Department of Computer Science
Series of Publications A
Report A-2022-5

# Mining Behavioral Patterns from Mobile Big Data

Tong Li

*Doctoral thesis, to be presented for public examination with the permission of the Faculty of Science of the University of Helsinki, in Room E204, Physicum, on the 20th of May, 2022 at 12 'o clock.*

University of Helsinki
Finland

**Supervisors**
Sasu Tarkoma, University of Helsinki, Finland
Pan Hui, University of Helsinki, Finland

**Pre-examiners**
Sarunas Girdzijauskas, Royal Institute of Technology (KTH), Sweden
Mario Di Francesco, Aalto University, Finland

**Opponent**
Kostas Stefanidis, Tampere University, Finland

**Custos**
Sasu Tarkoma, University of Helsinki, Finland

**Contact information**

Department of Computer Science
P.O. Box 68 (Pietari Kalmin katu 5)
FI-00014 University of Helsinki
Finland

Email address: info@cs.helsinki.fi
URL: http://cs.helsinki.fi/
Telephone: +358 2941 911


ISSN 1238-8645 (print)
ISSN 2814-4031 (online)
ISBN 978-951-51-8172-5 (paperback)
ISBN 978-951-51-8173-2 (PDF)
Helsinki 2022
Unigrafia

**Mining Behavioral Patterns from Mobile Big Data**

Tong Li

Department of Computer Science
P.O. Box 68, FI-00014 University of Helsinki, Finland
tong.li@helsinki.fi

PhD Thesis, Series of Publications A, Report A-2022-5
Helsinki, May 2022, 80+62 pages
ISSN 1238-8645 (print)
ISSN 2814-4031 (online)
ISBN 978-951-51-8172-5 (paperback)
ISBN 978-951-51-8173-2 (PDF)

**Abstract**

Mobile devices connected to the Internet are a ubiquitous platform that can easily record a large amount of data describing human behavior. Specifically, the data collected from mobile devices — referred to as mobile big data reveal important social and economic information. Therefore, analyzing mobile big data is valuable for several stakeholders, ranging from smartphone manufacturers to network operators and app developers.

This thesis aims to discover and understand behavioral patterns from mobile big data based on large real-world datasets. Specifically, this thesis reveals patterns from three domains: people, time, and location. First, we explore mobile big data from the people domain and propose a framework to discover users' daily activity patterns from their mobile app usage. By applying the framework to a real-world dataset consisting of 653,092 users, we successfully extract five common patterns among millions of people, including commuting, pervasive socializing, nightly entertainment, afternoon reading, and nightly socializing. Second, still from the people domain, we derive group health conditions by using their smartphone usage data. In particular, we collect mobile usage records of 452 users in North America. We then demonstrate the potential for inferring group health conditions (i.e., COVID-19 outbreak stages) by leveraging less privacy-sensitive smartphone data, including CPU usage, memory usage, and network connections. Third, we mine the behavior patterns from the time domain. We reveal the evolution of mobile

app usage by conducting a longitudinal study on 1,465 users from 2012 to 2017. The results show that users' app usage significantly changes over time. However, the evolution in app-category usage and individual app usage are different in terms of popularity distribution, usage diversity, and correlations. Last, with respect to the location domain, we leverage city-scale spatiotemporal mobile app usage data to reveal urban land usage patterns. We prove the strong correlation between mobile usage behavior and location features, which brings a new angle to urban analytics.

**Computing Reviews (2012) Categories and Subject Descriptors:**

Human-centered computing → Ubiquitous and mobile computing → Empirical studies in ubiquitous and mobile computing
Information systems → Information systems applications → Data mining
Social and professional topics → User characteristics

**General Terms:**

Ph.D. thesis, mobile computing, smartphone usage, data mining, pattern recognition

**Additional Key Words and Phrases:**

Clustering, representation learning, user behavior

# Acknowledgements

I am sincerely grateful to my main supervisor, Professor Sasu Tarkoma, for his enlightening instruction and fruitful discussion during my research. His broad knowledge and distinguished insights in data mining and mobile computing have been really beneficial to me.

I would like to thank my another supervisor, Professor Pan Hui, for the excellent directions and support through the ups and downs of my four-year Ph.D. study. His patience, understanding, and encouragement have helped me to overcome difficulties. Without his unwavering support, I would not have been able to get to this stage.

I would also like to thank my fellows in UbiCampus: Dianlei Xu, Yuxing Chen, Qingli Guo, Qingsong Guo, Gongsheng Yuan, Pengyuan Zhou, Xiang Su, Xinyang Li, Chen He, Eemil Lagerspetz, Mohammad Hoque, and many others. They made my Ph.D. experience more enjoyable and colorful.

I thank my thesis pre-examiners, Professor Mario Di Francesco and Professor Sarunas Girdzijauskas, for their kind support and insightful comments on improving this work. I would also like to thank the Doctoral Programme in Computer Science (DoCS), Academy of Finland, and Nokia for their generously financial support to my research work. Special thanks to Research Coordinator Pirjo Moen and all other colleagues from the HR team and IT team, who have made my work as convenient as possible.

I would like to thank my beloved one, Xiang Gu, for her encouragement and patience. Most importantly, I thank my parents, Jianwen Li and Hong Wang, for their unwavering love, support, encouragement, and belief in me.

Beijing, May 2022
Tong Li

# Contents

# List of Publications

This thesis is based on the following original publications, referred to as Papers I–IV in the text, and attached to the end of this thesis.

I. Tong Li, Yong Li, Mohammad Ashraful Hoque, Tong Xia, Sasu Tarkoma, and Pan Hui. To What Extent We Repeat Ourselves? Discovering Daily Activity Patterns Across Mobile App Usage. *IEEE Transactions on Mobile Computing*, 21(4):1492–1507, 2022.

   The author of this thesis was in the lead of the publication, implementing data preprocessing, probabilistic topic model, hierarchical clustering algorithm, analyzing the results, and writing the paper.

II. Tong Li, Yong Li, Mohammad Ashraful Hoque, Tong Xia, Sasu Tarkoma, and Pan Hui. To What Extent We Repeat Ourselves? Discovering Daily Activity Patterns Across Mobile App Usage. *IEEE Transactions on Mobile Computing*, 21(4):1492–1507, 2022.

   The author of this thesis was in the lead of the publication, implementing data preprocessing, probabilistic topic model, hierarchical clustering algorithm, analyzing the results, and writing the paper.

III. Tong Li, Mingyang Zhang, Yong Li, Eemil Lagerspetz, Sasu Tarkoma, and Pan Hui. The Impact of Covid-19 on Smartphone Usage. *IEEE Internet of Things Journal*, 8(23):16723–16733, 2021.

   The author of this thesis led the publication, who delivered the main ideas, proposed the methodology, designed the solution algorithms, implemented the data preprocessing and statistical analysis, and wrote the paper.

IV. Tong Li, Mingyang Zhang, Hancheng Cao, Yong Li, Sasu Tarkoma, and Pan Hui. "What Apps Did You Use?": Understanding the Long-term Evolution of Mobile App Usage. In Yennun Huang, Irwin King, Tie-Yan Liu, and

Maarten van Steen, editors, *Proceedings of The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, pages 66–76. ACM / IW3C2, 2020.

The author of this thesis led the publication, who implemented data preprocessing, statistical analysis, analyzing results to discover longitudinal evolution patterns, and writing the paper.

V. Tong Li, Zhaoqi Yang, Yong Li, Benjamin Finley, Sasu Tarkoma, and Pan Hui. Revealing Urban Dynamic Functions with Mobile App Usage Behavior and POIs. *Submitted to IEEE Transactions on Mobile Computing*, 2021.

The author of this thesis led the publication, who delivered the main ideas, designed and implemented the solution algorithms, analyzed the results, and wrote the paper.

Besides the above papers which contributed to this thesis, the author of this thesis has also participated in the following papers:

VI. Tong Li, Tristan Braud, Yong Li, and Pan Hui. Lifecycle-Aware Online Video Caching. *IEEE Transactions on Mobile Computing*, 20(8):2624–2636, 2021.

The author of this thesis participated in designing the solution, implemented the proposed algorithms, conducted experiments, and wrote the paper.

VII. Mingyang Zhang, Tong Li, Yue Yu, Yong Li, Pan Hui, and Yu Zheng. Urban Anomaly Analytics: Description, Detection and Prediction. *IEEE Transactions on Big Data*, pages 1–1, 2020.

This publication is a survey of urban anomaly analytic. The author of this thesis gathered and wrote the paragraph on urban data and anomaly detection.

VIII. Mingyang Zhang, Tong Li, Yong Li, and Pan Hui. Multi-View Joint Graph Representation Learning for Urban Region Embedding. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 4431–4437. ijcai.org, 2020.

The author of this thesis participated in designing the graph representation learning method, provided the baselines, analyzed the experiment results, and wrote a significant part of the paper.

IX. Tong Li, Ahmad Alhilal, Anlan Zhang, Mohammad Ashraful Hoque, Dimitris Chatzopoulos, Zhu Xiao, Yong Li, and Pan Hui. Driving Big Data: A First Look at Driving Behavior via a Large-Scale Private Car Dataset. In *Proceedings of 35th IEEE International Conference on Data Engineering Workshops, ICDE Workshops 2019, Macao, China, April 8-12, 2019*, pages 61–68. IEEE, 2019.

The author of this thesis participated in the design, implementation, testing, analysis of the work and wrote the paper.

X. Tong Li, Sylvia T. Kouyoumdjieva, Gunnar Karlsson, and Pan Hui. Data Collection and Node Counting by Opportunistic Communication. In *Proceedings of 2019 IFIP Networking Conference, Networking 2019, Warsaw, Poland, May 20-22, 2019*, pages 1–9. IEEE, 2019.

The author of this thesis participated in the design, implementation, testing, analysis of the work and wrote the paper.

XI. Mingyang Zhang, Tong Li, Hongzhi Shi, Yong Li, and Pan Hui. A Decomposition Approach for Urban Anomaly Detection Across Spatiotemporal Data. In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 6043–6049. ijcai.org, 2019.

The author of this thesis participated in designing the anomaly detection algorithm, analyzed the experimental results and wrote a significant part of the paper.

# Chapter 1

# Introduction

Mobile devices connected to the Internet are a ubiquitous platform that can easily record a large amount of data describing human behavior. Supported by plenty of mobile applications, just termed as apps for conciseness, mobile devices enable people to access diverse Internet services to support work-, social-life, education, and entertainment. At the same time, mobile users leave abundant fine-grained usage traces that can be collected by app developers and service providers through monitoring apps and mobile networks. These data referred to as mobile big data [11, 27, 68] have formed a cross-domain and multi-view data ecosystem, including various mobile usage behavior. For example, downloading, installing, launching, uninstalling mobile apps, CPU and memory usage of smartphones, and contextual information such as time, location, traffic, energy consumption.

## 1.1 Motivation

Mobile big data provides a new lens to discover and understand behavioral patterns of users, which has significant implications for several stakeholders. For example, smartphone manufacturers can optimize the scheduling of various smartphone resources, such as CPU, memory, and battery power, according to behavioral patterns to improve device performance and extend usage time [10, 45]. Network operators and market intermediaries can provide personalized services, through accurate recommendations and targeted advertisements to mobile users by inferring their preferences and interests from their usage behavior. By doing so, operators and intermediaries can not only enhance users' quality of experience (QoE) but also make more profits [40, 73]. Mobile app developers can upgrade

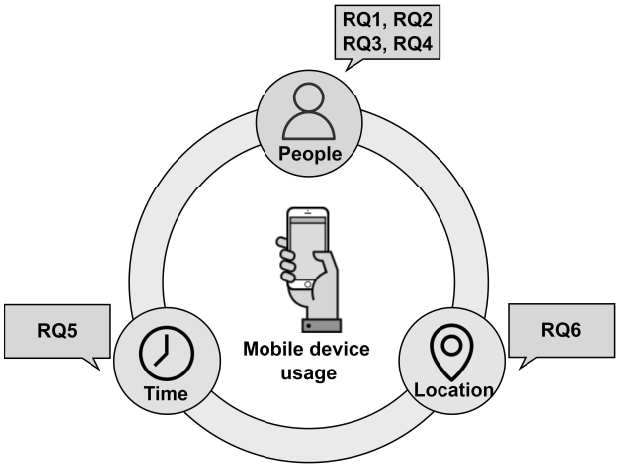



Figure 1.1: Mobile usage behavior acts as the core part linking the other domains, i.e., people, time, and location.

of existing apps and the design of new apps by analyzing behavioral patterns of mobile app usage and profiling mobile app popularity [23, 24]. Moreover, the government can understand people's living status [34] and detect group health conditions [56] from users' behavioral patterns, then make policies to improve people's well-being.

## 1.2 Problem Statement

Mobile big data record the mobile usage behavior from the domains in the 4W framework: what (usage), who (people), when (time), and where (location). In particular, mobile usage acts as the core part linking the other domains (Figure 1.1). In turn, mobile usage behavior is deeply shaped by different people, times, and locations. The complex interdependencies in different domains entail diverse behavioral patterns. This thesis aims to extract knowledge from mobile big data through a systematic study, which reveals behavioral patterns from three different domains: people, time, and location. Specifically, it focuses on answering the following research questions:

RQ1. What activities can be discovered from anonymized mobile app usage data?

RQ2. What common patterns do we share with others in our daily activities?

RQ3. Does the outbreak of COVID-19 affect users' smartphone usage? If so, how?

RQ4. Can we use smartphone usage data to infer the outbreak stages of COVID-19?

RQ5. What is the evolution of mobile app usage behavior over time?

RQ6. Is it possible to leverage mobile app usage data to reveal urban land usage patterns?

The first four research questions fall in the people domain and aim to understand user features by mining digital spatial and temporal behavioral patterns. Specifically, RQ1 aims to build connections between user activities and anonymized mobile app usage traces. RQ2 aims to discover common activity patterns among individuals on a large scale. By jointly answering RQ1 and RQ2, we build a framework that reveals users' daily activity patterns from their mobile app usage traces, both individually and collectively. While RQ1 and RQ2 focus on regular patterns, RQ3 and RQ4 aim to examine how an extreme event i.e., COVID-19 pandemic affects mobile usage behavior and explore mobile big data in sensing group health conditions i.e., the outbreak stages of COVID-19. RQ5 relates to the time domain, and aims to understand longitudinal temporal patterns of mobile app usage behavior. RQ6 addresses the location domain and explores the value of spatial features in mobile big data. This thesis presents several works to answer the above research questions with the contributions outlined in the following section.

## 1.3 Thesis Contribution

Figure 1.2 shows how the research questions are covered in the relevant publications and an overview of the research methodology used in this thesis. Paper I answers RQ1 and RQ2 by providing a framework to identify user activities from their mobile app usage data and recognize the common patterns across diverse groups of individuals. Paper II targets RQ3 and RQ4 by studying the differences in smartphone usage across the outbreak of COVID-19 and proposing an inference model to infer outbreak stages from smartphone usage data. Paper III answers RQ5 by studying how mobile app usage changes over time. Finally, Paper IV addresses RQ6 by proposing a graph-based representation learning framework that reveals dynamic regional functions using mobile app usage behavior. As

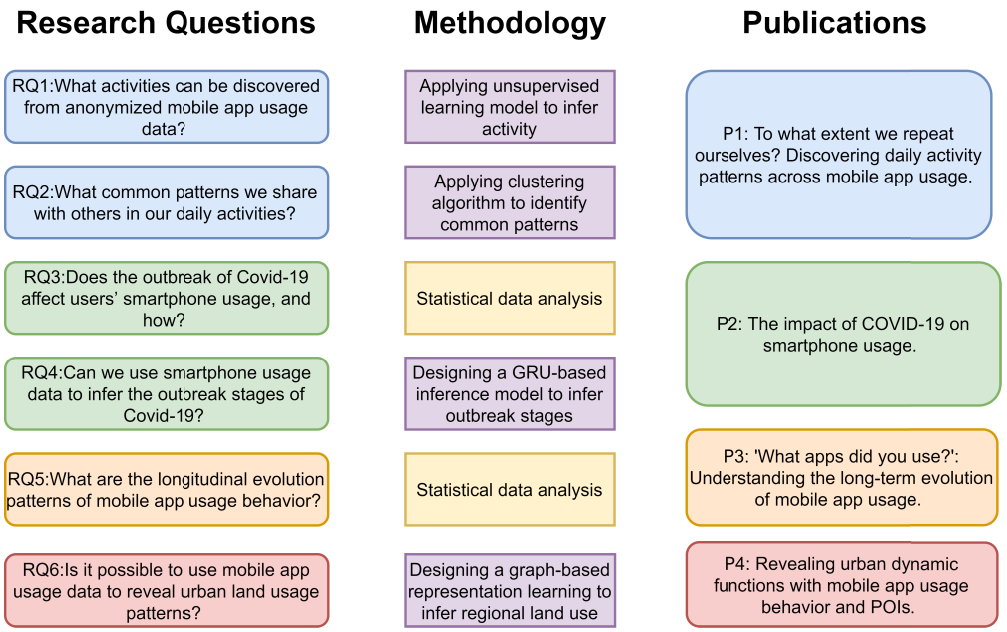



Figure 1.2: The methodology of this thesis and matching research questions along with corresponding publications.

for the methodology, statistical data analysis and machine learning algorithms are two critical methods we used to reveal behavioral patterns and understand interdependencies in mobile usage data. The behavioral patterns discovered in our work can provide prior knowledge to better protect the privacy of users. We next summarize the content of Papers I-IV.

**Paper I:** To What Extent We Repeat Ourselves? Discovering Daily Activity Patterns Across Mobile App Usage.

This paper addresses the problem of discovering the daily activity patterns of a large population through their mobile app usage data. We first segment mobile app usage traces into short time windows and then apply an unsupervised learning model, a probabilistic topic model, to infer users' activities in each time window. We next investigate the coherence of users' activity sequences to identify daily patterns for individuals. Furthermore, we employ hierarchical cluster analysis to identify the common patterns across individuals. We apply our framework on a large-scale and real-world dataset consisting of 653,092 users. We identify seven typical activities and discover the population follows five common patterns in

their daily activities. Chapter 3 of this thesis summarizes the proposed framework and critical findings.

**Paper II:** The Impact of COVID-19 on Smartphone Usage.

This paper studies the impact of COVID-19 on smartphone usage and explores the values of mobile data for sensing group health conditions. Based on a global data collection platform called Carat, we collect mobile usage records from 452 users in North America from November 2019 to April 2020. By conducting a statistical data analysis, we find that COVID-19 makes a drop in engagement with smartphones and network type switches but a rise in WiFi usage. Also, the outbreak causes new typical diurnal patterns of both memory usage and WiFi usage. Additionally, we design an inference model to infer outbreak stages from users' smartphone usage behavior, where both Macro-F1 and Micro-F1 can achieve values over 0.8. We introduce the main findings and proposed method in Chapter 4.

