we explore the most commonly used system accessibility tools (Section 4.4) and the differences between users that do and do not rely on any of these tools (Section 4.5).

4.1 Most Used Functions

We first explore how users interact with *i*Move by considering the app screens that the users visit. Appendix A reports the full list of iMove screens, each with a brief description. The same for actions, notifications and preferences. Figure 2 shows, for each *i*Move screen, the total number of times it was accessed calculated on the *DS*2 dataset. As expected, the root screen is the one accessed the most (213,298 times), followed by the main settings screen (41,967 times), and by the screen that shows the list of POIs around the user (33,783 times).
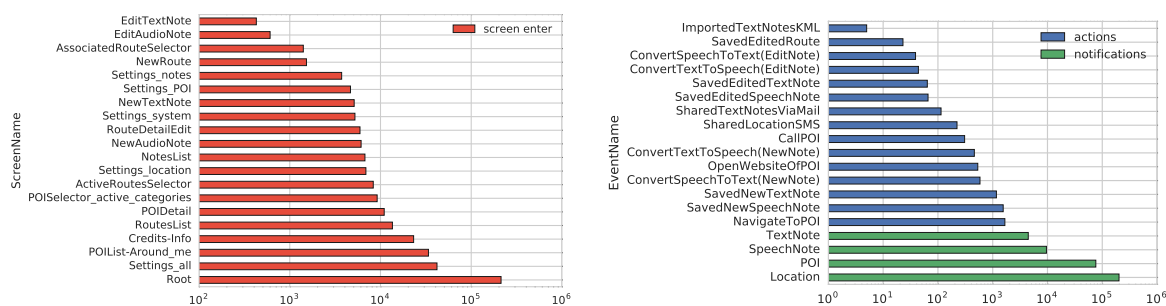
Fig. 2. Overall number of interactions with *i*Move for each of the screens, actions and notifications.

Considering notifications, 'Location', which reports the current address, is the most common one, followed by the 'POI', which reports the names of the nearby POI. Geo-notes, both textual and speech, are much less frequent, and their total number is one order of magnitude smaller than location and POI notifications. This is due to the fact that 90% of the users never created a geo-note. Among users creating a geo-note, the percentage of geo-note notifications ('TextNote' and 'SpeechNote') is 2.7% of the total notifications.

We observe that **explicit user actions are less frequent than notifications**, which is expected given the nature of the application. Interestingly, the 'NavigateToPOI' action, whose implementation was requested by many initial users of *i*Move and introduced in build version 31, is the most frequent user action.

4.2 *i*Move Settings and Preferences

*i*Move generates a log record every time a user preference is modified. These records account for 18% of the total log records in DS2. Figure 3 reports, for each preference setting, its default value and how many times it has been set to a given value. To better understand the impact of preference changes, it is important to consider the user-defined values jointly with the default preference values. However, the logged data cannot provide us the information on how many users intentionally choose to maintain the default value for a given parameter. We estimate this as the percentage of users that changed a parameter's default value at least once given all users that actually visited that parameter's settings screen. It is indicated in parentheses after the default value (Figure 3).

Our initial analysis [27] of the preferred settings values referred to *i*Move versions prior to 34. As shown in Figures 3(a) and 3(c), it highlighted a number of frequently-changed default settings when users visited their corresponding screens. Specifically, the verbosity settings 'SaySpeed', 'SayHeading', 'SayCourse', and 'SayCity', as well as the 'PreventIdle' setting were activated by the users. Moreover, the thresholds for the 'GeoNoteTemporal' and 'PoiTemporal' settings were lower. This observation led to new default values in the subsequent versions of *i*Move, which reflect the changes commonly made by the users. Specifically, the new default behaviour is for *i*Move to provide more detailed and frequent information.

To gain some insights on the users' reaction to these new default values we analyze their interactions with these settings on *i*Move versions 34 and after. As shown in Figure 3(b), we observe that the percentage of users that changed the new default values for 'SaySpeed', 'SayHeading', 'SayCourse', 'SayCity', and 'PreventIdle' is much lower. This may suggest that the new default values for these settings better capture users' needs. It indicates that **users prefer to have all available information, even at the cost of having a more verbose output speech**.

We also analyze users' changes to the threshold parameters for *i*Move versions 34 before and after. As shown in Figures 3(c) and 3(d)), there is no observed convergence in specific preferred values for these settings. Moreover,

(a) Number of changes for Boolean settings in *DS*2-B34.

(b) Number of changes for Boolean settings in *DS*2-A34.

(c) Number of changes for threshold settings in *DS*2-B34

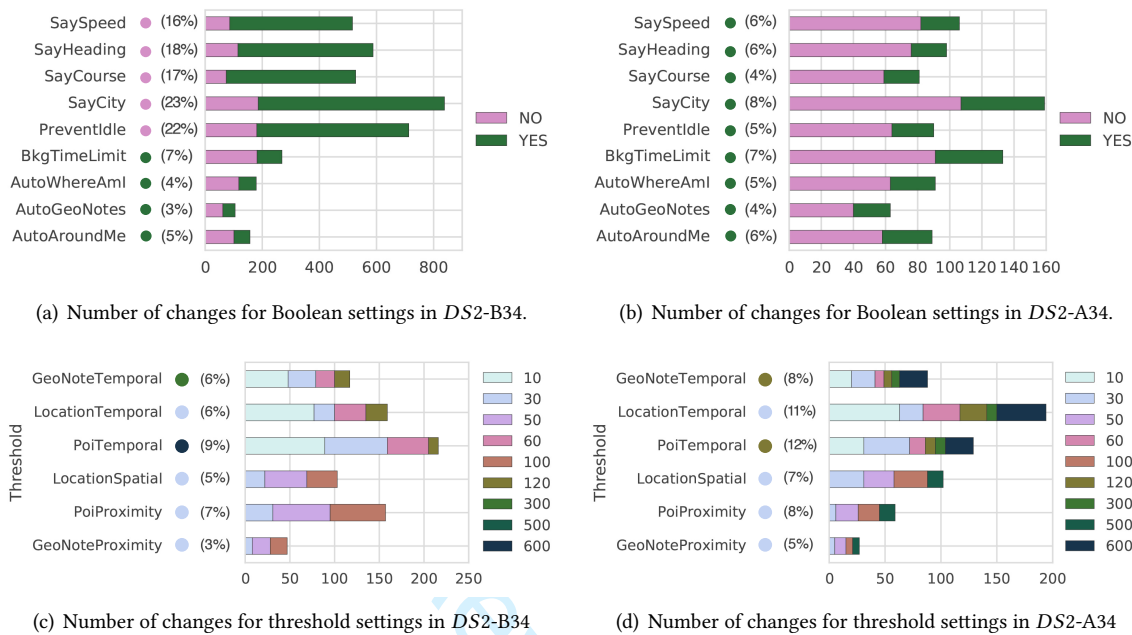(d) Number of changes for threshold settings in *DS*2-A34

Fig. 3. Changes in the settings preferences before *i*Move version 34 (left) and after (right). Defaults values are indicated with a circle following the setting's name. The percentage of users that changed a setting's default value at least once among users that visited the corresponding screen is reported in brackets.

we notice that in subsequent versions of *i*Move (*DS*2-A34), a higher percentage of users would adjust these parameters and their values tend to be more diverse. The same hold even for the 'GeoNoteTemporal' and 'PoiTemporal' settings that had new default values. We speculate that these threshold settings are affected by other factors such as residual sight, environment, context of use, and *i*Move experience. Specifically, we hypothesize that **users have very different needs on the proximity and frequency of notifications that they receive**.

4.3 Context of Use

Since the build version 34, *i*Move also logs information on the users' movement modality during user actions and notifications. The movement modality indicates whether the user is stationary, walking, running, cycling or on a vehicle (automotive). This information is available only for those users who have enabled this functionality on their mobile phones. There are 5,668 such users in *DS*2. As shown in Figure 4(a), we observe that the great majority of actions are performed while the user's movement modality is either 'stationary' or 'automotive'. On the contrary, more than 25% of notifications are received while users are walking (Figure 4(b)).

This suggests that **users avoid active interactions with the system while in motion, e.g. walking**. We believe that there are two possible explanations for this observation. One is that, when in motion, users concentrate on their activity and use their senses to preserve their safety. The other is that the interaction with the app involves the use of hands, that could be otherwise occupied (e.g., holding the guide dog or the white cane). In both cases users cannot actively interact with the device. We suspect that a user with visual impairment that is walking prefers to stop to interact with *i*Move.

(a) Number of user actions per modality.   (b) Number of notifications per modality.   (c) Automotive versus walking modality.
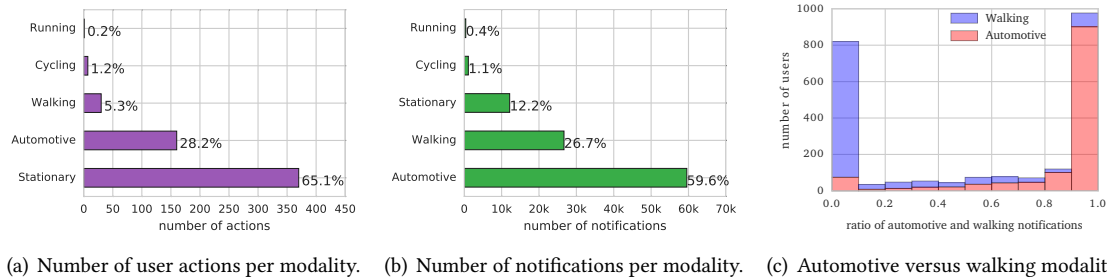
Fig. 4. User movement modality: actions and notifications.

In contrast, we suspect that **passive interactions such as notifications are acceptable while the user is in motion, e.g. walking**. As shown in Figure 4(b), we observe that the great majority of users receive notifications while walking or in a moving vehicle. Notifications are triggered based on time and space thresholds (e.g. time and distance from previous notification). Therefore, it is not a surprise that there is a high number of notifications while in a moving vehicle because users move faster and hence the distance constraint is easily met.

Moreover, we look deeper into the notifications from the perspective of the two common modalities: walking and automotive. We investigate whether users tend to receive notifications predominantly in a single modality. Figure 4(c) considers about 2, 300 users that receive notifications while walking or in a moving vehicle. The graph shows the number of users per ratio of automotive versus walking notifications. We observe that most users are polarized. More than 800 users received 90% or more notifications while walking. Likewise, almost 1, 000 users received more than 90% of notifications while in a moving vehicle. The other users (about 20%) receive at least 10% of their notifications in both modalities. This observation highlights **users' tendency to receive notifications in predominantly one moving modality** (either walking or automotive). It suggests that *i*Move could be personalized (e.g., in terms of settings values) to better adapt to the users, based on their preferred moving modality.

4.4 System Accessibility Tools

The additional contextual data collected by *i*Move in DS3 allows for a better understanding and categorization of *i*Move users based on the assistive technologies (ATs) that they are using to interact with their phones. Considering all users in DS3, we discover that 1, 939 do not use any system accessibility tool (these are the users grouped in DS3-NAT). An additional 614 rely on some ATs, but not on VoiceOver, the built-in screen reader on iOS. Thus, a total of 2, 553 user do not use VoiceOver, and therefore they surely rely on sight for interaction. Among those using VoiceOver, which we can expect to be blind or with severe visual impairment, 637 use at least one other accessibility tool in addition to VoiceOver, suggesting that they still have residual sight. Consequently, only 266 users base their interaction solely on audio feedback from VoiceOver and hence are likely to be blind. This is summarized in Table 2.

As shown in Table 2, less than 10% of *i*Move users may be blind, while many have sufficiently good residual sight to interact with the device without any AT. This suggests that **visual interaction tools (e.g., a map) even if not accessible through audio (e.g., through VoiceOver) can still be useful to many users of apps designed for PVI**.

Figure 5(a) shows that the most common AT among users in DS3-AT is VoiceOver, which is used at least once by 59.5% of users. The second most common tool, used by 39% of users, is "Enlarged Content Size". This accessibility tool allows resizing of the system fonts from XS to XXXL. The same tool allows the size of interface elements

0:12 • Kacorri et al.

Table 2. Inferred visual impairment based on the adopted ATs among users in DS3.

Activated ATs	Inferred disability	# users	% users
None	None or mild visual impairment	1, 939	56.1%
ATs-VO: ATs excluding VoiceOver	Low vision	614	17.8%
ATs+VO: ATs including VoiceOver	Low vision (limited residual sight)	637	18.4%
VO: VoiceOver only	Blind or severe low vision	266	7.7%

(a) ATs percentage of usage

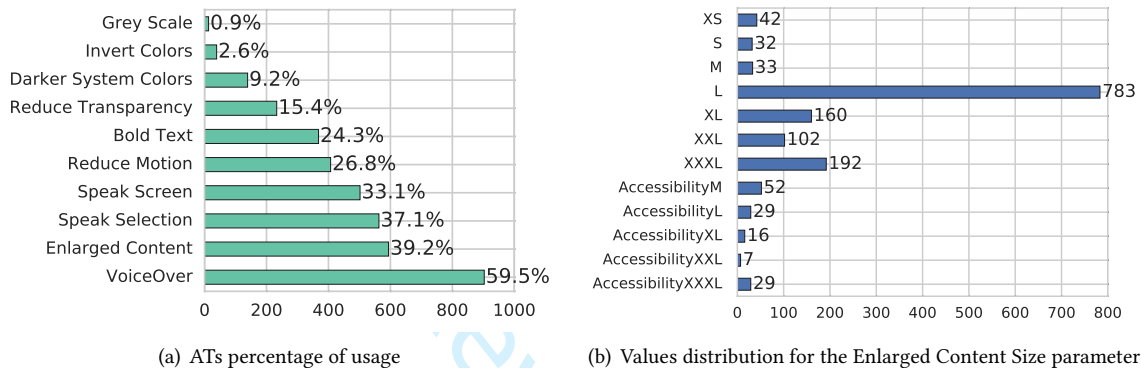(b) Values distribution for the Enlarged Content Size parameter

Fig. 5. Use of ATs among users in DS3-AT.

to be altered for better accessibility by PVI. This option is captured by settings ranging from AccessibilityM to AccessibilityXXXL. Figure 5(b) shows the detailed distribution of settings for "Enlarged Content Size". Values are ordered from the smaller content on the top, to the larger one on the bottom. "L" is the system default value, adopted by many of the *i*Move users (Figure 5(a) considers users with "enlarged content" those that select a value larger than **L**). Few *i*Move users select smaller content (7%), while 38.7% of the users select larger content. From this analysis we learn that, **when developing an assistive app for PVI, attention should be devoted to test the user interface with enlarged content, as this is a common setting**.

4.5 User Comparison Based on Preferred Assistive Technologies

We investigate the differences between users that activated ATs when interacting with *i*Move (*DS*3-AT) and those who didn't (*DS*3-NAT). From Table 1 we can observe that the average number of records per user is about three times higher for DS3-AT users than for DS3-NAT users. This suggests that DS3-AT users make a more intense use of *i*Move. To confirm this, we consider, for each group, the number of notifications and actions per user, as well as the period of use of *i*Move. Here, the period of use is measured as the span of days between the first and last time a user enters the *i*Move root screen. We compare the resulting mean ranks with a Mann-Whitney U test since the data was not normally distributed (also visible in severe skewness and outliers in the boxplots of Figure 6).

We find that users in DS3-AT receive significantly more notifications ($p < 0.001$), such as the "Location" notifications, as shown in Figure 6(a)). Similarly, these users perform significantly more actions ($p < 0.001$). For example, Figure 6(b) shows that the number of times a user asks for directions to navigate to a POI is significantly higher for DS3-AT than for DS3-NAT. Users in DS3-AT also use the application for a significantly longer period than the DS3-NAT users ($p < 0.0001$): on average, 168 and 61 days respectively, as shown in Figure 6(c).
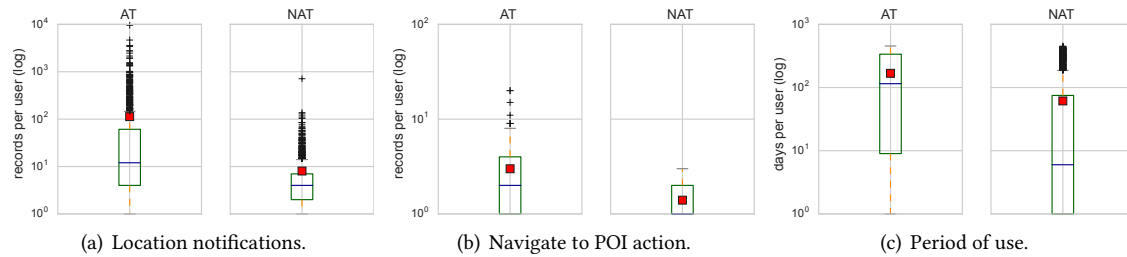
Fig. 6. Differences between users DS3-AT and SD3-NAT.