**Paper III:** "What Apps Did You Use?": Understanding the Long-term Evolution of Mobile App Usage.

This paper reveals long-term temporal patterns of mobile app usage behavior through statistical data analysis. We leverage the Carat dataset and select 1,465 long term users and their mobile usage records from 2012 to 2017. We then study the evolution process on a macro-level(i.e., app categories), and micro-level (individual apps). On both levels, a growth stage can be triggered by the release of new technologies. Then a plateau stage emerges due to high correlations between app categories and a Pareto effect in individual app usage. Additionally, individual app usage has an elimination stage due to fierce intra-category competition. The usage diversity in two levels exhibits opposing trends: app-category usage assimilates while individual app usage diversifies. The main findings are presented in Chapter 5.

**Paper VI:** Revealing Urban Dynamic Functions with Mobile App Usage Behavior and POIs.

The paper proposes a graph-based representation learning framework that reveals dynamic regional functions using mobile app usage behavior. Specifically, we use a graph structure to model mobile app usage data. In such a graph, nodes represent users, apps, and time-enhanced locations, and edges represent the co-occurrence of entities in mobile app usage records. The proposed framework is able to map time-enhanced location nodes into the same latent space by leveraging meta-paths and graph neural networks. As a result, a region at a specific time interval is represented by an embedding vector. We further use the learned region dynamic embeddings for the two tasks of static land usage identification

and regional economic level (GDP) prediction. We further discuss the details in Chapter 6.

## 1.4 Thesis Structure

The rest of the thesis is organized as follows. Chapter 2 introduces and compares different data collection methods and previous studies. Also, it presents two mobile usage datasets we used in our studies. Next, Chapter 3 discusses the framework to discover the daily activity patterns from users' mobile app usage data on a large scale. Chapter 4 explains how the outbreak of COVID-19 impacts smartphone usage behavior. After that, Chapter 5 presents the evolution process of mobile app usage behavior over a long-term period from 2012 to 2017. Chapter 6 introduces a representation learning framework learning dynamic location embeddings from spatiotemporal mobile app usage data. Finally, chapter 7 concludes the thesis with a summary and a discussion of future work.

# Chapter 2

# Background and Dataset Overview

This chapter introduces and compares different data collection methods and previous studies in the domain of mobile usage pattern discovery, user profiling from mobile data, and mobile app evolution analysis. Also, this chapter presents two mobile usage datasets we used in the following studies.

## 2.1 Background

This section introduces mobile usage data collection methods. We also summarize related studies in the fields of mobile usage pattern discovery, user profiling, and mobile app evolution analysis.

### 2.1.1 Data Collection Methods

Real-world data are the basis and core parts of mobile big data analysis. We first introduce different data collection methods for mobile data.

#### Monitoring Apps

One straightforward data collection method is to use monitoring apps installed in participants' mobile devices to record fine-grained mobile usage behavior automatically. In practice, researchers can deploy such a data collection method on both a small scale by recruiting volunteers [25] and a large scale by publishing monitoring apps on app stores [45]. Recruiting volunteers can focus on a particular group of users, e.g., students [60] and older adults [21]. Also, recruiting volunteers can pre-control the quality of data and mitigate the bias of analytical results by

cautiously selecting involved users according to their backgrounds and properties. Alternatively, although publishing to app stores cannot pre-determine engaged users, it can still control data quality by filtering out noisy data and alleviate the bias by leveraging the benefit from the large volume of both active users and collected data [8]. Moreover, benefited from globalization, publishing monitoring apps to international app stores like Google Play and Apple Store is more accessible to collect app usage data from multiple countries. This will enhance the generality and representativeness of the analysis results.

Monitoring apps apply the event-triggered collection scheme, i.e., collecting a smartphone usage record when an event happens. The event can be user actions [3] (e.g., screen-on, launching apps, and typing), message received [19] (e.g., notifications, emails), network requests [53], and hardware status changes [76] (e.g., CPU usage, battery levels). Researchers are feasible to collect different usage behavior and control the granularity of data collection by properly selecting trigger events. Also, mobile devices have embedded with many sensors [69], e.g., accelerometer, gyroscope, and GPS. Hence, monitoring apps can collect sufficient sensor data, such as CPU usage, movement status, GPS location, battery status. These sensor data can provide sensor contextual information for usage analyses.

### Network Operators

Nowadays, most mobile services are supported through the Internet. Therefore, apart from collecting mobile data directly from end devices, one alternative method is to collect app usage data from network operators. Network data can be collected and extracted from multiple network interfaces, including packet-switched core network (IU-Ps), serving gateway interface (SGi), mobility management interface, etc. Precisely, mobile app usage information, covered in traffic flow records and collected from SGi and Gi network interfaces, is typically inferred through deep packet inspection and deep flow inspection [64].

Due to the constraint on access to network interfaces, such a data collection method is usually performed by network operators and in large-scale measurements. The data collected generally cover most mobile users in an entire city [65] or a country [63]. Also, as the volume of network traffic data is extensive, network operators usually take sample strategies by collecting network traffic records at systematical time intervals, such as sampling every hour [74] or several minutes [26]. Apart from time information, the datasets collected by network operators usually have location information of smartphone usage records, which is commonly approximated to the GPS location of the associated base station.

We next make a comparison of the above two data collection methods. In terms of collection scale, both approaches can conduct large-scale measurements, covering tens of thousands of users, which is benefited from the development of communication and network technologies. However, as for monitoring apps, limited by response rates or app popularity, they usually cover up to hundreds of thousands of users, making it difficult to collect millions of users. In another line, monitoring apps can collect small-scale datasets by recruiting volunteers. Also, it is available to conduct control studies by properly selecting participants, which is impracticable for network operators. Network operators are also limited to associating certain types of behavior, i.e., network access. Nevertheless, monitoring apps, installed on smartphones and based on the event-triggered collection, are available to collect most kinds of mobile usage behavior by choosing different trigger events.

### 2.1.2 Mobile Usage Pattern Discovery

Discovering mobile usage patterns aims at identifying regularities in usage data to explore typical users' usage habits and further improve user experience quality. Existing studies principally worked on two sub-fields of pattern discovery, i.e., contextual pattern discovery and temporal pattern discovery.

The goal of contextual pattern discovery is to investigate the relationship between mobile usage behavior and contextual factors, including location, time, WiFi or cell connectivity, social environments, etc. For example, Do et al. [15] and Bohmer et al. [5] found out that users prefer to use web and multimedia apps while waiting for and during trips. Graells-Garrido et al. [22] analyzed city-scale app usage data and found that street types also affect mobile usage. For example, message apps consume more traffic in main streets, while dating apps are used more in pedestrian streets. Further, Shema et al. [52] proposed to determine environment context, e.g., home, workplace, commute, based on users' mobile usage behavior, where the accuracy achieved around 69%. Our work in [35] explored the correlations between mobile app usage and location features. We proposed a graph-based representation learning framework that reveals dynamic regional functions by leveraging spatiotemporal mobile app usage behavior.

Unlike contextual patterns focusing on static analysis, temporal pattern discovery explores the dynamics of mobile usage behavior. For example, the diurnal pattern is a basic temporal pattern of user behavior discovered by numerous existing studies [51]. That is, the intensity of mobile usage increases during the daytime while reduces over the night period. Also, Ghahramani et al. [20] showed

that different mobile usage behavior has different temporal patterns. For instance, the diurnal usage pattern of transport apps has more than two peaks on weekends, different from other app categories. Moreover, Jones et al. [28] analyzed users' re-visitation behavior across mobile app sessions and identified three distinct patterns, i.e., checkers, waiters, and responsives. Our work in Paper I analyzed a real-world mobile app usage dataset covering millions of users and discovered five daily activity temporal patterns based on their mobile usage traces. The five patterns include commuting, pervasive socializing, afternoon reading, nightly entertainment, and nightly socializing. Our work in Paper II investigated how the outbreak of Covid-19 affects users' temporal patterns of smartphone usage.

### 2.1.3 User Profiling from Mobile Data

Since different user profiles can lead to differences in mobile app usage behavior, many studies have sought to study the relationship between user personality traits and their app usage traces. For example, Zhao et al. [74] conducted a user-group level analysis. They analyzed one month of app usage from 106,762 users and discovered 382 distinct groups of users. They then gave a meaningful label to each user group, such as night communicators, evening learners, and financial users. Andone et al. [2] presented a descriptive analysis of how age and gender affect app usage. They discovered that females spent more time on communication and social apps, while males spent more time playing games. Also, teenagers, from 12 to 17 years old, have the highest usage time on communication, social, media, and game apps, over 40 minutes daily. Nevertheless, users over 30 years old only take less than 10 minutes on these apps. Meanwhile, Peltonen et al. [46] investigated how cultural affiliations of mobile users affect their usage behavior. By taking app category usage as features and applying the hierarchical clustering algorithm, they obtained three main clusters of cultural affiliations with respective usage patterns, i.e., European group, English speaking group, and mixed group. Our work in Paper VI also demonstrated that demographics have an important impact on users' daily activity patterns inferred from their mobile app usage traces.

Alternatively, some work has conducted predictive analyses, i.e., inferring user profiles by extracting features from mobile app usage data. For example, Seneviratne et al. [50] collected mobile app usage data from 200 users and exploited SVM to predict users' gender labels based on the apps used by users. Further, Malmi et al. [39] applied a larger dataset covering 3,760 users to verify Seneviratne's studies and predict new demographics, e.g., income and race. They determined that the most predictable trait is gender, while the hardest to predict

is income. Zhao et al. [75] extracted designed topic features from app descriptions and then used SVM and MLP to infer the gender of users.

### 2.1.4 Mobile App Evolution Analysis

Some scholars worked on analyzing mobile app evolution [6,7,55,57]. For example, Carbunar et al. [7] crawled app information from Google Play. The dataset includes 160,000 apps and lasts for six months. They then investigated how app properties, like downloads, price, and update frequency, changed over time. Calciati et al. [6] looked into the permission requests of apps. They tracked 227 Android apps and found that apps tend to require more permissions. Also, Taylor et al. [55] discovered similar evolutionary trend in permission requests by analyzing over 30,000 apps. Moreover, Wang et al. [57] crawled three snapshots of Google Play in 2014, 2015, and 2017, and explored the evolution of various app properties, including permission usage, privacy policy declaration, advertising libraries, updates, and malicious behavior. However, these studies principally focus on revealing the evolution of apps' inherent properties, such as permission requirements, downloads, price, update frequency, instead of users' usage behavior. Our work in Paper III studied the longitudinal evolution of mobile apps based on users' usage behavior. Compared with previous studies, our work includes user-related indicators of the evolution, like intra-user diversity and inter-user diversity of mobile app usage.

## 2.2 Dataset Overview

In this section, we overview two real-world mobile usage datasets used in our studies. Also, we discuss ethical considerations.

### 2.2.1 Cellular Dataset

Cellular dataset was collected by a primary Internet Service Provider (ISP) in China [58]. Specifically, the dataset was gathered during one week in April 2016, covering the whole metropolitan area of Shanghai, one of the world's largest cities. The dataset includes over 2 million users and their network access records during the data collection period. The mobile usage dataset is characterized by the ISP with an anonymized user ID, timestamp, base station ID and network metadata.

The ISP derived a mobile app usage dataset from users' network access records. To determine the corresponding app ID for each network access record, the ISP

inspected the HTTP head and used the destination domain and user-agent as the app identifier. By adopting a systematic tool, SAMPLE [67], the ISP constructed conjunctive rules to match specific apps. SAMPLE applies a supervised learning algorithm over a small set of labeled data streams to automatically generate the conjunctive rules, which can identify over 90% of apps with an average accuracy of 99% [67]. In practice, the ISP built the conjunctive rules by manually operating a small set of apps to generate data streams. They then crawled the 2,000 most popular apps across app stores and matched network traffic records to these apps. Also, the ISP manually verified the correctness of the matched apps. In terms of the statistics from the ISP, more than 95% of network traffic used HTTP at the time of data collection, and they could map up to 90% of the network traffic to specific apps. During data collection, although some apps used HTTPS for critical functions, e.g., log-in, most parts of their traffic still used HTTP. Also, we notice most apps use HTTPS in recent years. Some existing studies [59, 61] have demonstrated that app usage traces can also be identified from encrypted data traffic. Overall, the app usage dataset provided by the ISP, although not covering all traffic, is sufficient for our analysis of mainstreams of usage behavior modeling. Each entity of the mobile app usage dataset contains an anonymized user identification, timestamp, base station ID, used app ID, and traffic volume. The complete app usage traces of the top 1,000 active users in the dataset are released to the research community, also including the app category information and the distribution of points of interest (POIs) under each base station. The public dataset is available at 'http://fi.ee.tsinghua.edu.cn/appusage/'.

### 2.2.2 Carat Dataset

Carat dataset is collected using a monitoring app called Carat [45]. Carat applies an event-triggered collection scheme, gathering a data sample every time the battery level changes by 1%. Specifically, each data sample contains a list of apps being used as mobile app usage information, user-specific identifier, and timestamp. Also, Carat collects sensor contextual data consisting of battery level, battery status, CPU usage, memory usage, mobile country code, and time zone. In order to boost the number of participants in data collection, we publish Carat on both Google Play[1] and Apple Store[2]. In this way, we conduct a worldwide data collection. The user will be informed of all data collection items when installing Carat in the End-user License Agreement (EULA). The data-gathering

[1]https://play.google.com/store/apps/details?id=edu.berkeley.cs.amplab.carat.android
[2]https://apps.apple.com/us/app/carat/id504771500

part of the platform is open-source[3], enabling users to examine it easily. Also, to reduce data collection expense, we motivated users to use Carat by designing Carat as a collaborative energy diagnosis app. In other words, Carat can provide personalized recommendations for improving smartphone battery life.

Up to now, Carat has gathered data from over 500,000 mobile users from over 100 countries. We released a long-term app usage dataset to the research community. The public long-terms app usage dataset contains the top 1,000 users ranked by the total duration of using Carat from 2014 to 2018. In the public dataset, the user with the longest duration has 18,146,042 time-series records spanning 4.65 years, and even the shortest duration user has more than two years of records. In particular, the public dataset is available at 'https://www.cs.helsinki.fi/group/carat/data-sharing/'.

### 2.2.3 Ethical Considerations

We are very aware of the privacy implications of using the datasets for research and our research findings. We have taken adequate measures to safeguard the privacy of the users involved. As for the Cellular dataset, the ISP has the consent to collect mobile data and stripped all the personally identifiable information from the traces. The ISP only gave us the anonymized user IDs. We never had access to their actual identifiers. Alternatively, for the Carat dataset, the data-gathering part of Carat is open-source. The users are informed of the data collection and management procedures and grant their consent from their devices. A user-specific identifier is randomly generated when a user first installs Carat. We do not have users' sensitive information. Also, both datasets are stored in a secure local server protected by strict authentication mechanisms and firewalls. All researchers are regulated by a strict non-disclosure agreement to access both datasets. All conducted works have received approval from relevant local institutions.

[3]http://carat.cs.helsinki.fi/.

# Chapter 3

# Discovering Daily Activity Patterns

This chapter gives an overview of the results of Paper I that solve RQ1 and RQ2:

- What activities can be discovered from anonymized mobile app usage data?
- What common patterns do we share with others in our daily activities?

We explore mobile big data from the people perspective and propose a framework to discover users' daily activity patterns from mobile app usage. We apply the framework to the Cellular dataset and identify seven typical activities from app usage, i.e., commuting and transportation, entertainment, shopping, socializing, reading and checking, life and health, and exploring food. We then successfully extract five common patterns among millions of people, including afternoon reading, nightly entertainment, pervasive socializing, commuting, and nightly socializing. We also show that people usually follow yesterday's activity patterns and the demographics have a significant impact on users' daily lives.

## 3.1 Discovery of User Activities

We first identify users' activities based on their mobile app usage traces. This chapter leverages the Cellular dataset that includes over 2 million users and their usage records over one week.

### 3.1.1 App Usage Trace Representation

To capture users' short-term activities, we divide mobile app usage traces one day into multiple small time windows. In practice, we use a time-based segmentation

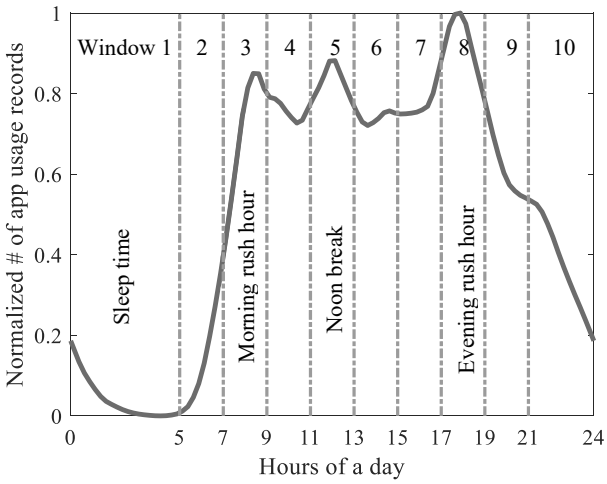



Figure 3.1: Min-max normalized number of app usage records during one day.

mechanism. As shown in Figure 3.1, we divide app usage traces of each day for each user into ten windows. In our case, a window refers to an app usage block composed of app usage records during a specific time slot. In the end, each user has 70 windows, i.e., 7 (# of days) $\times$ 10 (# of windows for each day). Also, we can apply the usual notions to some time slots, such as morning rush hour (7.00 to 9.00), noon break (11.00 to 13.00), and evening rush hour (17.00 to 19.00).

### 3.1.2 Activity Discovery

To characterize the activities of windows, we explore the power of the author-topic model [48]. As a probabilistic topic model, the author-topic model has been successfully used for discovering the hidden topic structure in documents. Given all words and authors of each document as observations, the author-topic model is trained to infer the hidden topic of each document. Alternatively, we aim to find the hidden activity structure of mobile app usage windows for activity discovery. Specifically, each window is a block of app usage traces, represented as a sequence of app IDs. Each window has multiple activity features, and each app usage of a window supports hidden activities in probability. Hence, the relationships among activities, apps, and windows, are highly similar to the relationships among topics, words, and documents. By considering they also have similar objectives, we build an analogy between the activity discovery of windows and the topic discovery of

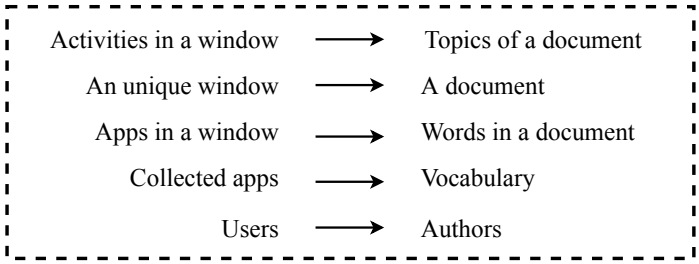



Figure 3.2: Analogy between window-activities to document-topics.

documents. As shown in Figure 3.2, a unique window represents a document, and an activity represents a topic. A window has multiple activity features, which is just like a document that has various topics. Apps in one window are deemed as words in a document. Vocabulary is the set of all words in documents, while apps collected are the set of all apps in windows. Thus, the apps collected are regarded as vocabulary. We then obtain the activity features of each window by applying the author-topic model.