Based on the results presented above we can confirm that **users that adopt system accessibility tools make more intense use of *i*Move**. We interpret this result with two key observations. First, while we filtered out incidental users by considering only those with two or more usage sessions (see Section 3.3), some users in DS3-NAT may still have been incidental users that stopped using the app after a few sessions. Second, the users in DS3-NAT most likely have less severe visual impairment than the users in DS3-AT, which is suggested by the fact that users in DS3-NAT do not use any ATs to interact with the device. Consequently some of these users might use *i*Move only sporadically, for example only when certain light conditions make orientation more challenging for them.

5   CLUSTER ANALYSIS BASED ON USER INTERACTION STREAMS

Previous sections reveal interesting observations and findings by performing exploratory and inferential analyses independent of the temporal structure of the logs. However, richer patterns of interaction lie in the sequential relationship between the log entries. To uncover these patterns we use unsupervised learning techniques on streams of log entries, which preserve the temporal structure of the data. We anticipate that users naturally fall into clusters based on common interaction patterns with *i*Move. Moreover, the nature of these interactions is likely multi-dimensional: user clusters form a hierarchy, where most prominent interaction patterns group users in high-level clusters while less significant interaction patterns characterize subclusters. The automatic discovery of these clusters and subclusters 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 $1,517$ users residing in the DS3-AT dataset that interact with the *i*Move through assistive technologies and thus are likely to have visual impairment.

5.1   Clustering Approach

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 clickstream analysis in [48] being the closest to our work. Our approach adopts cluster analysis and builds upon previous methods for the purpose of improved understanding in assistive orientation of PVI. One of the inherent challenges in analyzing our data is the diversity of the possible interactions. 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 3.3) into our feature engineering (described below) to yield more intuitive and high level descriptions for the discovered clusters.

As described in our ASSETS paper [27], 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 and more fine-grained subclusters of similar users by graph
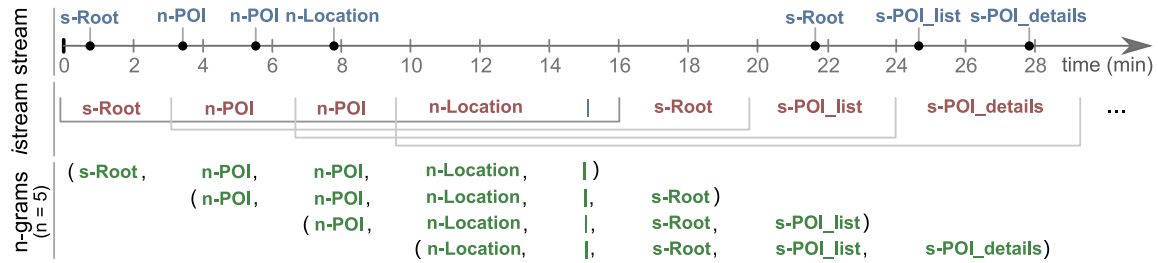
Fig. 7. Mapping interaction streams to n-grams. (s-Root: root screen, n-POI: notification about a POI, n-Location: notification about user location, s-POI_list: screen with list of POIs, and s-POI_details: screen with details about a POI.)
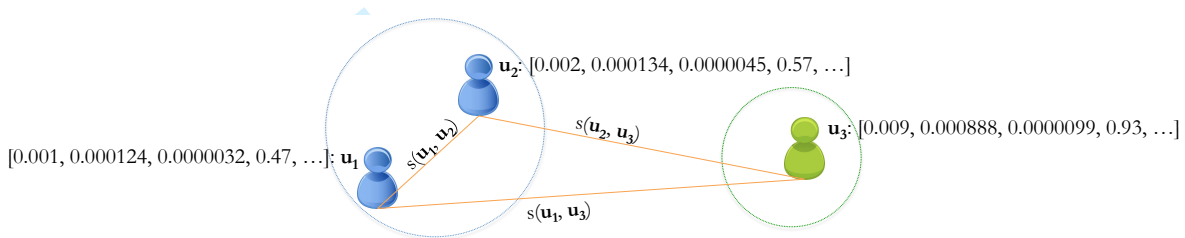

Fig. 8. Constructing a similarity graph, where nodes represent users and edges the similarity of their vectors.

partitioning. Finally, we interpret the meaning of the clusters and subclusters by isolating primary features that are responsible for their formation. To assist future researchers in adopting this analysis of their data, we describe the above steps, implementation, assumptions, and the hyper-parameters used in our approach.

Obtaining user *i*stream. We define an *i*stream as a sequence of interactions between the user and *i*Move, extracted from user's log entries ordered by their timestamp. It captures both the type of the log entry (i.e. screen, action, or notification) and the magnitude of time gaps between two consecutive log entries. Precise time gap values are omitted. If a gap is smaller than 10 minutes the log entries belong to the same session (as defined in Section 3.3). Instead time gaps greater than 10 minutes denote session boundaries, and are represented by the symbol "|". 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 [14]. 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 denoted 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. The *i*stream for a given user i is defined as a sequence $S_i = (s_1, s_2, ..., s_m)$ where m is the total length of the *i*stream for user i and, for each $j \in [1, m]$, $s_j \in A_s \cup A_a \cup A_n \cup \{|\}$. We also define F_n as the set of all possible n-grams (n consecutive elements) from all the users' *i*stream sequences: $F_n = \bigcup_{i=1}^{\#users} n\text{-gram}(S_i)$. We represent each user by a numerical k-dimension feature vector, where k is the number of all possible n-grams in F_n. To calculate these vectors we use the term frequency-inverse document frequency (tf–idf) vectorizer in [39], a typical practice to obtain features for document clustering. Simply put, we count the occurrences of n-grams in each user *i*stream and normalize and weight with diminishing importance n-grams that occur in the majority of *i*streams across all users. We experimented with different values of n in the n-gram and chose 5-grams for our analysis, though 4-grams and 3-grams revealed similar clusters.

Reducing the dimensionality of the feature space. 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. Thus, the vector space representing users is typically of high dimensionality ($k = 23,036$ possible 5-grams) and very sparse (most of the 5-grams are not present in the users' stream and thus many feature values are zeros). Data in high dimensional feature spaces exhibit poor similarity under measures such as cosine similarity, typically used for document clustering, thus may fall under clusters that are not meaningful. To overcome this challenge that was not previously addressed in our ASSETS paper [27], we use a linear dimensionality reduction method called Latent Semantic Analysis [15], which is typically performed on tf–idf features. With LSA, we project users to a lower dimensional space, by taking the list of $23,036$ unique n-grams across all users and approximate them as a linear combination of 735 unique features, while still explaining 98% of users' $i$stream variability.

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 projected in the low-dimensionality feature space. Figure 8 illustrates a toy example of this graph.

Clustering. We partition the graph into a dendrogram containing clusters and subclusters of similar users with community detection using the Louvain method [11], which, simply put, optimizes for higher density of edges inside communities compared to edges between communities. For our implementation we used the "generate_dendrogram" method with default parameters from the python-louvain library [8]. The Louvain method is a form of agglomerative hierarchical clustering [46]. Thus, it is a natural approach to uncover structure from our data, which are represented with a graph. Other approaches such as k-means [30], which fall under centroid-based clustering, would not be appropriate since they come with other geometric assumption about the data (e.g., sphericity) [25]. One of the advantages of the Louvain method is that it is highly efficient for unfolding a complete hierarchical community structure for large-scale graphs [11].

Identifying descriptive features. To interpret cluster meaning, we isolate the primary features responsible for a cluster by performing feature selection. Specifically, for each cluster, we build a binary classifier that distinguishes users belonging to that cluster from the remaining users at the same level of the dendrogram. We select 10 most important features based on their ability to discriminate between the two classes in the following way. We determine the dependency between each feature and the classifier assigned label using a chi-squared statistic [51]. If the feature is independent of the label, it is discarded. Otherwise, the feature is informative of the cluster formation. The chi-squared statistic is used to rank (score) such features and the features with top 10 scores are retained using the "SelectKBest" method from scikit-learn [39].