### 3.1.3 Activity Identification

To facilitate finding the semantic terms of each window, we take a further step, aggregating similar unique windows in terms of their activity features. Windows from the same cluster have similar activity features, and different clusters represent different activities. By applying the Bisecting K-means clustering algorithm on the activity feature vectors of windows, we obtain seven clusters. Windows in the same cluster have similar activities. Thus, we next aim to identify each window cluster with semantic terms, i.e., activity labels. Note that activity identification is a very challenging problem. Unlike small-scale datasets, the large-scale app usage datasets lack meaningful labels mapping user activities to mobile app usage records. Fortunately, we can explore the semantic information of app categories and prior knowledge of activity temporal patterns to identify activity labels. For example, if a user uses food & drink apps during lunchtime, it has a high probability of identifying the activity as exploring food.

In detail, we identify the cyber activity label of a window cluster by considering the following three aspects: 1) The app category configuration in a window cluster. We compute the average proportion of different app categories for each window cluster. According to the calculated proportion, we rank app categories in a window cluster, called *Internal Ranking (IR)*. 2) The window cluster configuration across different app categories. We also rank the window clusters for each app

category, called *External Ranking (ER)*. 3) The temporal distribution of windows in each cluster. In Table 3.1, the top-3 internal app categories for each window cluster are colored by blue from darkest to lightest, while the top window cluster for each app category is colored by red. The seven typical window clusters are identified as follows.

(C1) *Commute and Transportation.* The top-3 app categories in this window cluster are *Music & audio*, *SON & IM*, and *Navigation.* This cluster contains the maximum number of *Transportation* apps as well.

(C2) *Entertainment.* This cluster contains typical entertainment activities with the highest both internal ranking and external ranking of *Media & video* and *Game*, as shown in Table 3.1.

(C3) *Shopping.* This cluster mainly has shopping activities with the most *shopping* apps. In the windows of this cluster, the socializing apps, i.e., *SON & IM* category, account for the highest proportion in the internal ranking. We infer that people usually browse the products in online shops and share them with friends via socializing apps to ask for suggestions and comments.

(C4) *Socializing.* This cluster is identified as the social activity since not only the *SON & IM* category has the highest internal ranking and external ranking but also the other app categories are of lower proportions compared with other clusters (see Table 3.1).

(C5) *Reading and checking.* The most characteristic app categories in this window cluster are *News*, *Reading*, *Sport*, *Weather*, *Stock*, and *Education*, with a significant higher proportion than other window clusters. These app categories are all about reading and checking activities.

(C6) *Life and health.* In this window cluster, the typical app category features are *Lifestyle* and *Health & fitness*, which are with both high external ranking and internal ranking as presented in Table 3.1.

(C7) *Exploring food.* Table 3.1 shows that the windows in this cluster are of the maximum proportion of *Food & drink* app usage both in external ranking and internal ranking.

## 3.2 Discovery of Activity Patterns

Activity patterns are of significant value for both individuals and society. For individuals, service providers can provide personalized service by exploring users' lifestyles, habits, occupations, and socio-economic status from their activity patterns. For society, the government can understand people's living status and

Table 3.1: Proportion of app categories for window clusters and corresponding internal and external rankings. C: *cluster*, Pro: *proportion*, IR: *internal ranking*.

| | Commute & transportation (C1) | | Entertainment (C2) | | Shopping (C3) | | Socializing (C4) | | Reading & checking (C5) | | Life & health (C6) | | Exploring food (C7) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **App categories** | **Pro** | **IR** | **Pro** | **IR** | **Pro** | **IR** | **Pro** | **IR** | **Pro** | **IR** | **Pro** | **IR** | **Pro** | **IR** |
| Game | 0.0109 | 9 | 0.3435 | 2 | 0.0083 | 16 | 0.0093 | 5 | 0.0127 | 18 | 0.0076 | 10 | 0.0997 | 2 |
| Finance | 0.0063 | 13 | 0.0017 | 21 | 0.0127 | 12 | 0.0056 | 10 | 0.0087 | 19 | 0.0074 | 11 | 0.0254 | 6 |
| Stock | 0.0027 | 18 | 0.0059 | 12 | 0.0130 | 11 | 0.0024 | 14 | 0.0642 | 5 | 0.0028 | 17 | 0.0047 | 17 |
| Shopping | 0.0127 | 8 | 0.0077 | 11 | 0.1223 | 3 | 0.0141 | 3 | 0.0197 | 10 | 0.0171 | 5 | 0.0191 | 7 |
| Parent & child | 0.0009 | 24 | 0.0107 | 8 | 0.0093 | 15 | 0.0008 | 23 | 0.0049 | 24 | 0.0015 | 20 | 0.0049 | 16 |
| Education | 0.0026 | 19 | 0.0005 | 24 | 0.0048 | 22 | 0.0022 | 15 | 0.0151 | 13 | 0.0020 | 19 | 0.0015 | 23 |
| Weather | 0.0022 | 20 | 0.0017 | 22 | 0.0049 | 20 | 0.0016 | 18 | 0.0084 | 20 | 0.0014 | 21 | 0.0023 | 22 |
| Travel | 0.0016 | 22 | 0.0023 | 19 | 0.0049 | 21 | 0.0009 | 21 | 0.0133 | 17 | 0.0009 | 24 | 0.0038 | 18 |
| Navigation | 0.1812 | 3 | 0.0039 | 15 | 0.0387 | 5 | 0.0085 | 6 | 0.0715 | 4 | 0.0138 | 7 | 0.0346 | 5 |
| Transportation | 0.1322 | 4 | 0.0129 | 7 | 0.0269 | 7 | 0.0036 | 12 | 0.0199 | 9 | 0.0049 | 14 | 0.0148 | 8 |
| SON & IM | 0.2226 | 2 | 0.0551 | 5 | 0.4335 | 1 | 0.8294 | 1 | 0.1410 | 2 | 0.1141 | 2 | 0.0976 | 3 |
| Food & drink | 0.0480 | 5 | 0.0729 | 3 | 0.1374 | 2 | 0.0765 | 2 | 0.0432 | 6 | 0.0224 | 4 | 0.5825 | 1 |
| Photography | 0.0062 | 14 | 0.0041 | 14 | 0.0071 | 18 | 0.0016 | 16 | 0.0064 | 23 | 0.0010 | 23 | 0.0416 | 4 |
| Lifestyle | 0.0080 | 12 | 0.0032 | 17 | 0.0484 | 4 | 0.0060 | 7 | 0.0150 | 14 | 0.6962 | 1 | 0.0113 | 9 |
| Health & fitness | 0.0031 | 17 | 0.0098 | 9 | 0.0050 | 19 | 0.0016 | 19 | 0.0140 | 16 | 0.0478 | 3 | 0.0052 | 14 |
| Sports | 0.0017 | 21 | 0.0026 | 18 | 0.0073 | 17 | 0.0009 | 20 | 0.0406 | 7 | 0.0142 | 6 | 0.0027 | 21 |
| News | 0.0056 | 15 | 0.0052 | 13 | 0.0237 | 8 | 0.0056 | 9 | 0.3280 | 1 | 0.0038 | 16 | 0.0051 | 15 |
| Reading | 0.0046 | 16 | 0.0082 | 10 | 0.0098 | 14 | 0.0016 | 17 | 0.0878 | 3 | 0.0064 | 13 | 0.0071 | 13 |
| Media & video | 0.0103 | 10 | 0.3632 | 1 | 0.0342 | 6 | 0.0052 | 11 | 0.0236 | 8 | 0.0069 | 12 | 0.0076 | 12 |
| Music & audio | 0.2693 | 1 | 0.0579 | 4 | 0.0138 | 10 | 0.0134 | 4 | 0.0173 | 11 | 0.0090 | 9 | 0.0103 | 11 |
| Business | 0.0302 | 6 | 0.0208 | 6 | 0.0108 | 13 | 0.0028 | 13 | 0.0149 | 15 | 0.0104 | 8 | 0.0103 | 10 |
| House & home | 0.0097 | 11 | 0.0035 | 16 | 0.0040 | 24 | 0.0008 | 22 | 0.0076 | 21 | 0.0024 | 18 | 0.0034 | 19 |
| Car | 0.0013 | 23 | 0.0010 | 23 | 0.0043 | 23 | 0.0005 | 24 | 0.0151 | 12 | 0.0014 | 22 | 0.0010 | 24 |
| Tools & others | 0.0260 | 7 | 0.0017 | 20 | 0.0149 | 9 | 0.0058 | 8 | 0.0071 | 22 | 0.0046 | 15 | 0.0033 | 20 |

Table 3.2: Average Levenshtein distance between arbitrary two days.

| | MON | TUE | WED | THU | FRI | SAT | SUN |
|---|---|---|---|---|---|---|---|
| MON | / | 4.26 | 4.37 | 4.42 | 4.58 | 5.26 | 5.18 |
| TUE | / | / | 4.37 | 4.48 | 4.59 | 5.29 | 5.21 |
| WED | / | / | / | 4.28 | 4.49 | 5.07 | 5.01 |
| THU | / | / | / | / | 4.47 | 5.27 | 5.29 |
| FRI | / | / | / | / | / | 5.27 | 5.31 |
| SAT | / | / | / | / | / | / | 4.70 |

detect disrupting trends from activity patterns and then make policies to improve people's well-being. Next, we sequence users' activities[1] identified from their app usage traces and apply sequence analysis methods to extract activity patterns.

We treat users' app usage traces of one day as an incidence of sequential activities. For each user, each day app usage traces reveal a activity sequence of length ten that contains combinations of the eight general activities including *Commute and transportation*, *Entertainment*, *Shopping*, *Socializing*, *Reading and checking*, *Life and health*, *Exploring food*, and *Unknown*. Hence, each user's activity sequence can be expressed as,

$$A_u = \{[a_1^{d_1}, a_2^{d_1}, ..., a_{10}^{d_1}], ..., [a_1^{d_7}, a_2^{d_7}, ..., a_{10}^{d_7}]\}, \tag{3.1}$$

where $A_u$ stands for the activity sequence of user $u$ and $a_m^{d_n}$ denotes the activity label of window $m$ in the *n-th* day, $a \in \{C1, C2, C3, C4, C5, C6, C7, C8\}$.

To quantify the degree of similarity among activity sequences, we apply the string metric. Each user's activity sequence for one day is regarded as a string, which is a combination of eight kinds of characters, i.e., activities. Particularly, we use the Levenshtein distance metric [42].

### 3.2.1 Individuals' Activity Analysis

We first investigate the similarity of different days to examine the regularity of activities in days' scale. As shown in Table 3.2, we calculate the average Levenshtein distance between two days in pairwise. Given day $i$ and day $j$, their distance is computed as,

$$\frac{1}{U}\sum_{u=1}^{U} \text{lev}(\mathbf{A}_u^{d_i}, \mathbf{A}_u^{d_j}), \tag{3.2}$$

[1]Apart from discovered seven activities, we add an Unknown label to denote silent time slots and denote it as $C8$.

where $U$ is the number of unique users and $\mathbf{A}_u^{d_i}$ denotes the activity sequence of the *i-th* day for user $u$.

We notice that there is an apparent difference between weekdays' sequences and weekends' sequences because the average distance between any two weekdays is less than that between any weekday and weekend. Besides, we discover an interesting appearance that the activity sequence of a weekday is more similar to yesterday's sequence. This implies that people intentionally or unintentionally obey yesterday's activity sequence, and there should be a daily pattern of activities for individuals. We then give the definition of the daily activity pattern of an individual as follows.

**Definition 1** *Daily activity pattern of an individual. Given an individual's activity sequences on weekdays,* $\mathbf{A}^{d_1}$, $\mathbf{A}^{d_2}$, ..., $\mathbf{A}^{d_5}$, *the daily activity pattern of the individual,* $\mathbf{A}$, *has the minimum sum-distance between all pairs of* $\mathbf{A}$ *and* $\mathbf{A}^{d_i}$. *Mathematically, denoting* $\mathbf{A} = [\mathbf{a_1}, \mathbf{a_2}, ..., \mathbf{a_{10}}]$, *then*

$$\mathbf{A} \leftarrow \arg \min_{\mathbf{a} \in \{\mathbf{C1}, \mathbf{C2}, ..., \mathbf{C8}\}} \sum_{\mathbf{i=1}}^{\mathbf{5}} \mathbf{lev}(\mathbf{A^{d_i}}, \mathbf{A}). \tag{3.3}$$

### 3.2.2 Identifying Common Activity Patterns

We next investigate whether there are common activity patterns for millions of users. To do this, we first quantify the distance between each pair of daily activity patterns for all users. Once the distance matrix is calculated, we apply the agglomerative hierarchical algorithm [13] to identify homogeneous clusters of daily patterns.

To determine the most appropriate number of clusters, i.e., patterns, we apply the dendrogram to evaluate the agglomerative hierarchical clustering algorithm, as shown in Figure 3.3. The dendrogram is a branching diagram representing the hierarchy of clusters based on the degree of similarity. Highly similar nodes or subtrees have joining points farther from the root. Thus, we know how the nodes are combined into larger parent clusters from the dendrogram, i.e., the detailed clustering process. We find five is the most appropriate number of clusters, where the clusters are of high intra-cluster and low inter-cluster distances. The five clusters are boxed using orange lines.

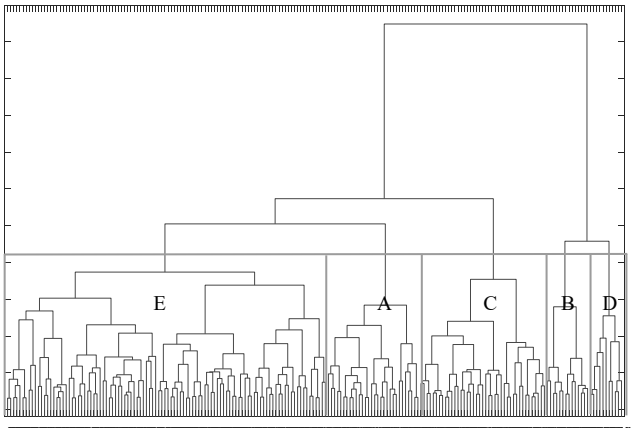
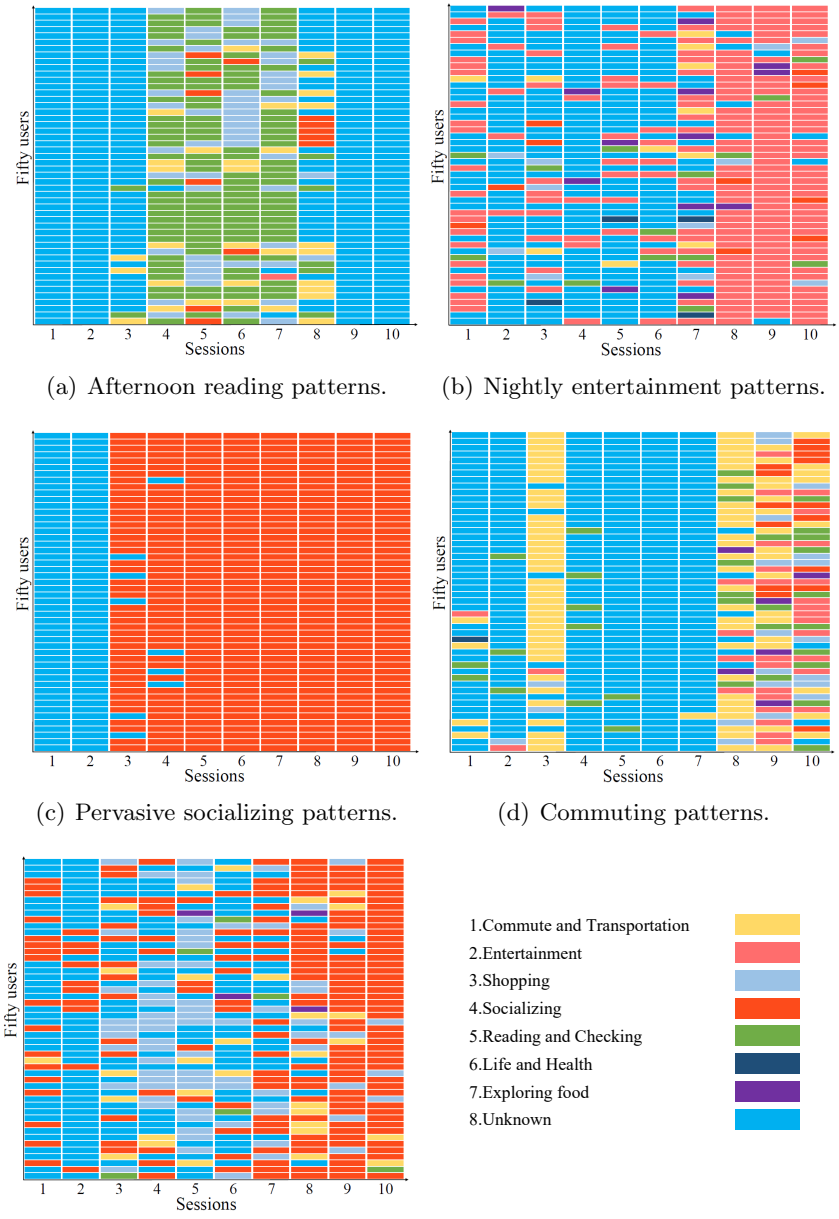



Figure 3.3: Dendrogram of the hierarchical clustering where highly similar nodes or subtrees have joining points farther from the root. Five is the most appropriate number of clusters.

### 3.2.3 Pattern Annotation

Given the clustering results, we then annotate each cluster of daily activity patterns with semantic terms, contributing to understanding the hidden image of these patterns. We first visualize them by randomly selecting fifty users for each cluster and show how their cyber activities are sequenced. As shown in Figure 3.4, the x-axis refers to the windows, and the y-axis indicates the random fifty users. Each bin refers to the activity label of that window for that user, and we use different colors to distinguish different activities.

**Afternoon reading** (Cluster A). The users in this cluster are mostly involved in *Reading and Checking* during the day, as shown in Figure 3.4(a). The users, on average, start to use mobile apps from time slot 4, 9.00 to 11.00, and become to be inactive after time slot 8, around 19.00. Hence, we annotate this cluster as afternoon reading to reflect this group's main active periods and activity. Although *Reading and Checking* activity dominates during time slots 4 to 7, there are still many users like *Shopping* during these hours. Generally, both two activities are leisure activities. We still notice that there are several *Commuting and Transportation* activities in this cluster. However, unlike Cluster D, *Commuting and Transportation* activities in this cluster are randomly distributed over time slots. Therefore, the users in this cluster do not have regular commute schedules, e.g., on and off work. Moreover, by considering the dominating pattern of reading and shopping activities, we infer the users in this group are senior citizens.




(a) Afternoon reading patterns.

(b) Nightly entertainment patterns.

(c) Pervasive socializing patterns.

(d) Commuting patterns.

(e) Nightly socializing patterns.

Figure 3.4: Visualization of daily activity patterns by randomly selecting fifty users in each cluster. Each row represent the activity patterns for one user.

**Nightly entertainment** (Cluster B). Figure 3.4(b) visualizes the daily patterns of this cluster. The users are engaged in *Entertainment* activities during evening and night, from 17.00 to 24.00. Hence, we annotate this cluster as nightly entertainment. Compared with clusters A, C, and E, the users have fewer activities during the usual active hours, i.e., from 7.00 to 15.00. Also, the users in this cluster are mostly nocturnal, and they are more than 18% of all the users. We infer that they are likely the younger generation. Due to the daytime classes, they only have free time in the evening and night, which may be why their app usage is so sparse during the daytime. Besides, the daily patterns show that many users in this cluster sleep late, still active in the time slot 1, from 0.00 to 5.00. Most of their *entertainment* activities last more than 6 hours, which indicates that the younger generation is addicted to the *Entertainment* activity, e.g., mobile games, and it is harmful to their health.

**Pervasive socializing** (Cluster C). As shown in Figure 3.4(c), the users in this cluster are engaged in the *Socializing* activity from 7.00 till 24.00. Compared with the patterns of other clusters, the patterns of this cluster are more regular concerning the active time of users and the duration of the dominating activity. This unusual pattern of social activities of nearly 7.5% users can be explained as follows. With modern social networking apps, social activities are not limited only to known friends and families. Peoples are making friends and communicating with people from different social classes via social apps. The social platforms are not only for interaction but also for various businesses, such as advertising and self-media. Therefore, we suspect that the users in this cluster work in call centers, customer services, or they are bloggers, cyberspace writers, and online shop owners.

**Commuting** (Cluster D). The patterns in this cluster shown in Figure 3.4(d) are typical commuting patterns. The *Commuting and Transportation* activities are sequenced regularly and mainly occur during rush hours. Due to the regular commute patterns, we infer most people in this cluster are involved in white-collar jobs. Meanwhile, we find an important phenomenon. In the morning, nearly 90% of *Commute and Transportation* activities happen in the morning rush hour. However, these activities are spread over multiple time slots around the evening rush hour. This phenomenon implies that many workers cannot knock off on time, and even working overtime becomes a habitual pattern for them. Like cluster B, most users are with limited activities from 9.00 to 17.00, as they are busy at work and do not have time to use smartphones.

**Nightly socializing** (Cluster E). The patterns of this cluster are shown in Figure 3.4(e). The dominating activity is *Socializing*, which mostly happens after

the evening, i.e., after time slot 6. The users in this cluster are mostly active during the day and engaged in other activities, such as *Shopping*. This implies their time is more flexible compared to other cluster users. The lack of sufficient commuting suggests that people are mostly staying at or near home, involved in household work during the daytime, and socializing in the evening. Hence, we infer the users in this cluster should be self-employed.

## 3.3 Chapter Summary

This chapter introduced Paper I, in which we leverage the cellular dataset to answer RQ1 and RQ2. We designed a probabilistic topic model based activity detection framework for discovering daily activity patterns across mobile app usage data. By applying our framework on the Cellular dataset, we identified seven typical activities, i.e., commuting and transportation, entertainment, shopping, socializing, reading and checking, life and health, and exploring food. From users' activity sequences, we examined the regularity of individuals' daily activities and successfully extracted five common patterns among millions of people, including afternoon reading, nightly entertainment, pervasive socializing, commuting, and nightly socializing.

# Chapter 4

# Understanding the Impact of Pandemic

This chapter introduces Paper II that solves RQ3 and RQ4:

- Does the outbreak of COVID-19 affect users' smartphone usage? If so, how?
- Can we use smartphone usage data to infer the outbreak stages of COVID-19?

We explore mobile big data from the people perspective and leverage less privacy-sensitive smartphone usage data to sense group health conditions, e.g., COVID-19 outbreak stages. Specifically, we gather smartphone usage records by using Carat, including the usage of mobile users in North America from November 2019 to April 2020. We then conduct the study on the differences in smartphone usage across the outbreak of COVID-19. We discover that COVID-19 leads to a decrease in users' smartphone engagement and network switches but an increase in WiFi usage. Also, its outbreak causes new typical diurnal patterns of both memory usage and WiFi usage. Additionally, we demonstrate the correlations between smartphone usage and daily confirmed cases of COVID-19 and leverage smartphone usage data the inference of outbreak stages, in which both Macro-F1 and Micro-F1 can achieve over 0.8.

## 4.1 Differences in Smartphone Usage

To determine whether the outbreak of COVID-19 changes users' mobile engagement, first of all, we need to determine the outbreak date in North America.

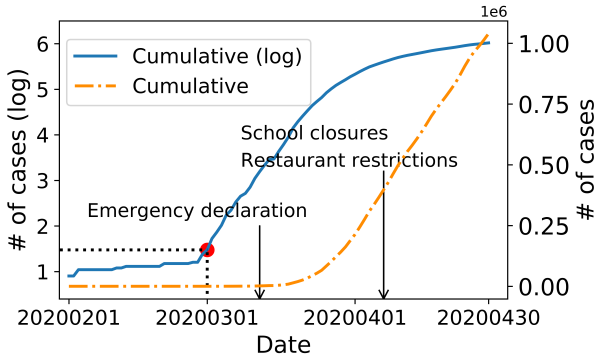


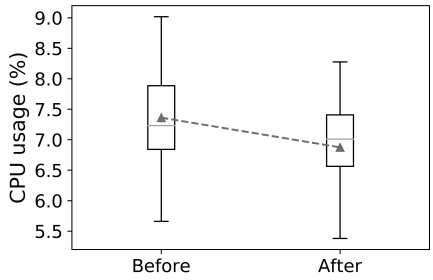

Figure 4.1: The cumulative number of confirmed cases changes over time. The federal government issued an emergency declaration on March 13, 2020. Most states issued school closure rules and restaurant restrictions by April 7, 2020.

Figure 4.1 shows the cumulative number of confirmed cases in North America from February 2020 to April 2020 and the governmental policies on the same timescale. The dashed curve is in the linear scale, while the solid curve depicts the cumulative number in the logarithmic scale. Notably, the propagation of COVID-19 is in exponential growth. Therefore, using the logarithmic scale curve makes it more accessible to detect the phase change of increase trend and determine the outbreak date accordingly [14, 38]. In terms of Figure 4.1, we can observe an apparent step-up around March 1, 2020, as denoted by the red point. Hence, we regard March 1, 2020, as the outbreak date of COVID-19 in North America.

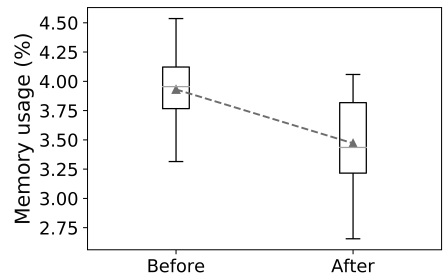
### 4.1.1 Differences in Number and Distributions

We then begin the analysis by comparing the distributions of smartphone usage variables before and after the outbreak of COVID-19. In Figure 4.2, we use box-plots to depict the distributions of the percentages of CPU usage, memory usage, WiFi usage, and network switches, respectively. Specifically, the 'Before' set contains the samples from November 1, 2019, to February 29, 2020, while the 'After' set contains the samples from March 1, 2020, to April 30, 2020.

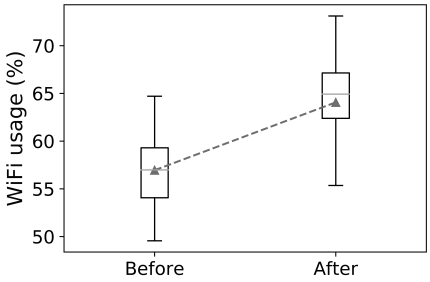
There is an apparent difference in smartphone usage across the outbreak in terms of all hardware variables. The mean values of CPU and memory usage drop from 7.36% and 3.93% to 6.87% and 3.47%, respectively. The decreases imply that users' smartphone engagement becomes less active after the outbreak, i.e., March 1, 2020. Meanwhile, the WiFi usage percentage grows dramatically, where the mean value rises from 56.95% to 64.06%. Since WiFi access points are



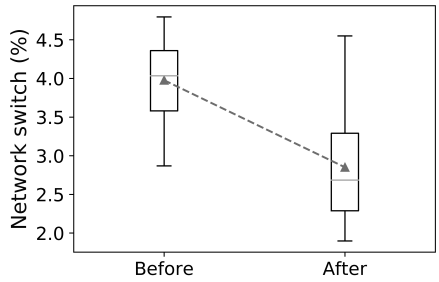

(a) The distributions of the percentage of CPU usage, $p = 5.239 \cdot 10^{-5}$.




(b) The distributions of the percentage of memory usage, $p = 7.383 \cdot 10^{-18}$.




(c) The distributions of the percentage of WiFi usage, $p = 2.585 \cdot 10^{-19}$.



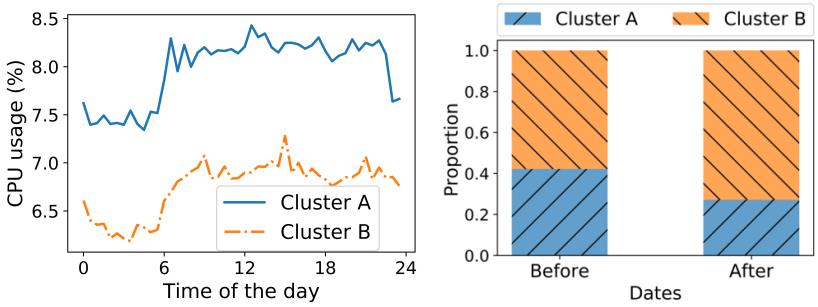

(d) The distributions of the percentage of network switches, $p = 1.526 \cdot 10^{-23}$.

Figure 4.2: The differences in smartphone usage before and after the outbreak of COVID-19.

usually deployed indoors, we can conclude that people have more time to stay indoors instead of going outside after the outbreak of COVID-19. Moreover, we also notice that the percentage of network switches drops remarkably. Similar to WiFi usage, network switches also reflect the movement of mobile users. Since the WiFi network is commonly deployed indoors and limited by its coverage, network switches usually occur when mobile users go from indoors to outside and from outside to indoors. Consequently, the percentage of network switches can reveal the mobility intensity of smartphone users. In this way, the decreasing trend of network switches suggests users have less mobility after the outbreak.

### 4.1.2 Differences in Diurnal Patterns

In terms of the above statistical analysis, we can conclude that the outbreak of COVID-19 has affected users' smartphone usage behavior. Next, we delve into the



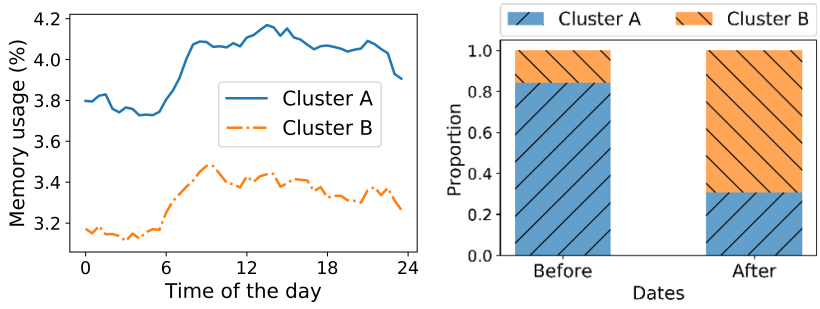

(a) The centroids of clusters. (b) Proportion of cluster labels.

Figure 4.3: Cluster results of CPU usage diurnal patterns.

dynamic analysis, i.e., revealing the differences in diurnal patterns. The diurnal pattern depicts how users' smartphone usage behavior unfolds over the time of the day, which is an essential temporal pattern studied by many previous studies.

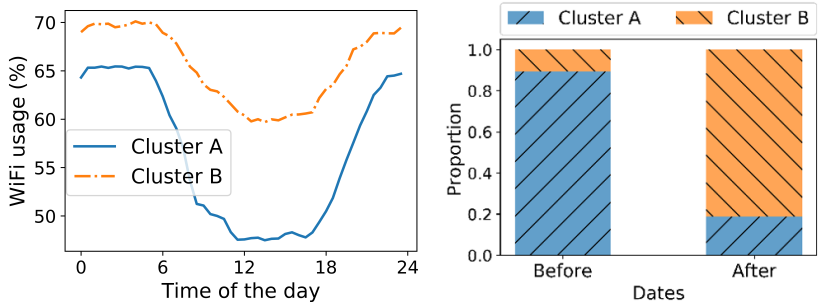
We define each day's diurnal pattern by averaging the usage data over the day's active users. In our case, we evenly divide one day into 48 time-slots, where each time slot represents half an hour. Therefore, each diurnal sequence is of 48 dimensions. Next, we compute smartphone usage data for each time slot. In practice, as for CPU usage and memory usage behavior, we take the averages in that time slot. For WiFi usage, we calculate the proportion of WiFi connection records in that time slot. Besides, for network switches, we calculate the proportion of network type changes in the time slot. By doing so, given one day, each type of smartphone usage behavior will have a diurnal sequence with 48 dimensions. In total, we have 728 diurnal sequences, i.e., 182 (# of days) $\times$ 4 (# of usage types).

We propose a hypothesis that the outbreak of COVID-19 will lead to a new diurnal pattern for smartphone usage. In our case, the new pattern means that it does not or rarely appears before the outbreak but is popular on the dates after the outbreak. To test the hypothesis, we apply K-means to cluster diurnal sequences of the entire 182 days for each type of smartphone usage behavior and examine whether the cluster results can be distinguished by the outbreak date of COVID-19. Since there are only two situations for any date, i.e., before or after the outbreak, we set the number of clusters to two. The clustering results are presented in Figures 4.3-4.6, where the cluster A and B refer to the two-cluster output of K-means. Also, we regard the centroid as the typical diurnal pattern of the cluster.




(a) The centroids of clusters. (b) Proportion of cluster labels.

Figure 4.4: Cluster results of memory usage diurnal patterns.

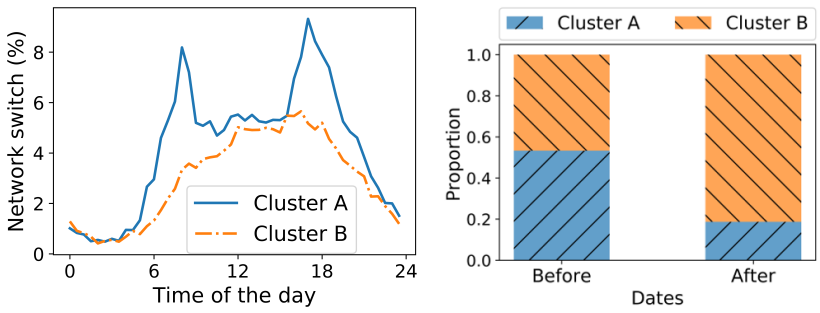



(a) The centroids of clusters. (b) Proportion of cluster labels.

Figure 4.5: Cluster results of WiFi usage diurnal patterns.

**Diurnal patterns of CPU usage.** As shown in Figure 4.3(a), the obtained two typical diurnal patterns of CPU usage have the same trend but different values. Both of them decrease during the night and increase during the day, while cluster B's centroid is of lower numerical values. Figure 4.3(b) shows that, compared to cluster A, cluster B accounts for a higher proportion of the dates after the outbreak, consistent with the dropping trend observed in Figure 4.2(a). We also observe that COVID-19 only affects the proportion of two cluster labels, and both typical patterns frequently appear on the dates before the outbreak. In other words, the outbreak did not create a new typical diurnal pattern of CPU usage.

**Diurnal patterns of memory usage.** As depicted in Figure 4.4(a), similar to CPU usage, two typical diurnal patterns obtained are also with the same trend but different numerical values. In terms of Figure 4.4(b), over 80% of the dates




(a) The centroids of clusters. (b) Proportion of cluster labels.

Figure 4.6: Cluster results of network switch diurnal patterns.

before the outbreak belong to cluster A. Meanwhile, more than 65% of the dates after the outbreak belong to cluster B. Therefore, we can conclude that the cluster results can be distinguished by the outbreak date. Also, cluster B's centroid can be regarded as a new typical diurnal pattern because it rarely appears before the outbreak and becomes common after the outbreak. In summary, COVID-19 leads to the appearance of a new typical diurnal pattern of memory usage.

**Diurnal patterns of WiFi usage.** Figure 4.5 displays the cluster results of WiFi usage. Unlike CPU and memory usage, apart from numerical differences, the centroids of WiFi usage clusters also have different changing trends. As depicted in Figure 4.5(a), the centroid of cluster B has a higher percentage of WiFi usage throughout the day. Instead of a cliff-like drop shown in cluster A, cluster B has a slow-down after 6 am. This indicates that users need less mobile network support on the dates in cluster B. Moreover, similar to memory usage, the dates after the outbreak have a dominating cluster, i.e., cluster B. Therefore, COVID-19 also brings a new diurnal pattern of WiFi usage, leading users to use more WiFi connections.

**Diurnal patterns of network switches.** We exhibit the clustering results of network switch patterns in Figure 4.6. Network switches can reflect the mobility intensity of smartphone users. In Figure 4.6(a), the centroid of cluster A presents two peaks in the morning and evening rush hours, which verifies the above discussion. We notice that less than 18% of the dates after the outbreak belong to cluster A, indicating that users' mobility intensity drops significantly. Alternatively, cluster B has fewer network switches throughout the day and without bimodal patterns, indicating that users have less mobility on the dates in that cluster. Although cluster B dominates the dates after the outbreak, it

also frequently appears before the outbreak. As a result, similar to CPU usage, COVID-19 only changes the proportion of different network switch patterns but does not trigger the appearance of new patterns.

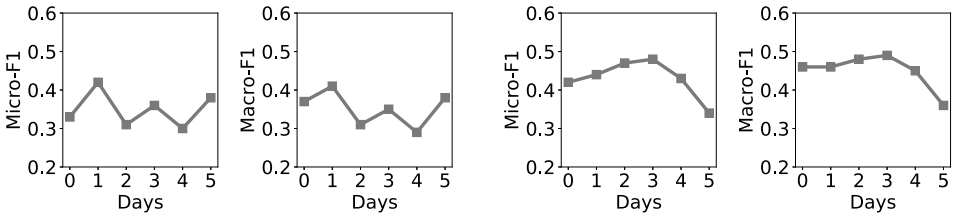
Consequently, the outbreak of COVID-19 also profoundly affects diurnal patterns of smartphone usage behavior. Such observations imply that the diurnal sequences of smartphone usage can be used to reflect the outbreak status.