5.2 Results and Interpretation

Our cluster analysis requires users to have at least one session with 5 log entries that fall under actions, screens or notifications categories. From the total of $1,517$ users residing in the S3-AT dataset $1,441$ met the criterion. The clustering procedure generates 9 clusters with a modularity of 0.39, where modularity [37] 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. Six of the detected clusters contain a total of 6 outlier users, which had at most 3 short sessions each. We omit these outlier users from the following discussion, hence focusing on three clusters with many users and their subclusters. Figure 9 visualizes the resulting clusters and the top 3 features with the highest discriminating power per cluster. To get a confirmation of the semantics we associate to each cluster and subcluster based on their top 20 primary features. To further study these clusters, we analyze and compare users' interaction characteristics such as session duration and time passed between consequent sessions among others.

C1: The first cluster contains 531 users. From the primary features having higher normalized frequencies for the users in this cluster compared to users outside the cluster, we observe that C1 users actively interact with the

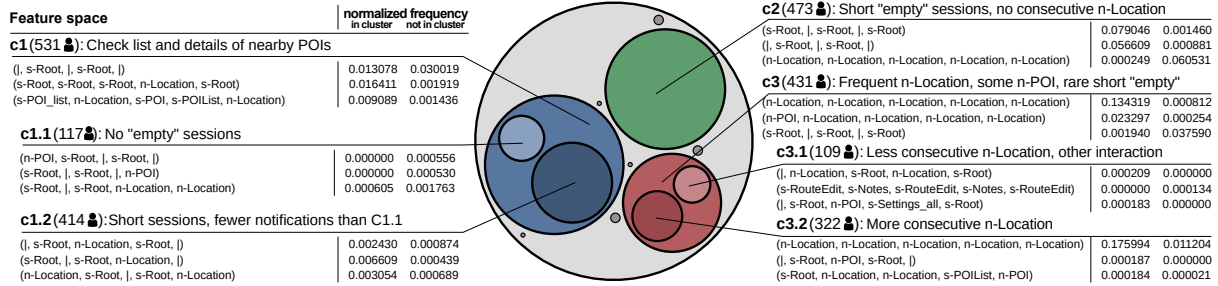| Feature space	normalized frequency	
	in cluster	not in cluster
c1(531 👤): Check list and details of nearby POIs		
(|, s-Root, |, s-Root, |)	0.013078	0.030019
(s-Root, s-Root, s-Root, n-Location, s-Root)	0.016411	0.001919
(s-POI_list, n-Location, s-POI, s-POIList, n-Location)	0.009089	0.001436
c1.1(117 👤): No "empty" sessions		
(n-POI, s-Root, |, s-Root, |)	0.000000	0.000556
(s-Root, |, s-Root, |, n-POI)	0.000000	0.000530
(s-Root, |, s-Root, n-Location, n-Location)	0.000605	0.001763
c1.2(414 👤): Short sessions, fewer notifications than C1.1		
(|, s-Root, n-Location, s-Root, |)	0.002430	0.000874
(s-Root, |, s-Root, n-Location, |)	0.006609	0.000439
(n-Location, s-Root, |, s-Root, n-Location)	0.003054	0.000689

	normalized frequency	
c2(473 👤): Short "empty" sessions, no consecutive n-Location		
(s-Root, |, s-Root, |, s-Root)	0.079046	0.001460
(|, s-Root, |, s-Root, |)	0.056609	0.000881
(n-Location, n-Location, n-Location, n-Location, n-Location)	0.000249	0.060531
c3(431 👤): Frequent n-Location, some n-POI, rare short "empty"		
(n-Location, n-Location, n-Location, n-Location, n-Location)	0.134319	0.000812
(n-POI, n-Location, n-Location, n-Location, n-Location)	0.023297	0.000254
(s-Root, |, s-Root, |, s-Root)	0.001940	0.037590
c3.1(109 👤): Less consecutive n-Location, other interaction		
(|, n-Location, s-Root, n-Location, s-Root)	0.000209	0.000000
(s-RouteEdit, s-Notes, s-RouteEdit, s-Notes, s-RouteEdit)	0.000000	0.000134
(|, s-Root, n-POI, s-Settings_all, s-Root)	0.000183	0.000000
c3.2(322 👤): More consecutive n-Location		
(n-Location, n-Location, n-Location, n-Location, n-Location)	0.175994	0.011204
(|, s-Root, n-POI, s-Root, |)	0.000187	0.000000
(s-Root, n-Location, n-Location, s-POIList, n-POI)	0.000184	0.000021

Fig. 9. Clustering results.

(a) Sessions length (# records).   (b) Sessions duration (minutes).   (c) Number of sessions.

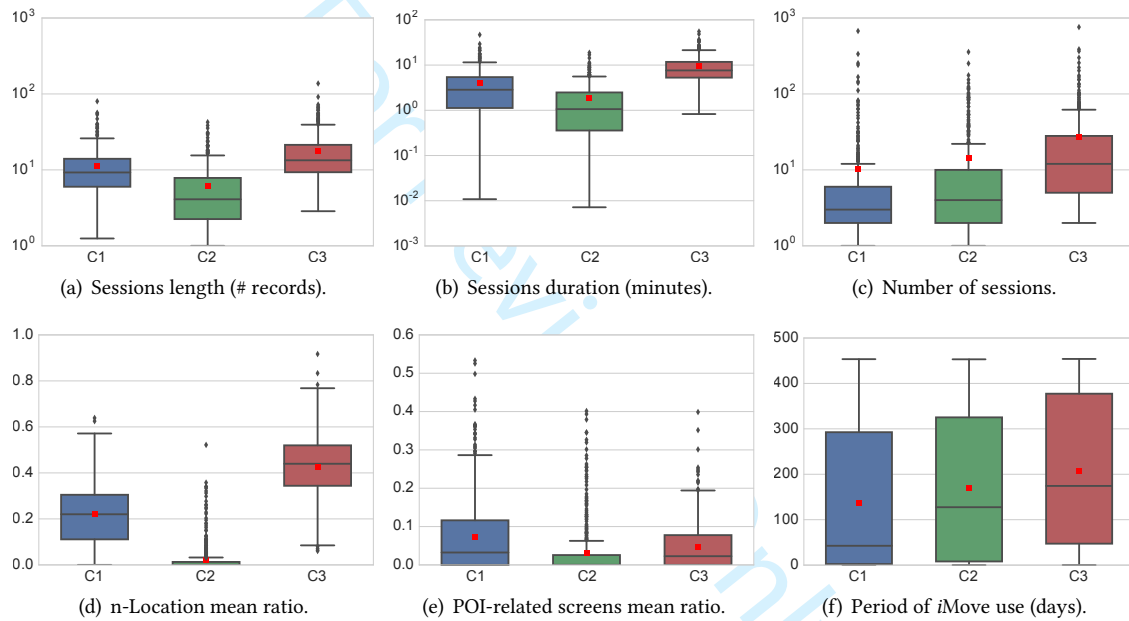(d) n-Location mean ratio.   (e) POI-related screens mean ratio.   (f) Period of *i*Move use (days).

Fig. 10. Analysis of the three high level clusters.

app. At the same time, they receive some location notifications and further inquire information about points of interest around them by visiting the *POIDetail* and *POIListAroundMe* screens. The primary features with lower normalized frequencies for this cluster compared to other clusters indicate that most of the sessions of C1 users tend not to be short "empty" sessions i.e., sessions in which the user only starts the app to visit the root screen (e.g., the n-gram "| s-Root | s-Root |"). Also, users receive notifications related to their location sparsely. We can infer that users in this cluster often open the application to check the list of nearby POIs and their details. In confirmation of these interpretations we found that users in C1 have higher frequency of sessions that explore POI-related screens than C2 and C3, shown in Fig. 10(e); they receive sparser notifications in a session about their location compared to C3 but more often than users in C2 (Fig. 10(d)); and their session lengths tend to be shorter than C3 but longer than C2 as captured by both number of logs in a session (Fig. 10(a)) and session duration in minutes (Fig. 10(b)).