## 4.2 Inference of Outbreak Stages

We study whether we can use smartphone usage data, e.g., CPU usage, memory usage, and network connections, to infer the outbreak stages of COVID-19. Recalling Figure 4.1, we can witness that the outbreak of COVID-19 has shown three stages from March 1, 2020, to April 30, 2020. First, the dates from February 1, 2020, to March 1, 2020, are the early stage of COVID-19, with only a few cases appearing. Second, during the dates from March 1, 2020, to April 1, 2020, the daily confirmed cases increased dramatically. Third, on the dates after April 1, 2020, the increasing trend of COVID-19 cases is stable. Therefore, we label COVID-19 outbreak stages with three classes, i.e., early, dramatic, and stable. By doing so, the inference problem is converted into a 3-class classification problem. Specifically, we infer the outbreak stages of one day by using its diurnal sequences of different smartphone usage behavior, including CPU usage, memory usage, WiFi usage, and network switches. Also, to evaluate the performance, we use Macro-F1 and Micro-F1 as metrics. The higher the value of Macro-F1 and Micro-F1, the better the performance. For all experiments, we obtain the results by employing a five-fold cross-validation policy on our dataset.

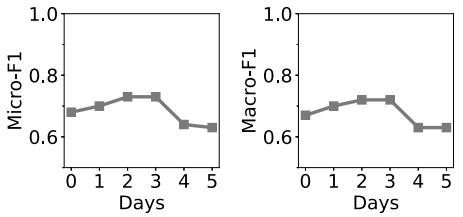
### 4.2.1 Delay Analysis of Stage Inference

Users' smartphone usage behavior can reflect their physical activities and the outbreak stages of COVID-19. However, the reflection may not be immediately expressed by the daily cases of COVID-19 due to the incubation period and diagnosis delay. Hence, we explore the typical time delay of stage inference. Specifically, we infer the outbreak stage of one day by utilizing the smartphone usage features of the days before it. In practice, we conduct the inference with the three most commonly used classification algorithms, logistic regression (LR) [30], support vector machine (SVM) [54] and Xgboost [9]. We infer the outbreak stages of one day by concatenating all behavior types' diurnal sequences, including CPU usage, memory usage, WiFi usage, and network switches.




(a) Performance with LR classifier.

(b) Performance with SVM classifier.




(c) Performance with Xgboost classifier.

Figure 4.7: COVID-19 outbreak stage inferences with different time delays.

We show the results in Figure 4.7. The LR classifier has poor performance, and F1 scores fluctuate on different delays. That is because the LR classifier only uses a logistic function to model the correlation, which is more susceptible to outliers tampering with the performance. Therefore, it is hard to capture the relations between smartphone usage features and COVID-19 outbreak stages with the LR classifier using the real-word dataset that might have noisy data points. Alternatively, as shown in Figure 4.7(b) and Figure 4.7(c), SVM and Xgboost classifiers have better performance. Also, we can observe that F1 scores achieve the highest value under a delay of 2 or 3 days. This observation confirms that the reflection of users' smartphone usage behavior will emerge in COVID-19 trends with a time delay of a few days.

### 4.2.2 Smartphone Usage Behavior Embedding

We further propose an embedding model to fuse different smartphone usage behavior effectively for improving inference performance. Given a day, we first construct a diurnal smartphone usage feature sequence $\{u_i\}_{i=1}^{48}$, where $u_i$ is a vector containing all four usage features in the $i$-th timeslot of the day. We then

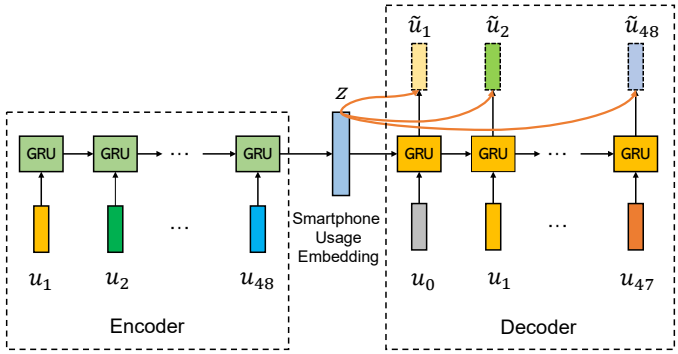



Figure 4.8: Seq2Seq model for smartphone usage embedding.

utilize a Seq2Seq [12] model to learn an embedding from the diurnal sequence. As shown in Figure 4.8, the model consists of an encoder and a decoder, which are implemented with a GRU network [12]. The sequence $\{u_i\}_{i=1}^{48}$ is fed into the encoder to obtain an encoding vector of $z$. Then, $z$ and a shifted usage sequence $\{u_i\}_{i=0}^{47}$ are fed into the decoder to reconstruct the original sequence, where $u_0$ is a vector that contains all 1. Moreover, to encode comprehensive information in vector $z$, we engage $z$ in the reconstruction. Formally, the $i$-th unit of the decoder takes $u_{i-1}$ as input and outputs hidden state $\hat{h}_i$ , we infer $\hat{u}_i$ as,

$$\hat{u}_i = \sigma(W[\hat{h}_i, z] + b), \tag{4.1}$$

where $[,]$ is the concatenating operation, $\sigma$ is the sigmoid activating function, $W$ and $b$ are trainable parameters. Finally, we train the model by minimizing the reconstruction loss,

$$\mathcal{L} = \sum_{i=1}^{48} |\hat{u}_i - u_i|^2. \tag{4.2}$$

We train the model and obtain a usage embedding vector for each day. To evaluate whether the embedding fuses different usage features better, we conduct the inference on the original features (Raw) and the original features concatenated with the learned embeddings (Raw + Embedding). We again use the Xgboost classifier as the inference model. We compare the performance with embeddings, as shown in Figure 4.9. We can observe that, by combining with embeddings, we improve the entire performance under different delay settings. Especially when the delay is set as two days, the performance of raw features combined with embeddings reaches around 0.87 for both Macro-F1 and Micro-F1, which has an

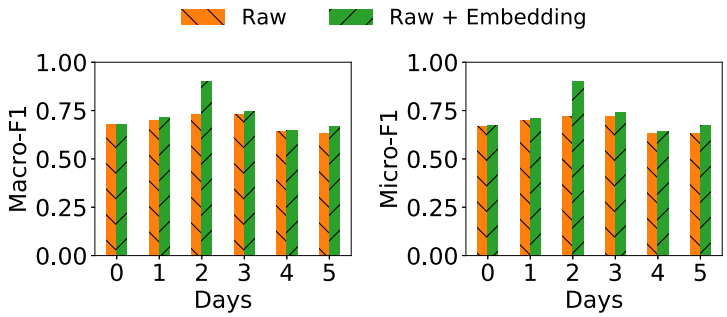



Figure 4.9: Outbreak stage inferences with embeddings.

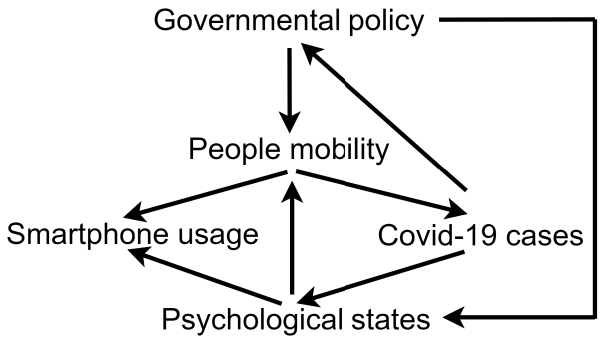



Figure 4.10: A potential causality diagram of smartphone usage and COVID-19 cases.

over 20% improvement compared with the best performance of only using raw features. These results demonstrate that the learned embeddings fuse multiple features more effectively indeed.

## 4.3 Discussion

Although we have examined the correlation between smartphone usage behavior and COVID-19 cases, their causality relationship still needs further exploration. In Figure 4.10, we depict a potential causality diagram of smartphone usage and COVID-19 cases. People mobility and psychological state serve as a confounder and mediator connecting smartphone usage and COVID-19 cases, respectively. Smartphone usage is directly affected by mobility and can act as a mobility indicator. Also, smartphone usage is still affected by the psychological states of

users [17, 41]. Meanwhile, the causation between people mobility and COVID-19 cases is bidirectional. On the one hand, frequent people mobility will trigger new COVID-19 cases. On the other hand, COVID-19 will affect people's mobility through governmental policies and their psychological states. Therefore, the causation between smartphone usage and COVID-19 cases might be complex. As for checking the potential causality diagram we proposed, we leave it to future work.

## 4.4 Chapter Summary

This chapter introduced Paper II, in which we leverage the Carat dataset to answer RQ3 and RQ4. Our findings indicate that users' smartphone usage indeed changes across the outbreak of COVID-19. However, the outbreak has different effects on different usage behavior regarding changing trends and diurnal patterns. Also, we demonstrate the potential of using smartphone usage data to infer the outbreak stages, achieving over 0.8 for both Macro-F1 and Micro-F1. Our findings provide a novel application of smartphone usage data and explore their values for fighting against the epidemic and detecting group health conditions.

# Chapter 5

# Longitudinal Evolution of Mobile App Usage

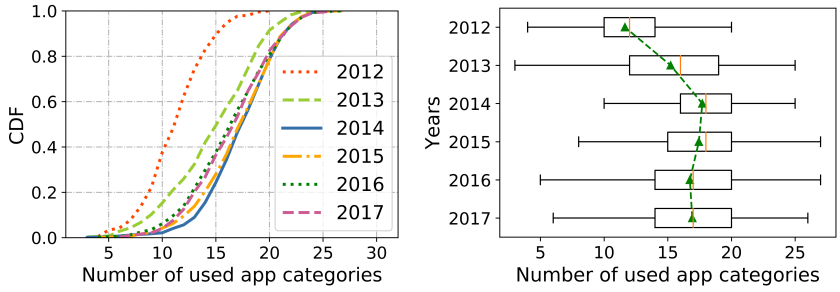
In this chapter, we give an overview of the results of Paper III aiming to solve RQ5:

- What is the evolution of mobile app usage behavior over time?

We aim to reveal how mobile app usage evolves over a long-term period. Specifically, we leverage the Carat dataset and extract mobile app usage records of long-term users (1,465 users) from 2012 to 2017. We then conduct the study on the long-term evolution processes on a macro-level, i.e., app-category, and micro-level, i.e., individual app. We discover that, on both levels, there is a growth stage enabled by the introduction of new technologies. Then there is a plateau stage caused by high correlations between app categories and a Pareto effect in individual app usage, respectively. Additionally, the evolution of individual app usage undergoes an elimination stage due to fierce intra-category competition. Nevertheless, the diverseness of app-category and individual app usage exhibit opposing trends: app-category usage assimilates while individual app usage diversifies.

## 5.1 Evolution of App-category Usage

In this section, we reveal the evolution of app category usage in terms of number of app categories used, diversity of app-category usage and popularity of app categories.




(a) CDF of the number of app categories used by each unique user.

(b) Distribution of used app categories across different years.

Figure 5.1: Evolution of app-category usage across six years.

### 5.1.1 Number of App Categories

We begin our analysis by investigating the most intuitive metric of app-category usage, i.e., the number of app categories used by each user during a given year. Figure 5.1(a) presents the Cumulative Distribution Function (CDF) of the number of used app categories for all long-term users from 2012 to 2017. We observe that the evolution of app-category usage undergoes two stages:

- **Stage one (2012 - 2014).** In this stage, the number of app categories used by each user increased significantly. The increasing trend suggests that during this stage, *smartphones were endowed with more functions, and people started using smartphones in more diverse activities.*
- **Stage two (2014 - 2017).** During this stage, the number of used app categories remained stable over time, which implies that *both smartphones' functions and users' usage at the app-category granularity became steady.*

Alternatively, to better illustrate the changes in the number of app categories used, we depict the distributions across different years using box-plots in Figure 5.1(b). From 2012 to 2014, the values in the interquartile range, i.e., the boxed area, increased significantly, reinforcing Figure 5.1(a). However, after 2014, the third quartile is constant, implying the group of users who use relatively more app categories remained stable. Although the first quartile dropped slightly until 2016, there was no discernible change in terms of the average value.

One possible reason for the increase in app categories used in stage one is the development of mobile networks. In terms of the mobile network types in our

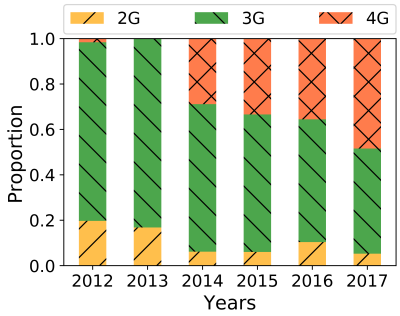



Figure 5.2: Proportions of mobile network types.

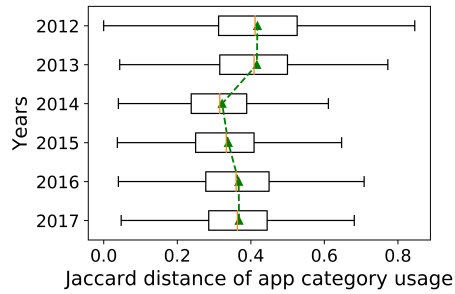



Figure 5.3: Jaccard distance of app-category usage.

dataset, we present how the proportions of different mobile network types changed from 2012 to 2017 in Figure 5.2. We can observe that by 2014, around 30% of users collected were using 4G networks, and the fraction grew steadily after that. Compared to 3G providing up to 21.6 Mbit/s download rate, 4G networks can support 1 Gbit/s or about 50 times that of 3G [29]. As a result, mobile networks no longer inhibit the usage of latency-sensitive apps and data consuming apps, e.g., online gaming apps, online video apps, and map apps. Therefore, more app categories are widely used by mobile users to facilitate and color their lives.

### 5.1.2 Diversity of App-category Usage

We next study the diversity of app-category usage across different users, which measures the magnitude of one user's usage behavior different from others [18]. In practice, we apply Jaccard distance to measure the difference in app-category usage between two users. For each year, we compute the Jaccard distance between every two users and illustrate the distributions using box-plots, as shown in Figure 5.3. We notice that the average pairwise distance, denoted as the green triangle, shows a downtrend from 2013 to 2014. The average value dropped dramatically from 0.42 to 0.32. Although there was a slight increase after 2014, the average pairwise distance was still much lower than that of 2013. Also, the distribution did not significantly change from 2014 to 2017. The trend reflects that users' requirements for smartphone functions tend to be consistent.

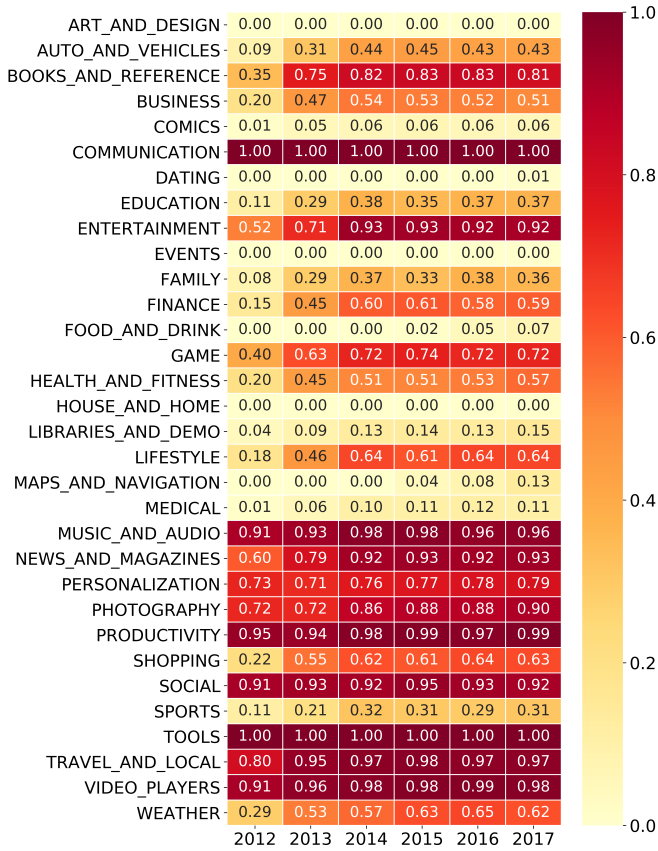



Figure 5.4: App category popularity across different years.

### 5.1.3 Popularity of App Categories

To understand which app categories are more competitive and explore general laws in usage evolution, we next investigate how the popularity of each app category changes over time. In our case, we measure the popularity in terms of unique users, which is the ratio of the users who used that app category to all long-term users. For instance, if one app category has a popularity of 0.9, it means that 90% of long-term users have used at least one app belonging to that app category. Figure 5.4 shows the popularity of each app category across different years.

We first focus on the prevalent app categories. We define an app category as prevalent if its popularity is higher than 0.9. The prevalent app categories

represent the critical requirements and preferences of mobile users. We discover that there are two types of prevalent app categories distinguished by their evolution processes:

- **Prior prevalent app category**. This type refers to the category whose popularity has exceeded 0.9 since 2012. There are six prior prevalent categories, including 'Communication', 'Music and audio', 'Productivity', 'Social', 'Tools', and 'Video players', which suggests smartphones have already acted as communication devices and multimedia players since 2012.

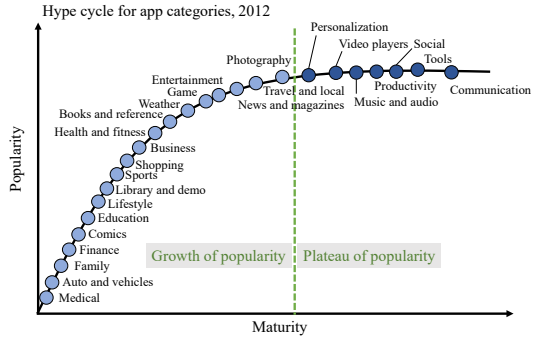
- **Posterior prevalent app category**. This type refers to the category whose popularity reached 0.9 after 2012, which suggests changes in smartphone roles. There are four posterior prevalent categories, i.e., 'Entertainment', 'News and magazines', 'Photography', and 'Travel and local'.

Compared to prior prevalent categories, posterior prevalent categories are more relevant to life services. The emerging of posterior prevalent categories implies smartphones changed from communication tools to life assistants coloring users' daily lives. Also, in terms of the popularity changes across app categories, we present the hype cycles of popularity for app categories in Figure 5.5. The hype cycle shows the relationship between the maturity of app categories with their popularity. In the hype cycle, we only focus on depicting changes in popularity rather than exact values. Generally, if the app category is more mature, then its popularity is more stable. As shown in Figure 5.5, the evolution of app category popularity undergoes two stages, i.e., growth of popularity and plateau of popularity:

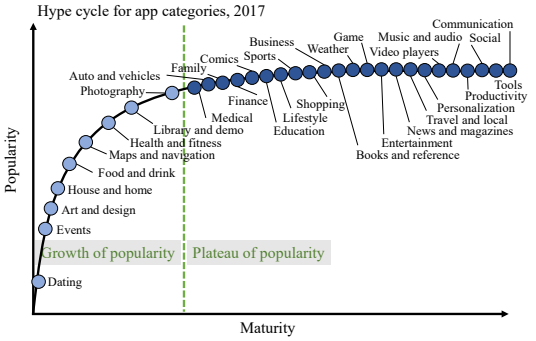
- **Growth of popularity.** In this stage, the popularity of the app category increases. When an app category is newly introduced, it will be at this stage initially. The development of technologies and smartphone designs, like 4G networks and larger screen sizes, will trigger an increase in multiple app categories' popularity.

- **Plateau of popularity.** In this stage, the popularity of the app category tends to be stable, which suggests that the market in this app category is mature. For different app categories, their steady popularity is different because they have different potential customers.

Surprisingly, there is no discernible decline stage during the popularity evolution of app categories. One major reason might be the high correlations between




(a) The hype cycle of popularity for different app categories in 2012.




(b) The hype cycle of popularity for different app categories in 2017.

Figure 5.5: The evolution of app category popularity.

app categories. With the development of the app market, a stable and highly correlated app ecosystem has been formed. Various app categories are connected with and reliant on others. Due to the high correlations among app categories, users have to keep using multiple app categories together. For example, for online shoppers, apart from 'Shopping' apps, they have to use 'Finance' apps for online payment as well.

### 5.1.4 Correlations of App Categories

To validate the previous inference about the app ecosystem, we then study the correlations of app categories. In our case, we use the co-usage of app categories for unique users to represent their correlations. Figure 5.6 displays the correlations of

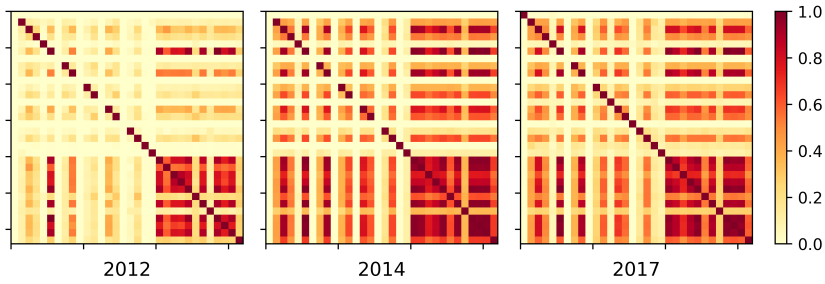



Figure 5.6: The correlations of app categories across different years.

app categories in 2012, 2014, and 2017, respectively. In the heatmap, each row or column represents one app category. The categories are sorted in descending order by their first letter (the same as Figure 5.4). From Figure 5.6, we can observe that the strength of correlations between app categories generally increased from 2012 to 2014. Comparing the heatmaps in 2014 and 2017, the correlations across various app categories were high and tended to be stable, suggesting that a robust app ecosystem had formed.

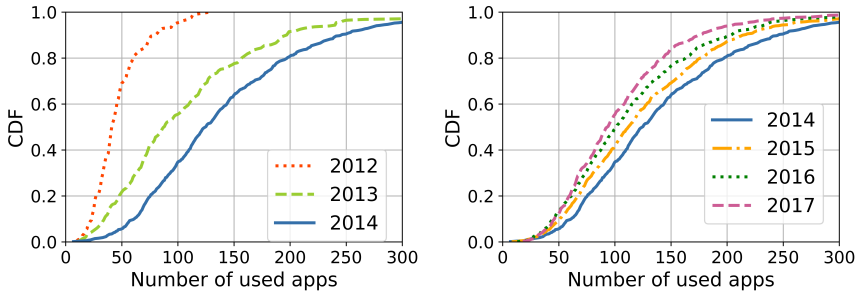
## 5.2 Evolution of App Usage

In this section, we depict the evolution of app usage in terms of number of app used, diversity of app usage, app popularity, and app usage Within app categories.

### 5.2.1 Number of Used Apps

We first analyze the number of apps used by each unique user. As shown in Figures 5.7(a) and 5.7(b), similar to app categories, the evolution of app usage is also separated into two stages by the year 2014:

- **Stage one (2012 - 2014).** During this stage, users increased the number of apps used on their smartphones. This boosting period at the micro-level is consistent with the macro-level. As analyzed before, the occurrence of this stage should be motivated by the release of new technologies.
- **Stage two (2014 - 2017).** During this stage, the number of apps used by each user decreased year by year, which is significantly different from the trend at the macro-level. In 2017, the proportion of users who used over 150 apps fell to 20%.




(a) CDF of the number of used apps for each user from 2012 to 2014.

(b) CDF of the number of used apps for each user from 2014 to 2017.

Figure 5.7: The evolution of app usage from 2012 to 2017.

### 5.2.2 Diversity of App Usage

We next explore how the diversity of app usage changes over time. By applying Jaccard distance to measure the difference of app usage between every two users, we depict the distribution of pairwise Jaccard distances across different years in Figure 5.8. From 2012 to 2013, the average distance between two users jumped from 0.75 to 0.85, implying the diversity of app usage increased. The trend is contrary to that at the macro-level in Figure 5.3. After 2013, the distribution became stable, i.e., the strength of diversity stopped increasing. However, users' used apps were still extremely different from others considering the minimum distance is nearly 0.7.

As a result, the diversity between users exhibits two opposite evolutionary trends at the micro-level, i.e., apps, and the macro-level, i.e., app categories, respectively. At the macro-level, mobile users fully explore the functionality of smartphones and tend to use more and similar app categories. On the other hand, at the micro-level, mobile users have different preferences and use a diverse array of apps.

### 5.2.3 Distribution of App Popularity

We further study the distributions of app popularity from 2012 to 2017. Figure 5.9 reports the CDF of app popularity (the ratio of app users to all users). Our results reveal a typical *Pareto effect* for app usage. Over 80% of apps have less than 0.01 popularity in 2012, while this number increased to 90% by 2017. The Pareto effect suggests that although the set of apps used by one user are quite

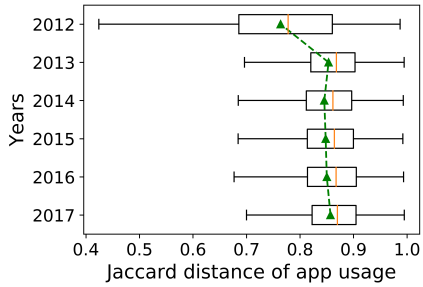



Figure 5.8: Jaccard distance of app usage.

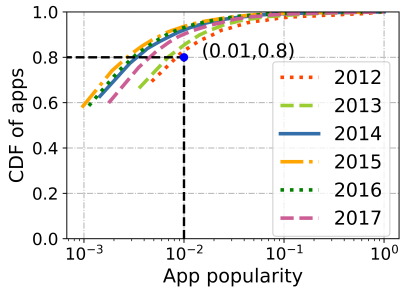



Figure 5.9: CDF of the popularity of apps.

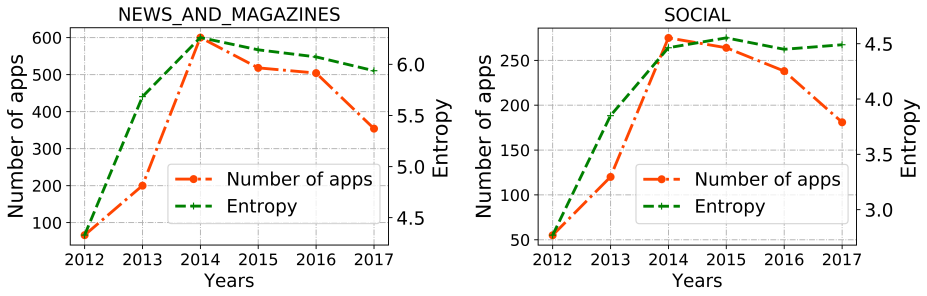
different from others, the app market is still governed by a small number of dominating apps. This observation is consistent across all six years.

### 5.2.4 App Usage Within App Categories

Up to now, we have discovered that the evolutionary processes at the macro-level and the micro-level show considerable differences, especially during stage two, i.e., from 2014 to 2017. Therefore, we next delve into the reasons behind this phenomenon and investigate how app usage changes in a particular app category. For the sake of representativeness, we actually select two typical app categories, i.e., 'News and magazine' representing a posterior prevalent app category and 'Social' representing a prior prevalent app category.

In our case, we apply the number of apps and app usage entropy to measure the evolution processes. Figure 5.10 shows the results. The entropy is a common metric to measure the randomness of a system. We use entropy to measure the centralization of app usage in one specific app category, i.e., whether app usage in that category concentrates on a few apps. The lower the entropy, the higher the centralization of app usage.

In terms of Figure 5.10, for both 'News and magazine' and 'Social' categories, the number of apps peaked in 2014 and then decreased. Additionally, we have also examined the other app categories and found their trends are consistent as well. However, the evolution in entropy exhibits different trends in 'News and magazine' and 'Social' categories. For the 'Social' category, entropy first increased and then kept steady. The increase stage is caused by the growing number of apps in the category. New apps disperse users' concentration. On the other hand, the Pareto effect leads to the plateau stage. As a prior prevalent app category, 'Social' had a

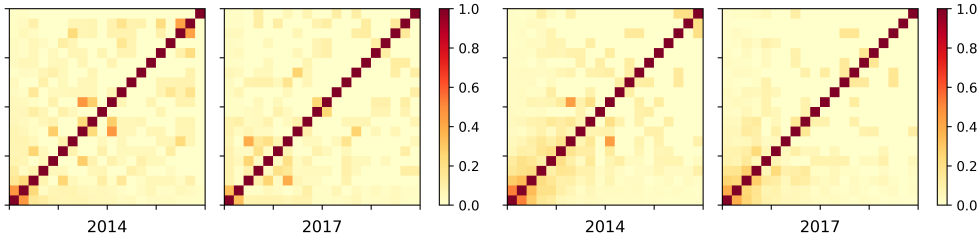



(a) 'News and magazine' category. (b) 'Social' category.

Figure 5.10: Evolution of app usage in 'News and magazine' and 'Social' categories.

few governing apps dominating usage before 2012. Therefore, during the boosting period, the newly introduced apps would compete with these old governing apps, and some low-quality would be eliminated. Meanwhile, new governing apps would emerge. As a result, in 2014, apart from the increasing entropy, users' usage was also hugely dominated by both previous and new governing apps. Therefore, after 2014, the entropy did not change dramatically. For the 'News and magazine' category, the evolution in entropy still experienced the decrease stage. Since 'News and magazine' is a posterior prevalent app category, limited by its maturity, it had few governing apps before 2012. Hence, its entropy is deeply affected by the number of apps in the category.

In order to better understand the app elimination stage, we also investigate how the correlations of apps in the same app category changed from 2014 to 2017. Similar to Subsection 5.1.4, we use the co-usage of apps for unique users to represent their correlations. For consistency, we still use 'News and magazine' and 'Social' to represent posterior and prior prevalent app categories, respectively. In Figure 5.11, we depict the correlations of the top 20 popular apps in these two categories. In the heatmap, each row or column represents one app. The apps are listed in descending order in terms of their popularity. Compared with app categories, the correlations of apps in the same category is much lower, and most are below 0.2. Since the functionality of apps in the same category is similar, installing multiple apps from the same category is often redundant. By comparing the top 20 popular apps in both 'News and magazine' and 'Social' categories from 2014 to 2017, we then discover the relationship between correlations and popularity of apps. The apps with high correlations have a greater chance of gaining popularity in the future.




(a) Correlations of apps in 'News and magazines' category.

(b) Correlations of apps in 'Social' category.

Figure 5.11: Correlations of apps in 'News and magazine' and 'Social' categories.

## 5.3 Chapter Summary

This chapter introduced Paper III, in which we leverage a long-term mobile app usage dataset collected by Carat to solve RQ5, understanding the long-term evolution of mobile app usage. Our analysis covers about 1,500 Android users with six-year app usage records from 2012 to 2017. Our findings indicate that users' app usage indeed changes over time. However, the evolutionary processes in app-category usage and individual app usage are different in terms of popularity distribution, usage diversity, and correlations.

# Chapter 6

# Revealing Urban Land Usage Patterns

This chapter overviews the contributions of Paper IV for solving RQ6:

- Is it possible to use mobile app usage data to reveal urban land usage patterns?

A city is composed of many regions providing different functions for urban residents, for example, residential regions and business regions. Additionally, due to daily urban dynamics, a region might provide different functions at different times of the day. In this chapter, we leverage spatiotemporal mobile app usage data to reveal urban land usage patterns. Specifically, we propose a graph-based representation learning framework, where an embedding vector represents a region at a specific time interval. We then evaluate our framework through a series of experiments conducted on the cellular dataset.

## 6.1 Framework Overview

We present an overview of our proposed framework in Figure 6.1. The complete process of the framework can be described as follows. First, based on road networks, we partition the city into multiple disjoint regions. These regions are treated as atomic units to study dynamic region functions. Then using this region data along with spatiotemporal mobile app usage data, we construct a heterogeneous app usage graph. Next, we derive a relational location graph from the app usage graph with the POI distribution as region features. Then, we obtain dynamic region embeddings by feeding the attributed relational location graph

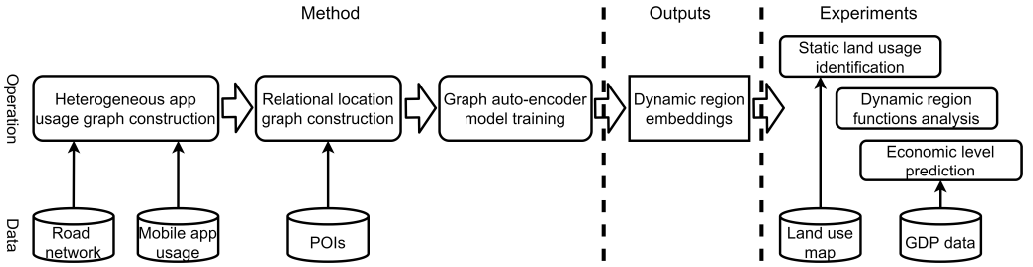



Figure 6.1: Overview of our proposed framework.

into a graph auto-encoder model for training. The main purpose of the graph auto-encoder model is to fuse both graph structure and node feature information into the node embeddings. In our case, each region at a specific time interval has a corresponding embedding, representing the characteristics of that region in that time interval. Finally, we verify our model using three illustrative applications, including static land usage identification, dynamic region functions analysis, and economic level prediction. In practice, for static land usage identification, we use the official land use map as the ground truth, while for economic level prediction, we take GDP data of administrative districts as the ground truth.

## 6.2 Method

Mobile app usage data contains dynamic relationships between users, apps, and locations. To encode such connections between different entities, we first build a heterogeneous app usage graph and then formalize it as a meta-path guided homogeneous relational location graph.

### 6.2.1 Heterogeneous App Usage Graph

The interactions in mobile app usage behavior can be abstracted as a heterogeneous graph containing three types of entities, i.e., users, apps, and locations. Figure 6.2 shows the structure of the heterogeneous app usage graph, where $U$ refers to *user* nodes, $A$ refers to *app* nodes, $L$ refers to *time-enhanced location* nodes, and the edges reflect the co-occurrence of different objects in mobile app usage records. We note that since particular regions can exhibit different roles at different time slots, we use time-enhanced location nodes to represent these dynamic relationships.

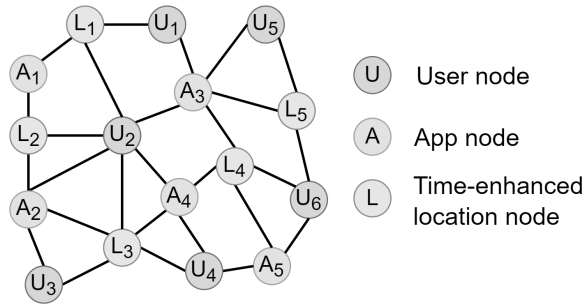



Figure 6.2: An example of a heterogeneous app usage graph.

For the sake of simplicity, in this chapter, the term 'location node' refers to 'time-enhanced location node' by default.

To distinguish different connection strengths between nodes, we model the heterogeneous app usage graph as an undirected weighted graph. There are three types of edges, including *user app* edges that reflect the usage of apps by users, *user time-enhanced location* edges that reflect the trajectories of users, and *app time-enhanced location* edges that reflect the spatiotemporal nature of app usage.

### 6.2.2 Homogeneous Relational Location Graph

As we are interested in uncovering urban dynamics, i.e., learning representations of the time-enhanced location nodes, we next derive a homogeneous relational location graph from the heterogeneous app usage graph.

We apply a meta-path based method to the time-enhanced location nodes in the heterogeneous app usage graph. A meta-path defines a compositional relation connecting two entities while still accounting for the heterogeneity and semantics of nodes and edges between those entities.

**Definition 2** ***Meta-path.*** *A meta-path $\phi$ is defined as a path generation rule on a heterogeneous graph in the form of $V_1 \rightarrow V_2 \rightarrow \cdots \rightarrow V_l$, where $V$ denotes node types. In other words, a meta-path $\phi$ describes a composite connection relation between nodes of node types $V_1$ and $V_l$.*

**Definition 3** ***Meta-path reachable nodes.*** *Given a meta-path $\phi$ and a node $v$, the meta-path reachable nodes $\mathcal{N}_v^{\phi}$ of node $v$ are a set of nodes connected with node $v$ through a path in the generated path set $P_\phi$ based on meta-path $\phi$.*

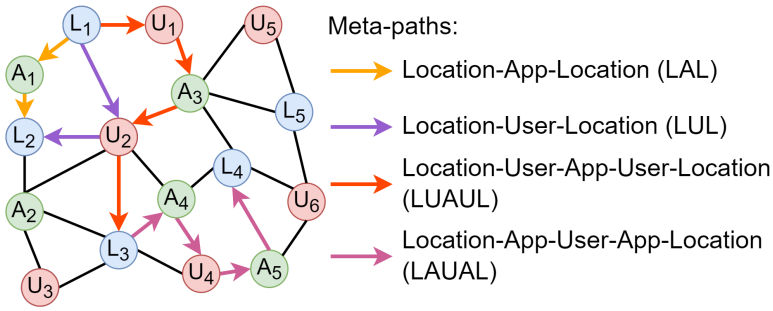



Figure 6.3: An example of several meta-paths in an app usage graph.

The key idea behind a meta-path is to generate a set of paths through the heterogeneous graph based on a semantic-aware relation. For example, the meta-path of *Location-User-Location* (abbreviated as 'LUL') enables the system to start with a given location node and find other location nodes visited by the same user. In particular, as shown in Figure 6.3, given the meta-path 'LUL', $L_1 \rightarrow U_2 \rightarrow L_2$ is an entity in the generated path set and $L_1$ and $L_2$ are meta-path reachable based on the meta-path 'LUL'. Based on different meta-paths, the meta-path reachable connections reveal different semantic relations of nodes by exploiting the structural information in the heterogeneous graph.