(a) POI-related screens mean ratio. (b) Sessions length (# records). (c) n-Location mean ratio. (d) Period of *i*Move use (days).
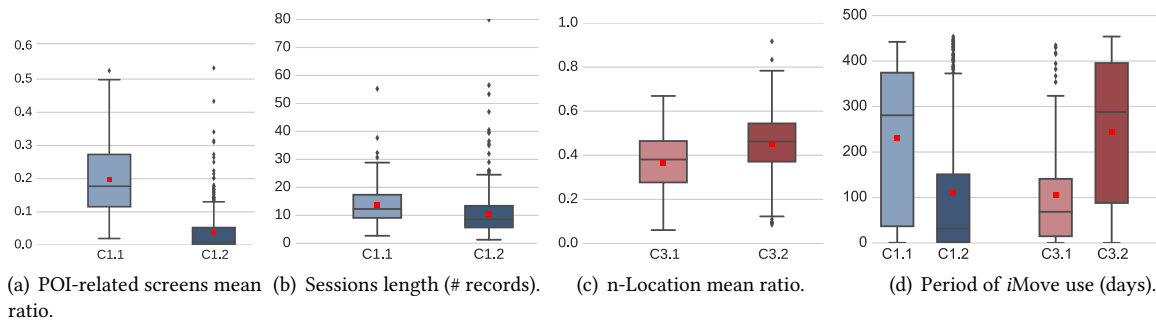
Fig. 11. Analysis of the subclusters identified in C1 and C3.

Two subclusters are generated from C1:

C1.1 contains 117 users that don't have "empty" sessions. Further analysis of their records indicate higher ratio of POI-related screens per session compared to C1.2 users (Fig.11(a)).

C1.2 contains 414 users that tend to have more often short sessions with a fewer notifications than C1.1. We suspect that C1.1 is picking users that have longer session length within C1. This is also confirmed by the boxplots of mean session length of C1.1 users compared to C1.2 in Fig.11(b).

C2: The second cluster contains 473 users. In this case most of the primary features denote higher frequencies for short "empty" sessions and lower frequency of consecutive location notifications within the same sessions for users in C2 than outside C2. This suggests that C2 contains users that starts the application, do not wait for any notifications, and then close the application. Since the Root screen displays the current address, we speculate C2 users often open *i*Move simply to access (though VoiceOver or other AT) the current address. As confirmed by the boxplots in Fig. 10(a) and Fig. 10(b), users in C2 have shorter mean session length as measured by the number of records in their sessions and durations in minutes. Typically, users in this cluster receive the lowest number of Location notifications (Fig.10(d)) and C2 users inquire less often about POI (Fig.10(e)) compared to C1 and C3. Our clustering technique did not generate any subcluster for C2.

C3: The third cluster contains 431 users. From the top features characterizing this cluster, most of them indicate high frequency of location and POI notification sequences in a single session for users in C3; and others indicate low frequency of "empty" sessions. These features suggest that C3 is a set of users running the application for long sessions during which they frequently receive location and POI notifications. Indeed, in Fig. 10(a) and 10(b) we observe a higher mean session length for C3 users compared to users in the other clusters. Also, while C1 users also receive location notifications, C3 users have a substantially higher average ratio of location notifications per session than both C1 and C2.

Two subclusters are generated from C3:

C3.1 differentiates 109 users from the C3 cluster as the ones that receive less repeated location notifications. We suspect that these users may still interact with *i*Move, e.g. visit the Root screen or Settings_all, while receiving Location notifications.

C3.2 contains the rest of the 322 users that receive more repeated location notifications, as confirmed in Fig. 11(c).

More importantly, Figure 10(c), 10(f), and 11(d) demonstrate a distinct difference between our high and low level clustering results. Specifically, at a high level our clustering approach is able to group users based on their interaction patterns independently of the number of their total sessions and total days using *i*Move. However,

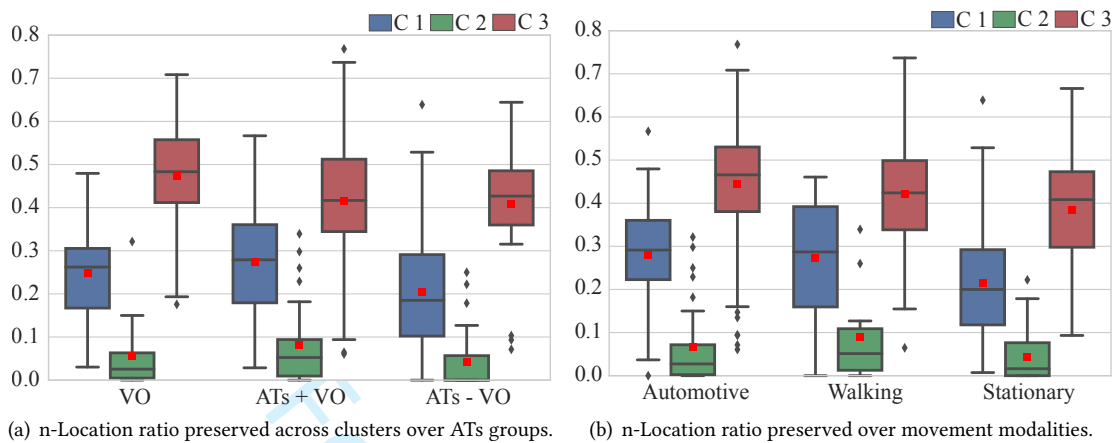(a) n-Location ratio preserved across clusters over ATs groups.

(b) n-Location ratio preserved over movement modalities.

Fig. 12. Mean ratio of location notifications is preserved both across ATs-based grouping and movement modalities.

(a) Categorization of AT groups across clusters.

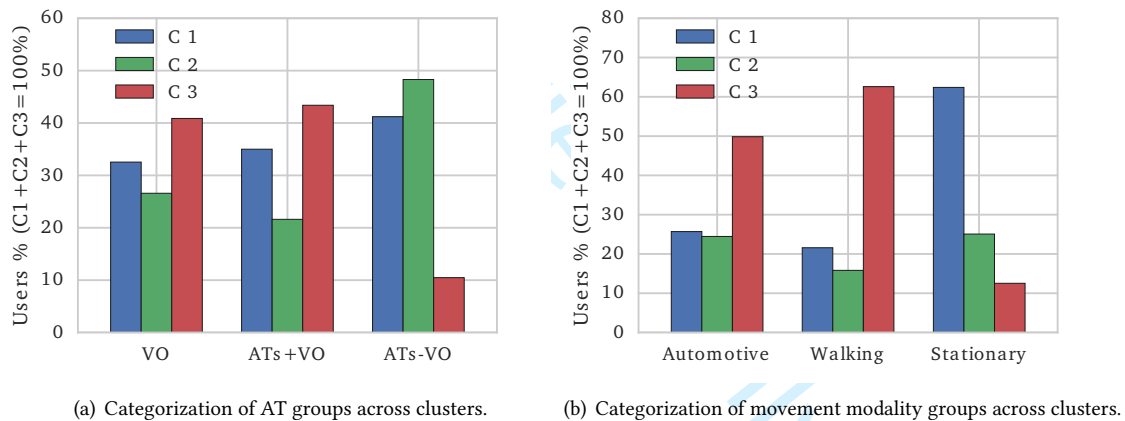(b) Categorization of movement modality groups across clusters.

Fig. 13. Distribution of users per cluster conditioning on the AT and movement modality groups.

in the lower level our clustering approach tends to uncover some of the sub-behaviors that are more common among "novice" users, users that are using *i*Move for a shorter period, within the higher level clusters.