Next, we employ the meta-path reachable connections to construct a homogeneous relational location graph, while retaining the structural information of the heterogeneous app usage graph. Specifically, there are two steps, *i*) path set generation, *ii*) relational connection construction.

1) **Path set generation**. Given a meta-path $\phi$, in this step, we generate a set of node paths $P_\phi$ guided by this meta-path. By using multiple meta-paths $\phi_1, \phi_2, ..., \phi_n$, we can generate corresponding path sets $P_{\phi_1}, P_{\phi_2}, ..., P_{\phi_n}$ where $n$ is the number of meta-paths. Each path set has a semantic meaning and represents a specific structure in the heterogeneous graph.

2) **Relational connection construction**. Given multiple path sets $P_{\phi_1}, P_{\phi_2}$, $..., P_{\phi_n}$ generated in step (1), in this step, we determine node connections in the location graph. Specifically, for each path set $P_\phi$, we build a meta-path guided location graph where location nodes will be connected if they are meta-path reachable. An example is depicted in Figure 6.4. For meta-paths 'LAL', 'LUL', 'LUAUL', and 'LAUAL', we construct four meta-path guided homogeneous location graphs that correspond to those meta-paths, respectively. For different location graphs, their edges reflect different semantic meanings and relations. As

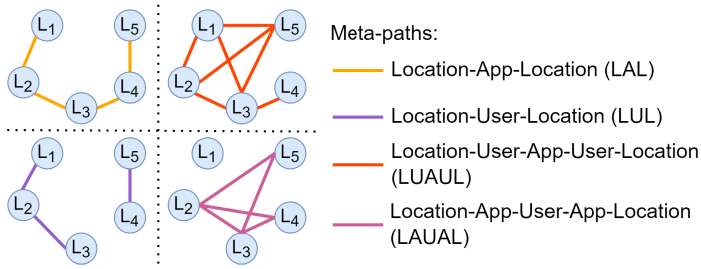



Figure 6.4: The corresponding meta-path guided location graphs of the heterogeneous app usage graph from Figure 6.2.

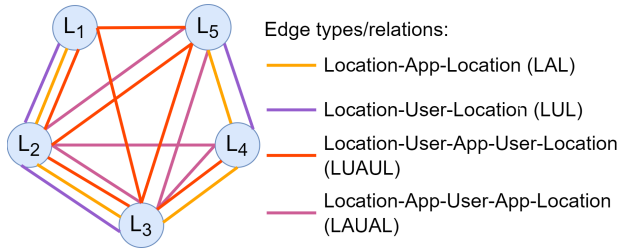



Figure 6.5: The corresponding relational location graph containing all graph structures of the meta-path guided location graphs from Figure 6.4.

all the location graphs have the same node sets, i.e., the set of time-enhanced location nodes, we can merge them using a relational graph to distinguish edges with different semantic meanings. An example is shown in Figure 6.5 in which we construct a corresponding relational location graph that contains all connection structures of the meta-path guided location graphs in Figure 6.4. Specifically, we distinguish different types of edges by different colors.

Using the above steps, we can derive a homogeneous relational location graph from the heterogeneous app usage graph. In particular, we denote the relational location graph as $G_{hom} = (L, E_\Phi, \Phi, H)$, where $L$ and $E_\Phi$ denote the sets of time-enhanced location nodes and relational edges, respectively. $\Phi$ is the set of relation types (i.e., meta-paths), and $H$ is the set of node features.

### 6.2.3 Node Features

In order to leverage the POI data of regions, we assign a feature vector $\boldsymbol{h}$ to each time-enhanced location node $l \in L$ in the homogeneous relational location graph $G_{hom}$. Specifically, the time-enhanced location node features contain two parts: static components $\boldsymbol{h_s}$ and dynamic components $\boldsymbol{h_d}$, where $\boldsymbol{h} = [\boldsymbol{h_s}, \boldsymbol{h_d}]$.

**Static components $h_s$.** We use the density of nearby Point-of-Interests (POIs) to represent the static components of location node features. POI data depict various venues located in the region, such as shopping malls, theaters, parks, and office buildings. Thus, nearby POIs describe the inherent characteristics of that region.

**Dynamic components $h_d$.** We use the human mobility flows in a region within a time slot to represent the dynamic components of location node features. In particular, human mobility flows describe people's arrive-stay-leave behavior. In detail, people arrive at a specific region and stay for a certain period, and then leave that region. Many previous studies have shown that human mobility flows within a region reflect that region's dynamic characteristics [70]. Specifically, areas with similar flow patterns have similar functions. We note that since human mobility flows in a region change over time in a day, the mobility flow features are dynamic for individual regions.

### 6.2.4 Auto-Encoder for Relational Location Graph

The information in the relational location graph $G_{hom} = (L, E_\Phi, \Phi, H)$ is contained in both the network structure and node features. Expressly, the node features represent the internal characteristics of time-enhanced locations, while the network structure depicts their relationships. Next, we aim to learn a numerical representation for each time-enhanced location node by simultaneously considering both node features and network structure.

Specifically, we utilize a deep auto-encoder framework for learning time-enhanced location embeddings. An auto-encoder is an unsupervised neural network model, which consists of two parts: a graph encoder and a graph decoder. The whole architecture of the framework is shown in Figure 6.6. The encoder projects the original node feature matrix $H$ to a hidden representation $Z$. While the decoder attempts to reconstruct the node feature matrix $H'$ from the generated hidden representation $Z$. The auto-encoder framework aims to guarantee that the reconstructed node feature matrix $H'$ is as similar to the original feature matrix $H$ as possible. Also, in order to introduce network structure information into the hidden representation $Z$, both graph encoder and decoder characterize node features over the relational location graph $G_{hom}$ by using relational graph attention networks (i.e., Rel-GAT), which enhance the graph attention network to relation-specific operations.

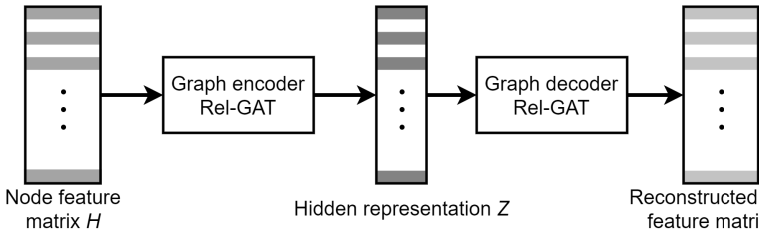



Figure 6.6: The architecture of the graph auto-encoder. The graph encoder and decoder capture node features from the relational location graph by using relational graph attention networks, i.e., Rel-GAT.

## 6.3 Experiment

We evaluate our proposed model through a set of experiments conducted on the cellular datasets. Specifically, we test our model on three applications: identifying static land usage, revealing dynamic region functions, and predicting economic levels of districts.

### 6.3.1 Baselines

We compare our model with four commonly used and state-of-the-art approaches for urban exploration:

• **POI**. An intuitive approach is to represent a region using intra-region POI data. We use TF-IDF [49] to measure different POI categories' importance to a region. Specifically, each region can be represented by a $C$-dimensional vector, where $C$ is the total number of unique POI categories. This baseline only considers the static features of regions.

• **Hidden Markov model (HMM)** [62]. HMM is a state-of-the-art method for modeling urban dynamics with app usage data. For one region at a time slot, it endows a state for that region. Each region can be represented by a state sequence across all time slots. However, this baseline cannot represent or model urban dynamics precisely because of the limited number of states.

• **DeepWalk** [47]. DeepWalk extends the word2vec model to the scenario of network embedding. We employ DeepWalk on the heterogeneous app usage graph and obtain the embeddings of time-enhanced location nodes. Specifically, each region can be represented by a vector, which is the average of its embeddings in all time slots.

• **Metapath2Vec** [16]. Metapath2Vec employs meta-path based random walks to construct the heterogeneous neighborhood of nodes and then leverages

the skip-gram model to perform node embeddings. We take the heterogeneous app usage graph as input and use 'LAL', 'LUL', 'LUAUL', and 'LAUAL' as meta-path schemes. Like DeepWalk, we represent each region by using the average of its embeddings in all time slots.

• **Graph Auto-encoder**. Our proposed method. Given the heterogeneous app usage graph, we construct the corresponding relational location graph guided by meta-paths 'LAL', 'LUL', 'LUAUL', and 'LAUAL'. By feeding the relational location graph into the Rel-GAT-based graph auto-encoder, we obtain the embeddings of time-enhanced location nodes.

### 6.3.2 Identifying Static Land Usage

For the task of identifying static land usage, we perform k-means clustering on region representations to partition regions into $k$ clusters. Regions with similar static land usage should, in theory, be assigned to the same cluster. To validate identification performance, we use the official land use map of Shanghai as the ground-truth, which classifies land use into 6 categories, i.e., residence, business, industry, public infrastructure, farming and forestry, and ecological restoration area. Therefore, we partition regions into 6 clusters by using k-means and setting $k = 6$. Next, we use the following metrics to evaluate the region clustering results of our proposed method and baselines:

• **Normalized Mutual Information (NMI)**. NMI is a widely used metric to measure the purity of clustering results from an information-theoretic perspective. A higher NMI indicates that the clustering results are closer to the ground-truth.

• **Adjusted Rand Index (ARI)**. ARI is the corrected-for-chance version of the Rand index. The higher the ARI, the better the clustering performance.

• **F-score**. F-score is a measure of clustering accuracy, which is calculated from precision and recall. Specifically, the higher the F-score, the better the clustering results. The maximum F-score is 1 and minimum is 0.

The evaluation results are shown in Table 6.1. From the results, we have the following key observations: 1) Graph Auto-encoder performs the best among all methods by a large margin. Compared with the best baseline, Graph Auto-encoder shows an improvement of 18.73%, 25.62%, and 19.44%, in terms of NMI, ARI, and F-score, respectively. 2) The network embedding methods, including DeepWalk and Metapath2Vec, show better performance than the POI method, implying that mobile app usage data are more informative for region profiling compared with POI distribution. 3) HMM shows the best performance among all baselines. The main reason is that HMM jointly uses mobile app usage and human

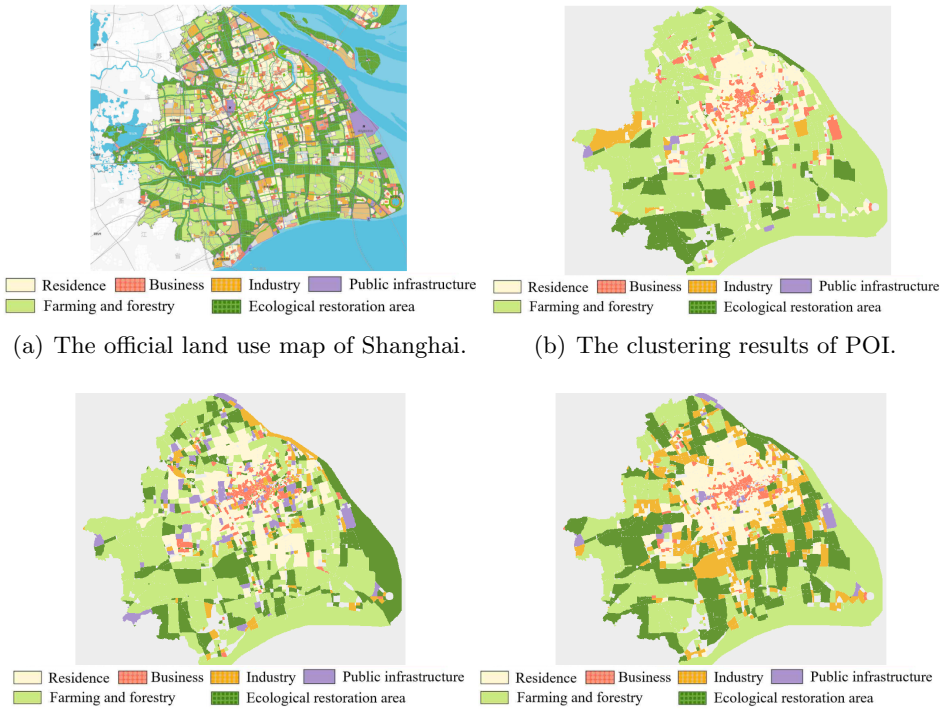
Table 6.1: Performance of Graph Auto-encoder (our proposed model) and baseline methods for static land use identification. NMI refers to normalized mutual information, ARI refers for adjusted rand index, and Imp. refers to improvement.

| Model | NMI | Imp. on NMI | ARI | Imp. on ARI | F-score | Imp. on F-score |
|---|---|---|---|---|---|---|
| POI | 0.3359 | 103.22% | 0.2926 | 122.52% | 0.3505 | 120.14% |
| DeepWalk | 0.4459 | 53.08% | 0.3937 | 65.38% | 0.4971 | 55.22% |
| Metapath2Vec | 0.5121 | 33.29% | 0.4332 | 50.30% | 0.5673 | 36.01% |
| HMM | 0.5749 | **18.73**% | 0.5183 | **25.62**% | 0.6460 | **19.44**% |
| Graph Auto-encoder | **0.6826** | - | **0.6511** | - | **0.7716** | - |

mobility flows as region features. However, compared with Graph Auto-encoder, HMM only leverages individual region features and neglects interactions between regions, which leads to performance degradation.

In order understand the clustering differences in depth, we select the models of POI, HMM, and Graph Auto-encoder and visualize the clustering results in Figure 6.7, where color denotes regions in the same cluster. We notice that using POI distributions can identify the central business area (red) and residence area (yellow). An important reason is that the POI categories of residence, restaurant, shopping mall, and corporation & business are popular and have sufficient records. On the other hand, since the other POI categories are less common, the land-use types like public infrastructure and industry can not be easily identified. Although we use TF-IDF to mitigate this uneven distribution of POI data, the model still does not perform well compared with other baselines. Alternatively, through leveraging mobile app usage data, HMM can differentiate the public infrastructure areas (purple), e.g., the airport. Moreover, as HMM also leverages human mobility flow patterns, HMM has the ability to recognize the farming and forestry area (light green) and ecological restoration area (dark green) to some extent. In terms of Figure 6.7(d), we observe that our proposed model, Graph Auto-encoder, can accurately identify all six land-use types. The main reason is that apart from intra-region features, Graph Auto-encoder also builds a relational location graph to leverage various inter-relations among regions.

Through the land use identification task we demonstrate the effectiveness of learned embeddings in representing region properties. More importantly, we obtain six anchor embeddings, i.e., centroids of the six clusters, representing the six types of region functions. Specifically, we use $z_R$, $z_B$, $z_I$, $z_P$, $z_F$, and




(a) The official land use map of Shanghai.

(b) The clustering results of POI.

(c) The clustering results of HMM.

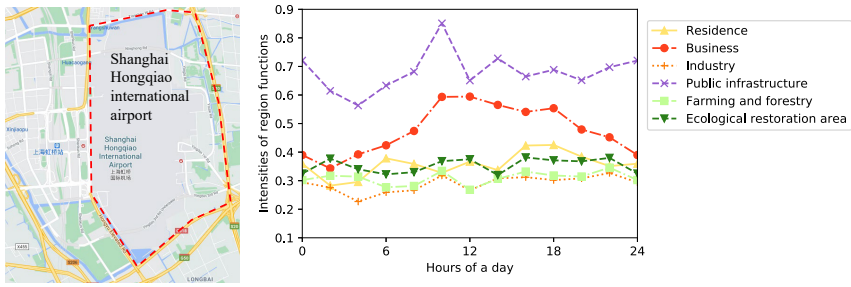
(d) The clustering results of Graph Auto-encoder (our proposed model).

Figure 6.7: The official land use map of Shanghai and region clustering results of POI, HMM, and Graph Auto-encoder. Each cluster is denoted by a unique color.

$z_E$ to denote the anchor embeddings of residence, business, industry, public infrastructure, farming and forestry, and ecological restoration area, respectively.

### 6.3.3 Revealing Dynamic Region Functions

We next aim to investigate the changes in region functions throughout the day. In particular, for a region $i$ at time slot $t$, we measure its region functions by computing the cosine similarity between its embedding $z_{l_i^t}$ and the anchor embeddings. Given a region, we reveal its dynamic region functions by depicting how the region's intensities of the six region function types change over the course of a day. In our case, we only show our analysis of three regions.




(a) The map of Region A.

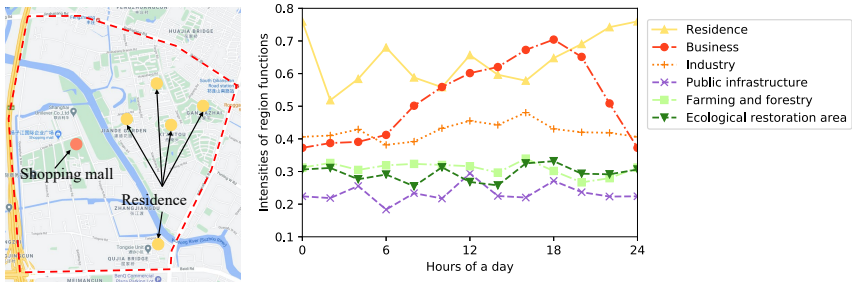
(b) The dynamic region functions of Region A.

Figure 6.8: The map of Region A and the intensities of the six region function types, including residence, business, public infrastructure, farming and forestry, and ecological restoration area.

**Region A.** First, we take Shanghai Hongqiao international airport as an example to analyze how its intensities of the six region function types change throughout the day. As shown in Figure 6.8(b), Region A, i.e., the international airport, has a higher intensity of public infrastructure function compared with other function types. This corresponds to the official land use map, marking the airport as public infrastructure. Moreover, we detect that Region A has a business function during the daytime, which might be due to duty-free shops and restaurants located in the airport.

**Region B.** The map of Region B and its dynamic region functions are depicted in Figure 6.9. Specifically, the area of Region B is denoted by a red dotted polygon in Figure 6.9(a). In the official land use map, Region B is classified as a residence type. We can observe that there are five major residential areas in Region B, denoted by yellow dots. Also, in terms of Figure 6.9(b), Region B has a high intensity of residence function throughout the day, corresponding to the official land use map classification. Moreover, we still notice that the intensity of residence function of Region B fluctuates over the day, peaking at night with valleys at around 11.00 and 16.00. One possible reason is the working rhythm of people. When people leave home and go to work, the intensity of the residence function is weakened due to population decrease.

Meanwhile, a large shopping mall is located in Region B, indicated by a red dot, which causes Region B to exhibit a business function to some extent. As depicted in Figure 6.9(b), the intensity of business function rises during the




(a) The map of Region B.

(b) The dynamic region functions of Region B.