5.3 Interaction Clustering vs Grouping Based on ATs and Movement Modality

Sections 4.3 and 4.4 define groups of users based on their assistive tools and movement modality, respectively. We explore how our clustering approach based on user interactions relate to these alternative categorizations of *i*Move users. At a high level, *i*Move users fall under three clusters: in C1 users are interested in surrounding points of interest, in C2 they interact in short bursts to inquire about current location, and in C3 users keep the app active for long sessions while in motion. As shown in Fig. 12 *the relationship between these clusters tends to be preserved across users with different severity of visual impairment and preferred movement modality.*

Interaction clustering and AT-based grouping. Figure 13(a) shows how *i*Move users adopting different ATs fall into the C1, C2, and C3 interaction clusters. We note that users who do not include VoiceOver as one of their ATs to interact with *i*Move have a less marked presence in C3. We recall that these users are most likely the ones with mild visual impairment since they are capable of always accessing *i*Move through visual means. Thus, it is not surprising that these users are less present in C3, the cluster of users who constantly receive notifications about their surroundings. In contrast, the majority of this group (almost half) is located in C2, where users tend to open the app in short bursts to investigate their surroundings. Based on these findings we suggest that *users with mild visual impairment use* i*Move sporadically and only to retrieve specific information about their surroundings*. On the other hand, users who rely on VoiceOver, either exclusively or also with other ATs, are instead more present in C3 and less in C2. These users are the ones likely to have *moderate or severe visual impairment and therefore they are often interested in receiving notifications* while using *i*Move.

Interaction clustering and movement modality. Our analysis of preferred movement modalities conducted in Section 4.3 uncovers that users prevalently access *i*Move through a single movement modality: stationary, automotive, or walking. Figure 13(b) shows how grouping users based on these preferred movement modalities relates to their interaction clusters. Specifically, we see that users that prevalently interact with the app while walking or in an automotive modality are often grouped in C3. This suggests that both *automotive and walking users are interested in continuously receiving notifications about their location and surroundings while in motion*; they rarely open the app to just inquire about their surroundings. Accessing POI information seemed to be more typical for stationary users, people who stop moving to interact with the app.

6 DISCUSSION

Our analysis of large-scale longitudinal data based on exploratory, inferential, and descriptive methods provides evidence for how PVI interact with an assistive orientation and mobility application in real-world scenarios as well as how to make similar applications more responsive to variability within this user group. Given the tendency of prior work in this field towards supervised experiments with few local participants, the release and analysis of observational data in the wild, presented in this article, can play an important role in achieving a broader impact in independent mobility that these technologies can provide.

6.1 Implications

Our analysis highlights a number of use properties that are specific to *i*Move and provide insights that can be generalized to other assistive applications, in particular those aimed at supporting orientation and mobility of PVI.

Functions and defaults. Our analysis indicates that users prefer to receive all available information, even when this results in a higher verbosity. The corresponding settings can be activated by default with the option to allow users to adjust them as they become more familiar with the application or a new environment. Users receive most of their notification while in motion, which highlights the need for notifications to be pushed to the user automatically. However, there is high variability in the preferred proximity and frequency of these automatic notifications. Hence, users should be given the opportunity to tune these parameters early on. As expected, users who make the most intense use of *i*Move are those who activate the accessibility features in their phones. What was not expected is that at least 3/4 of these users interact with the system using residual sight. This means that visual interfaces (e.g., a map), even if not accessible through a screen reader, can still be useful to many users of apps designed for PVI. Enlarged content is the second most activated accessibility features by *i*Move users beyond VoiceOver. This emphasizes the need for similar applications to guarantee full compatibility with this built-in accessibility feature beyond the screen reader.

Novel interfaces. We observe that users avoid active interactions with the system while in motion, e.g. walking. This could be due to the fact that, when in motion, users concentrate on their activity and use their senses to preserve their safety, or simply their hands are occupied with a guide dog or a white cane. We suspect that a user with visual impairment that is walking prefers to stop to interact with *i*Move. This observation calls for new interfaces that allow users to inquire for information while in motion. Such interfaces could for example minimize the cognitive load in performing these actions, incorporate gesture recognition, or employ other hands-free input techniques.

Inference and adaptation. In this analysis users are grouped across three axes: i) activated built-in accessibility features, ii) movement modality, and iii) interaction behavior as captured by clustering. We discuss both how these axes differentiate interactions with *i*Move and how they interplay. We confirm that the relationship between the uncovered user clusters tends to be preserved across users with different severity of visual impairment and preferred movement modality. Future applications can infer the group that a user belongs to across these axes and adapt its settings and interface accordingly. For example, *i*Move users have a tendency to receive notifications in predominantly one moving modality, walking or automotive. The frequency of notifications as well as the amount of information enclosed could be adapted based on the modality.

6.2 Limitations

Our analysis of large-scale observational data from remote usage logs can overcome many of the challenges that are typical of experimental studies in accessibility. Since it is not constrained to specific participants, scenarios, or laboratory environments it can reflect the diversity of real-world situations and user demographics. Moreover, it is not susceptible to the Hawthorne effect, where participants behave differently due to their awareness of being observed. However it comes with limitations.

Incomplete contextual information. Users demographics are not known and we can only infer some of them heuristically. For example, we don't know users' age and we can only infer if they are blind or low visioned based on the activated built-in accessibility features. Whereas for those users who did not activate any of these features we can not be really conclusive. We assume they are sighted though they may be users with low vision. Also, we can't estimate users' technology experience or expertise in mobility as well as other background information that may influence how they interact with *i*Move.

Due to privacy concerns, collected data must be anonymous. They cannot contain neither explicit identifiers such as full name nor quasi-identifiers such as location [9]. Hence *i*Move does not collect any location-related information, like address, nearby points of interest, or geonotes. Knowing this information could allow for deeper insights and interpretability of the observed user behavior. For instance, location information can also influence behavior; wanting to be notified of the street name may be more important in cities that have complex street layout, etc.

As future work, we are interested in examining best practices to combine observation data with a remote study. For example, users could be invited to participate and share more detailed and anonymous information about their background and their interaction with the app. Data from this smaller sample of users could contribute to validate our findings and to provide deeper interpretations.

Simplifying assumptions. The definition of a session can have an impact in the interpretability of the user clusters. In this analysis we define a session based on heuristic temporal gaps in interaction. However there may be other aspects that delimit usage sessions that we are not considering. Provided more contextual information, session boundaries could be better estimated.

Both the dataset and details of the clustering technique have evolved in this article compared to our initial analysis [27]. However, we find the clustering results to be consistent, with the exception of the two smaller clusters with burst inquiries about location and nearby point of interest in [27] converging

into one cluster with short sessions. This could be due to the longer timespan of this dataset, 15months vs.
3 months in the previous analysis. A limitation of this analysis is the fact that the dataset is a snapshot in
the lifespan of *i*Move, 15 last months. This means we are comparing users' interaction with the app over
different periods. Some users just started using the app. Others had been using it for long time before
the logging system and only their last interactions are logged. There were practical reasons behind the
decision to use all available data. However, we are continuing collecting data and in future work we
would perform our analysis only on a subset of users, e.g. users for whom a year has passed since they
installed the app. By controlling for this factor, we could also explore the evolution of user interactions
over time.

Approach portability. We see the analysis methodology proposed in this contribution to be adapted to
the study of other applications. However, the challenge in the field of accessibility is for researchers to
have access to such data. A similar approach requires a publicly accessible app, not a prototype; this
entails engineering, communication, maintenance, localization, and other components often outside the
realm of a research lab. In our case this was achievable through a collaboration between industry and
academia.

7 CONCLUSIONS

This article presents an analysis of data collected in the real world from *i*Move, a mobile app that supports the
outdoor orientation of PVI. We initially collected a dataset containing usage logs with the *i*Move system of
more than 60, 000 users. We then filtered "incidental" users, that is those users that are not really interested in
the functions of the app. Our final analysis covered the remaining 15, 000 users worldwide and more than 1.5
million log records. We found that the most popular interaction mode for these users is passive. They receive
more notifications, often verbose, while in motion and they perform fewer actions. The use of built-in assistive
features such as enlarged text indicated a high presence of users with residual sight. Moreover, we observed
users' tendency to receive notifications in predominantly one moving modality, either walking or in a moving
vehicle.

*i*Move was originally designed with a main user target in mind: PVI that would keep the app active along a
route to get notifications. By clustering about 1, 400 users with visual impairment based on common interaction
patterns, our initial user group was successfully identified from one of the major clusters (C3). It contained more
than 25% of the users. In addition, our clustering method was able to capture and provide semantics for the
remaining 75% of the VO-users with two more clusters. The first, C1, identified users that mainly use the app to
know which are the nearby point of interests. The second, C2, grouped users that employ the app in short bursts
to check their location.

One important characteristic of the clusters identified in this contribution is that they actually uncover user's
behaviour. Differently from the users' grouping based on assistive technologies and moving modality, clusters
capture user's interaction patterns, with a clear and meaningful semantic confirmed by a follow-up exploratory
analysis.

Our clustering approach also identifies subclusters for C1 and C3. Interestingly, these subclusters differentiate
between novice users and those that used *i*Move for a long period. This suggests research questions that we
intend to investigate in future work: how does a user's interaction pattern change during time? Which are the
behavioural differences between novice and expert users? Which are the preferred settings of experts users?

From the point of view of users' clustering, there are two possible directions along which this contribution can
be improved. First, it is possible to study the link between preferences for user settings and the automatically
detected user clusters. Second, clustering techniques can be possibly used for effectively identifying "incidental
users" hence removing them more reliably.

Our clustering method can contribute to the analysis of similar assistive applications. We are currently applying this approach to guide the extension of *i*Move for outdoor navigation in addition to orientation support. A second mobility assistive tool that can benefit from the work in this article is NavCog [4], an indoor navigation assistant. In both cases, we believe that knowledge extraction in this unsupervised way from automatically collected usage data will help gain a clearer understanding of this user population – hence improving the design of navigation and other assistive applications for PVI.

ACKNOWLEDGMENTS

REFERENCES

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

[2] Dragan Ahmetovic, Cristian Bernareggi, and Sergio Mascetti. 2011. Zebralocalizer: identification and localization of pedestrian crossings. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*. ACM, 275–284.

[3] Dragan Ahmetovic, Cole Gleason, Kris Kitani, Hironobu Takagi, and Chieko Asakawa. 2016. NavCog: turn-by-turn smartphone navigation assistant for people with visual impairments or blindness. In *Web for All Conference*. ACM.

[4] Dragan Ahmetovic, Cole Gleason, Chengxiong Ruan, Kris Kitani, Hironobu Takagi, and Chieko Asakawa. 2016. NavCog: A Navigational Cognitive Assistant for the Blind. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '16)*. ACM.

[5] Dragan Ahmetovic, Masayuki Murata, Cole Gleason, Erin Brady, Hironobu Takagi, Kris Kitani, and Chieko Asakawa. 2017. Achieving Practical and Accurate Indoor Navigation for People with Visual Impairments. In *Web for All Conference*. ACM.

[6] Ahmetovic, D., Bernareggi, C., Gerino, A., and Mascetti, S. 2014. ZebraRecognizer: efficient and precise localization of pedestrian crossings. In *Proceedings of the 22nd International Conference on Pattern Recognition*. IEEE.

[7] Aries Arditi, Jeffrey D Holtzman, and Stephen M Kosslyn. 1988. Mental imagery and sensory experience in congenital blindness. *Neuropsychologia* 26, 1 (1988), 1–12.

[8] Thomas Aynaud. 2017. Louvain Community Detection. https://github.com/taynaud/python-louvain. (2017). [Online; accessed 10-July-2017].

[9] Claudio Bettini, Sergio Mascetti, X Sean Wang, and Sushil Jajodia. 2007. Anonymity in location-based services: Towards a general framework. In *Mobile Data Management, 2007 International Conference on*. IEEE, 69–76.

[10] Jeffrey P Bigham, Anna C Cavender, Jeremy T Brudvik, Jacob O Wobbrock, and Richard E Ladner. 2007. WebinSitu: a comparative analysis of blind and sighted browsing behavior. In *Proceedings of the 9th international ACM SIGACCESS conference on Computers and accessibility*. ACM, 51–58.

[11] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment* 2008, 10 (2008), P10008.

[12] Matthias Böhmer, Brent Hecht, Johannes Schöning, Antonio Krüger, and Gernot Bauer. 2011. 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*. ACM, 47–56.

[13] Erin L. Brady, Daisuke Sato, Chengxiong Ruan, Hironobu Takagi, and Chieko Asakawa. 2015. 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)*. ACM, New York, NY, USA, 387–388. DOI:http://dx.doi.org/10.1145/2700648.2811383

[14] Marc Damashek. 1995. Gauging similarity with n-grams: Language-independent categorization of text. *Science* 267, 5199 (1995), 843.

[15] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science* 41, 6 (1990), 391.

[16] Himel Dev and Zhicheng Liu. 2017. Identifying Frequent User Tasks from Application Logs. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces*. ACM, 263–273.

[17] Christine Dickinson. 1998. *Low vision: principles and practice*. Butterworth-Heinemann Berwick upon Tweed, UK.

[18] Trinh Minh Tri Do, Jan Blom, and Daniel Gatica-Perez. 2011. Smartphone usage in the wild: a large-scale analysis of applications and context. In *Proceedings of the 13th international conference on multimodal interfaces*. ACM, 353–360.

[19] Navid Fallah, Ilias Apostolopoulos, Kostas Bekris, and Eelke Folmer. 2013. Indoor human navigation systems: A survey. *Interacting with Computers* 25, 1 (2013), 21–33.

[20] Nicholas A Giudice and Gordon E Legge. 2008. Blind navigation and the role of technology. *Engineering handbook of smart technology for aging, disability, and independence* (2008), 479–500.

[21] Martin Goldberg. 2015. *Assisting Visually Impaired People with Mobility through Technology in the Age of Context.* Ph.D. Dissertation. City University of New York.

[22] Joo Guerreiro, Dragan Ahmetovic, Kris M Kitani, and Chieko Asakawa. 2017. Virtual Navigation for Blind People: Building Sequential Representations of the Real-World. In *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility.* ACM.

[23] Amy Hurst, Scott E Hudson, Jennifer Mankoff, and Shari Trewin. 2013. Distinguishing users by pointing performance in laboratory and real-world tasks. *ACM Transactions on Accessible Computing (TACCESS)* 5, 2 (2013), 5.

[24] Hideyuki Iwahashi. 1983. *Toward white wave - Story of Seiichi Miyake (in Japanese).* Traffic Safety Research Center.

[25] Anil K Jain. 2010. Data clustering: 50 years beyond K-means. *Pattern recognition letters* 31, 8 (2010), 651–666.

[26] Hernisa Kacorri, Kris M Kitani, Jeffrey P Bigham, and Chieko Asakawa. 2017. People with Visual Impairment Training Personal Object Recognizers: Feasibility and Challenges. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, 5839–5849.

[27] 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 *International ACM SIGACCESS Conference on Computers and Accessibility*. ACM.

[28] Quannan Li, Yu Zheng, Xing Xie, Yukun Chen, Wenyu Liu, and Wei-Ying Ma. 2008. Mining user similarity based on location history. In *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*. ACM, 34.

[29] Richard G Long and EW Hill. 1997. Establishing and maintaining orientation for mobility. *Foundations of orientation and mobility* 1 (1997).

[30] James MacQueen and others. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, Vol. 1. Oakland, CA, USA., 281–297.

[31] Sergio Mascetti, Dragan Ahmetovic, Andrea Gerino, and Cristian Bernareggi. 2016. ZebraRecognizer: Pedestrian Crossing Recognition for People with Visual Impairment or Blindness. *Pattern Recognition* (2016).

[32] Sergio Mascetti, Dragan Ahmetovic, Andrea Gerino, Cristian Bernareggi, Mario Busso, and Alessandro Rizzi. 2016. Robust traffic lights detection on mobile devices for pedestrians with visual impairment. *Computer Vision and Image Understanding* 148 (2016), 123–135.

[33] Sergio Mascetti, Lorenzo Picinali, Andrea Gerino, Dragan Ahmetovic, and Cristian Bernareggi. 2016. Sonification of guidance data during road crossing for people with visual impairments or blindness. *International Journal of Human-Computer Studies* 85 (2016), 16–26.

[34] Florian Meier, Johannes Aigner, and David Elsweiler. 2017. Using Sessions from Clickstream Data Analysis to Uncover Different Types of Twitter Behaviour. In *Everything Changes, Everything Stays the Same? Understanding Information Spaces. Proceedings of the 15th International Symposium of Information Science (ISI 2017)-(isi2017)*. Humboldt-Universität zu Berlin.

[35] Daniel R Montello. 2001. Spatial cognition. (2001).

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

[37] Mark EJ Newman and Michelle Girvan. 2004. Finding and evaluating community structure in networks. *Physical review E* 69, 2 (2004), 026113.

[38] Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval* 2, 1-2 (2008), 1–135.

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

[40] Torben Poulsen. 1982. Acoustic traffic signal for blind pedestrians. *Applied Acoustics* 15, 5 (1982), 363–376.

[41] N Pressey. 1977. Mowat sensor. *Focus* 11, 3 (1977), 35–39.

[42] Christian Quadri, Matteo Zignani, Sabrina Gaito, and Gian Paolo Rossi. 2016. Clique-aware mobile social clouds. In *IFIP Networking Conference (IFIP Networking) and Workshops, 2016*. IEEE, 512–517.

[43] Lisa Ran, Sumi Helal, and Steve Moore. 2004. Drishti: an integrated indoor/outdoor blind navigation system and service. In *Pervasive Computing and Communications, 2004. PerCom 2004. Proceedings of the Second IEEE Annual Conference on*. IEEE, 23–30.

[44] Daniele Riboni, Claudio Bettini, Gabriele Civitarese, Zaffar Haider Janjua, and Rim Helaoui. 2016. SmartFABER: Recognizing Fine-grained Abnormal Behaviors for Early Detection of Mild Cognitive Impairment. *Artificial Intelligence in Medicine* (2016).

[45] Victor R Schinazi, Tyler Thrash, and Daniel-Robert Chebat. 2016. Spatial navigation by congenitally blind individuals. *Wiley Interdisciplinary Reviews: Cognitive Science* 7, 1 (2016), 37–58.

[46] Lei Tang and Huan Liu. 2010. Community detection and mining in social media. *Synthesis Lectures on Data Mining and Knowledge Discovery* 2, 1 (2010), 1–137.

0:24 • Kacorri et al.

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

[48] Gang Wang, Xinyi Zhang, Shiliang Tang, Haitao Zheng, and Ben Y Zhao. 2016. Unsupervised Clickstream Clustering for User Behavior Analysis. In *SIGCHI Conference on Human Factors in Computing Systems*.

[49] William R Wiener, Richard L Welsh, and Bruce B Blasch. 2010. *Foundations of orientation and mobility.* American Foundation for the Blind.

[50] Michele A Williams, Amy Hurst, and Shaun K Kane. 2013. 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.* ACM, 28.

[51] Yiming Yang and Jan O Pedersen. 1997. A comparative study on feature selection in text categorization. In *ICML*, Vol. 97. 412–420.

[52] Kumar Yelamarthi, Daniel Haas, Daniel Nielsen, and Shawn Mothersell. 2010. RFID and GPS integrated navigation system for the visually impaired. In *Circuits and Systems (MWSCAS), 2010 53rd IEEE International Midwest Symposium on.* IEEE, 1149–1152.

A *I*MOVE SCREENS, ACTIONS AND NOTIFICATIONS

We report a brief description for each screen (Table 3), action (Table 4), notification (Table 5) and parameter (Table 6).

Table 3. *i*Move screens.

Screen name	Description
Root	Main screen, shows info about current address, the closer POI and the number of other POIs close-by. Enables notes creation and access to settings.
Settings_all	Allows users to toggle the "notify me" setting and to access four setting screens: Address, POIs, Notes, and System.
POIList-around_me	Detailed list of nearby POIs. By selecting a POI the "POIDetail" screen is shown.
Credits-Info	Reports information about the developer and sponsors. Allows users to email the developer.
RoutesList	List of all routes. By selecting a specific route the "RouteDetailEdit" screen is shown.
POIDetail	Reports available details about a POI such as address, phone number, and website and allows for associated actions such as navigate to, call, and open in browser.
POISelector_active_categories	Only the POIs in the selected categories will be shown and communicated to the users.
ActiveRouteSelector	Users can activate and de-activate a route from this screen. A note is rendered only if its route is active.
Settings_location	A screen specific to location and address settings, which among other allows users to select minimum spatial and temporal distances between two consecutive location notifications.
NotesList	List of all audio and text notes; upon selecting a note it is possible to edit it.
NewAudioNote	Allows users to create a new audio note.
RouteDetailEdit	Shows the details of a route and allows users to see its associated notes and share the route.
Settings_system	A screen specific to system settings, e.g. "prevent the screen lock" toggle.
NewTextNote	Allows users to create a new text note.
Settings_POI	A screen specific to POIs settings. Allows users to select the distance at which a POI will be communicated.
Settings_notes	A screen specific to notes' settings. Allows users to select the distance at which a note is rendered.
NewRoute	Allows users to create a new route.
AssociatedRouteSelector	Allows users to associate a note to a route.
EditAudioNote	Allows users to edit an audio note.
EditTextNote	Allows users to edit a text note.

Table 4. *i*Move actions.

Action name	From screen	Description
NavigateToPOI	POIDetail	Users open the default navigation app in their system (e.g. Google Maps), with directions to the selected POI.
SavedNewSpeechNote	NewAudioNote	Users save a new audio note.
SavedNewTextNote	NewTextNote	Users save a new text note.
ConvertSpeechToText(newNote)	NewAudioNote	Users convert a new audio note to a text note.
OpenWebsiteOfPOI	POIDetail	Users open a selected POI webpage in their browser.
ConvertTextToSpeech(NewNote)	NewTextNote	Users convert a new text note to an audio note.
CallPOI	POIDetail	Users place a call to a selected POI.
SharedLocationSMS	Root	Users share their position via a text message (SMS).
SharedTextNotesViaMail	RouteDetailEdit	Users share their text notes of a route by email in the form of a KML file.
SavedEditedSpeechNote	EditAudioNote	Users edit an audio note and save it.
SavedEditedTextNote	EditTextNote	Users edit a text note and save it.
ConvertTextToSpeech(EditNote)	EditTextNote	Users convert a text note to an audio note.
ConvertSpeechToText(EditNote)	EditAudioNote	Users convert an audio note to a text note.
SavedEditedRoute	RouteDetailEdit	Users edit a route and save their changes.
ImportedTextNotesKML	RouteDetailEdit	Users import text notes from a KML file.

Table 5. *i*Move notifications.

Notification name	Description
Location	Reads information on current address such as city, speed, orientation and other, based on verbosity settings. Users are notified of their current address when both the user-defined time and distance thresholds between two current-address notifications are met.
POI	Reads the closest POI to users as well as the number of other nearby POIs. The distance at which a POI is considered as nearby is defined by the user. POIs are announced if they meet the closeness threshold and if a user-defined time threshold from previous POI notification has passed.
SpeechNote	Plays an audio note when users are close to the location where the audio note was recorded. Closeness is defined by the user and an audio note is played only if it meets this user-defined closeness threshold. The note is not played back again until a certain time (user-defined) has passed.
TextNote	Analogous to audio note.

Table 6. *i*Move parameters from build 34.

Parameter name	Values	Default Value	description
SaySpeed	Yes/No	Yes	Defines whether user speed should be provided as part of a location notifications.
SayHeading	Yes/No	Yes	Defines whether user heading should be provided as part of a location notifications.
SayCourse	Yes/No	Yes	Defines whether user course should be provided as part of a location notifications.
SayCity	Yes/No	Yes	Defines whether current city should always be provided as part of a location notifications (if not, it is only provided when it changes from last location notification).
PreventIdle	Yes/No	Yes	If set to "Yes", prevents the device from going into idle mode when *i*Move is running.
BkgTimeLimit	Yes/No	Yes	If set to "Yes", *i*Move stops running in background after 30 minutes.
AutoWhereAmI	Yes/No	Yes	Enables location notifications.
AutoGeoNotes	Yes/No	Yes	Enables SpeechNote and TextNote notifications.
AutoAroundMe	Yes/No	Yes	Enables POI notifications.
GeoNoteTemporal	10s, 30s, 60s, 120s, 300s, 600s	120s	Minimum temporal distance between two SpeechNote or TextNote notifications for the same note.
LocationTemporal	10s, 30s, 60s, 120s, 300s, 600s	30s	Minimum temporal distance between two Location notifications.
PoiTemporal	10s, 30s, 60s, 120s, 300s, 600s	120s	Minimum temporal distance between two POI notifications.
LocationSpatial	30m, 50m, 100m, 500m, 1, 000m	30m	Minimum spatial distance between two location notifications.
PoiProximity	30m, 50m, 100m, 500m, 1, 000m	30m	Spatial distance at which a POI is considered as close-by.
GeoNoteProximity	30m, 50m, 100m, 500m, 1, 000m	30m	Spatial distance at which a note is considered as close-by.