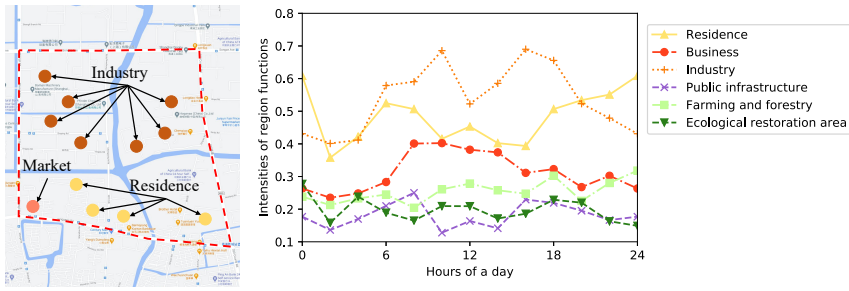
Figure 6.9: The map of Region B and the intensities of the six region function types, including residence, business, public infrastructure, farming and forestry, and ecological restoration area.

daytime and reaches a peak at around 18.00. Between 14.00 and 18.00, the intensity of business function overrides the residence function, which indicates that the most significant region function changes from residential to business.

**Region C.** We depict the map of Region C and its dynamic region functions in Figure 6.10. In the official land use map, Region C is marked as industrial. Nevertheless, according to Figure 6.10(a), Region C is a mosaic consisting of seven industry areas (marked by brown dots), four residence areas (marked by yellow dots), and one market (marked bya red dot). From Figure 6.10(b), we can observe that the industry function, as the most significant function type, has the highest intensity during the daytime, from 6.00 to 20.00. Alternatively, after 20.00, the residence function becomes the dominating function type. Again, the main reason is likely daily working rhythms. Moreover, a market is located in Region C, which gives the region a business function. However, compared with residence and industry, the business function intensity is weak.

### 6.3.4 Predicting Economic Levels of Districts

Naturally, an area's urban functions are highly related to the area's economic development. In this section, we aim to predict districts' economic levels by using the dynamic functions as input features. In practice, we use GDP data as a measure of economic development for each district. From the Shanghai Economy Almanac (2017) [1], we obtain the official GDP data of the 188 administrative districts of Shanghai, ranging from 21.75 to 671.11 and with an average of




(a) The map of Region C.

(b) The dynamic region functions of Region C.

Figure 6.10: The map of Region C and the intensities of the six region function types, including residence, business, public infrastructure, farming and forestry, and ecological restoration area.

Table 6.2: Performance of several classifiers using district-level dynamic function features for economic (GDP) level prediction.

| Method | Precision | Accuracy | F-score |
|---|---|---|---|
| Logistic regression | 0.6655 | 0.8157 | 0.7330 |
| Support vector machine | 0.7780 | 0.4211 | 0.5215 |
| Random forest | 0.8616 | 0.8421 | 0.8265 |

142.66[1]. Next, we discretize the GPD data into three levels, i.e., $[21.75, 42.66)$, $[42.66, 242.66)$, and $[242.66, 671.11]$.

Since one administrative district contains multiple regions, we represent its intensities of dynamic functions by averaging all regions in that district. We conduct a 5-fold cross-validation experiment using three popular classifiers, i.e., logistic regression, support vector machine, and random forest, to predict district economic levels. Table 6.2 presents the classification performance in terms of precision, accuracy, and F-score. We can observe that random forest achieves the best performance with an F-score of 0.8265, outperforming the linear classifiers, i.e., logistic regression and support vector machine. Also, such a high F-score and accuracy illustrates the strong correlations between dynamic functions and the economic development of administrative districts.

[1]The unit is 100 million RMB.

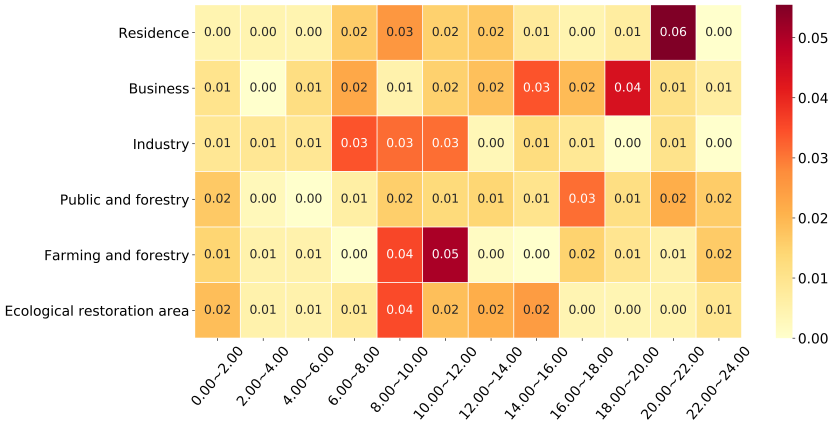



Figure 6.11: The importance of different dynamic function features for economic (GDP) level prediction of districts.

We also explore the importance of dynamic functions for predicting economic level by computing the mean decrease impurity (MDI) of all input features when using the random forest model. The higher the MDI, the more important the feature. Figure 6.11 shows the MDI score of six function types across different time slots for predicting the economic level. We can observe that the same function type has different importances at different time slots, thus validating the use of dynamic functions. Specifically, the intensities of residence and business functions have higher importance in the evening, i.e., between 18.00 and 22.00. While, the intensities of industry, farming and forestry, and ecological restoration functions are of higher importance in the morning. Such differences might be caused by human flow interactions across different functional areas throughout the day.

## 6.4 Chapter Summary

We conducted a series of experiments based on the Cellular dataset, including static land usage identification, dynamic region functions analysis, and economic (GDP) level prediction. The experimental results demonstrate the superiority and effectiveness of our framework. The study brings a new angle to urban analytics by leveraging mobile app usage data and lights the way for further urban-related applications, including urban planning, urban dynamic modeling, and economic analyses.

# Chapter 7

# Conclusions and Future Work

This thesis provides techniques for analyzing mobile usage in the wild and reveals behavioral patterns from three different domains, i.e., user, time, and location. The studies demonstrate that mobile big data generated from mobile devices can be a real benefit in revealing user features and location functions by mining digital spatial and temporal behavioral patterns. In this chapter, we conclude this thesis by revisiting the research questions provided at the beginning of this thesis. Furthermore, we present some interesting further research directions.

## 7.1 Summary

We first revisit the research questions and summarize the corresponding answers as follows:

RQ1. **What activities can be discovered from anonymized mobile app usage data?**

Based on a city-scale mobile app usage dataset, We have discovered a set of seven dominant activities and provided these seven activities with meaningful labels based on their temporal patterns and the semantic information of app categories. The discovered activities are commute and transportation, entertainment, shopping, socializing, reading and checking, life and health, and exploring food options.

RQ2. **What common patterns we share with others in our daily activities?**

We have examined the existence of daily activity patterns for individuals and found a set of five common regular activity patterns for different groups of people. More specifically, the afternoon reading pattern for senior citizens, the nightly entertainment pattern for the younger generation, the pervasive socializing pattern for socially active people, the commuting pattern for white-collar workers, and the nightly socializing for freelancers.

RQ3. **Does the outbreak of Covid-19 affect users' smartphone usage, and how?**

Based on a mobile usage dataset collected from North America, we discover that users' smartphone usage indeed changes across the outbreak of Covid-19. The outbreak of Covid-19 causes a decrease in users' smartphone engagement in terms of both CPU usage and memory usage. Also, the outbreak of Covid-19 makes an increase in WiFi usage and a decrease in network switches, implying that users reduce their mobility intensity.

RQ4. **Can we use smartphone usage data to infer the outbreak stages of Covid-19?**

We examine the Pearson correlations between smartphone usage and daily confirmed cases of Covid-19. The results reveal that memory usage, WiFi usage, and network switches of smartphones have significant correlations, whose absolute values of Pearson coefficients are greater than 0.8. By conducting the inference task, we demonstrate that using smartphone usage data to infer the outbreak stages can achieve Macro-F1 and Micro-F1 of over 0.8.

RQ5. **What are the longitudinal evolution patterns of mobile app usage behavior?**

Based on a long-term mobile app usage dataset covering 1,465 users from 2012 to 2017, we discover the longitudinal evolution patterns of app categories and apps exhibit different processes. A complete usage evolution of an app-category undergoes two stages, i.e., a growth stage and a plateau stage. However, apart from the above two stages, apps have one more additional stage, i.e., an elimination stage. The diversity of app-category usage declines over time and then keeps stable, reflecting app categories used by different users tend to be consistent. However, the diversity of app

usage increases greatly, showing large differences between mobile users at the app level.

RQ6. **Is it possible to use mobile app usage data to reveal urban land usage patterns?**

We propose a graph-based representation learning framework that reveals dynamic regional functions using mobile app usage behavior. By applying our framework to a city-scale mobile app usage dataset, we can successfully identify all types of land usage patterns. The learned dynamic location embeddings can be used to illustrate how regional functions change throughout the day and predict regional economic level (GDP). We show the significant potential of mobile app usage data in urban analytic.

## 7.2 Future Work

This thesis has presented several studies for mining behavior patterns from mobile big data. However, there are still multifold potential extensions of this thesis in both theoretical and technical directions. Below we present some of them:

- Privacy preservation is always an important aspect when accessing, using, and sharing mobile app usage data from individuals. As we discussed in Chapter 3, users' attributes might be inferred from their app usage data. Thus, privacy-preserving technologies for data analysis are essential when we deal with sensitive data. A commonly used method is anonymization. However, only anonymizing user IDs is still not enough for preserving privacy. One promising analysis framework might be Federated Learning [66]. Federated Learning enables mobile smartphones to collaboratively learn a shared model while keeping all the data locally. In this way, users do not need to upload their usage data to cloud servers, and their sensitive data can be well protected. These mechanisms will protect user privacy to some extent and further mitigate user privacy concerns.

- Context-aware spatiotemporal mobile app usage modeling is an important but challenging problem. Solving this problem will make essential breakthroughs in mobile app usage pattern discovery, app usage prediction, and app recommendation. However, as demonstrated in Chapter 3 and Chapter 6, app usage changes with different contexts, e.g., location and time, making it hard to model such complicated and dynamic behavior. One

research direction would be about developing a unified framework to model spatiotemporal app usage behavior by considering the inter-dependencies of mobile app usage, location, and time, as well as the effect of user characteristics.

- Investigating the behavior evolution process across different countries is also a further step. In Chapter 5, we revealed the worldwide evolution patterns of mobile app usage behavior. However, different countries might have different patterns. On the one hand, in the past decades, globalization accelerated interaction and integration among people from different countries and cultural backgrounds [4]. Such interaction and integration also affect users' app usage behavior. Apps are more comfortable to spread around the world and attract a large number of users. On the other hand, localization is still a trend in mobile app adoptions. Many apps support local-related information and are useful for tourists and local residents [44]. Affected by globalization and localization simultaneously, the behavior evolution patterns in different countries will be more complicated.

- Our studies leverage statistical data analysis and machine learning algorithms to explore interdependencies and correlations in mobile usage data. Although interdependencies and correlations are useful for prediction and classification tasks, they still lack interpretability in some cases. For example, in Chapter 4, we examined the correlation between smartphone usage behavior and Covid-19 cases. However, their causality relationship still needs further exploration, which is not enough to provide direct guidance to policymakers. Moreover, the correlation analysis might get stuck in Simpson's Paradox [43] and lead to erroneous analysis results. In this way, deep reasoning app usage behavior for users by adding causal analysis is necessary. Fortunately, in recent years, causal inference has developed significantly, which might support reasoning analysis of mobile user behavior.

# References


[1] Shanghai Economy Almanac. Development Research Center of Shanghai Municipal People's Government, 2017.

[2] Ionut Andone, Konrad Blaszkiewicz, Mark Eibes, Boris Trendafilov, Christian Montag, and Alexander Markowetz. How age and gender affect smartphone usage. In Paul Lukowicz, Antonio Krüger, Andreas Bulling, Youn-Kyung Lim, and Shwetak N. Patel, editors, *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp Adjunct 2016, Heidelberg, Germany, September 12-16, 2016*, pages 9–12. ACM, 2016.

[3] Ionut Andone, Konrad Blaszkiewicz, Mark Eibes, Boris Trendafilov, Christian Montag, and Alexander Markowetz. Menthal: a framework for mobile data collection and analysis. In Paul Lukowicz, Antonio Krüger, Andreas Bulling, Youn-Kyung Lim, and Shwetak N. Patel, editors, *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp Adjunct 2016, Heidelberg, Germany, September 12-16, 2016*, pages 624–629. ACM, 2016.

[4] Ulrich Beck. *What is globalization?* John Wiley & Sons, 2018.

[5] Matthias Böhmer, Brent J. Hecht, Johannes Schöning, Antonio Krüger, and Gernot Bauer. Falling asleep with angry birds, facebook and kindle: a large scale study on mobile application usage. In Markus Bylund, Oskar Juhlin, and Ylva Fernaeus, editors, *Proceedings of the 13th Conference on Human-Computer Interaction with Mobile Devices and Services, Mobile HCI 2011, Stockholm, Sweden, August 30 - September 2, 2011*, pages 47–56. ACM, 2011.

[6] Paolo Calciati and Alessandra Gorla. How do apps evolve in their permission requests?: A preliminary study. In Jesús M. González-Barahona, Abram

Hindle, and Lin Tan, editors, *Proceedings of the 14th International Conference on Mining Software Repositories, MSR 2017, Buenos Aires, Argentina, May 20-28, 2017*, pages 37–41. IEEE Computer Society, 2017.

[7] Bogdan Carbunar and Rahul Potharaju. A longitudinal study of the google app market. In Jian Pei, Fabrizio Silvestri, and Jie Tang, editors, *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2015, Paris, France, August 25 - 28, 2015*, pages 242–249. ACM, 2015.

[8] Min Chen, Shiwen Mao, and Yunhao Liu. Big data: A survey. *Mobile Networks and Applications*, 19(2):171–209, 2014.

[9] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.

[10] Yimin Chen, Xiaocong Jin, Jingchao Sun, Rui Zhang, and Yanchao Zhang. POWERFUL: mobile app fingerprinting via power analysis. In *Proceedings in 2017 IEEE Conference on Computer Communications, INFOCOM 2017, Atlanta, GA, USA, May 1-4, 2017*, pages 1–9. IEEE, 2017.

[11] Xiang Cheng, Luoyang Fang, Xuemin Hong, and Liuqing Yang. Exploiting mobile big data: Sources, features, and applications. *IEEE Network*, 31(1):72–79, 2017.

[12] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1724–1734. ACL, 2014.

[13] William HE Day and Herbert Edelsbrunner. Efficient algorithms for agglomerative hierarchical clustering methods. *Journal of Classification*, 1(1):7–24, 1984.

[14] Jonas Dehning, Johannes Zierenberg, F Paul Spitzner, Michael Wibral, Joao Pinheiro Neto, Michael Wilczek, and Viola Priesemann. Inferring change

points in the spread of covid-19 reveals the effectiveness of interventions. *Science*, 369(6500), 2020.

[15] Trinh Minh Tri Do, Jan Blom, and Daniel Gatica-Perez. Smartphone usage in the wild: a large-scale analysis of applications and context. In Hervé Bourlard, Thomas S. Huang, Enrique Vidal, Daniel Gatica-Perez, Louis-Philippe Morency, and Nicu Sebe, editors, *Proceedings of the 13th International Conference on Multimodal Interfaces, ICMI 2011, Alicante, Spain, November 14-18, 2011*, pages 353–360. ACM, 2011.

[16] Yuxiao Dong, Nitesh V. Chawla, and Ananthram Swami. Metapath2vec: Scalable Representation Learning for Heterogeneous Networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*, pages 135–144. ACM, 2017.

[17] Jon D Elhai, Dean McKay, Haibo Yang, Charlene Minaya, Christian Montag, and Gordon JG Asmundson. Health anxiety related to problematic smartphone use and gaming disorder severity during covid-19: Fear of missing out as a mediator. *Human Behavior and Emerging Technologies*, 3(1):137–146, 2021.

[18] Hossein Falaki, Ratul Mahajan, Srikanth Kandula, Dimitrios Lymberopoulos, Ramesh Govindan, and Deborah Estrin. Diversity in smartphone usage. In Sujata Banerjee, Srinivasan Keshav, and Alec Wolman, editors, *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services (MobiSys 2010), San Francisco, California, USA, June 15-18, 2010*, pages 179–194. ACM, 2010.

[19] Benjamin Finley and Tapio Soikkeli. Mobile device type substitution. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(1):8:1–8:20, 2018.

[20] Niloofar Ghahramani and Candace Brakewood. Trends in mobile transit information utilization: An exploratory analysis of transit app in New York City. *Journal of Public Transportation*, 19(3):9, 2016.

[21] Mitchell L. Gordon, Leon A. Gatys, Carlos Guestrin, Jeffrey P. Bigham, Andrew Trister, and Kayur Patel. App usage predicts cognitive ability in older adults. In Stephen A. Brewster, Geraldine Fitzpatrick, Anna L. Cox,

and Vassilis Kostakos, editors, *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, CHI 2019, Glasgow, Scotland, UK, May 04-09, 2019*, page 168. ACM, 2019.

[22] Eduardo Graells-Garrido, Diego Caro, Omar Miranda, Rossano Schifanella, and Oscar F. Peredo. The WWW (and an H) of mobile application usage in the city: The what, where, when, and how. In Pierre-Antoine Champin, Fabien Gandon, Mounia Lalmas, and Panagiotis G. Ipeirotis, editors, *Companion of the The Web Conference 2018, WWW 2018, Lyon , France, April 23-27, 2018*, pages 1221–1229. ACM, 2018.

[23] Bin Guo, Yi Ouyang, Tong Guo, Longbing Cao, and Zhiwen Yu. Enhancing mobile app user understanding and marketing with heterogeneous crowdsourced data: A review. *IEEE Access*, 7:68557–68571, 2019.

[24] Niels Henze and Susanne Boll. Release your app on sunday eve: finding the best time to deploy apps. In Markus Bylund, Oskar Juhlin, and Ylva Fernaeus, editors, *Proceedings of the 13th Conference on Human-Computer Interaction with Mobile Devices and Services, Mobile HCI 2011, Stockholm, Sweden, August 30 - September 2, 2011*, pages 581–586. ACM, 2011.

[25] Alexis Hiniker, Shwetak N. Patel, Tadayoshi Kohno, and Julie A. Kientz. Why would you do that? Predicting the uses and gratifications behind smartphone-usage behaviors. In Paul Lukowicz, Antonio Krüger, Andreas Bulling, Youn-Kyung Lim, and Shwetak N. Patel, editors, *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp 2016, Heidelberg, Germany, September 12-16, 2016*, pages 634–645. ACM, 2016.

[26] Jiaxin Huang, Fengli Xu, Yujun Lin, and Yong Li. On the understanding of interdependency of mobile app usage. In *14th IEEE International Conference on Mobile Ad Hoc and Sensor Systems, MASS 2017, Orlando, FL, USA, October 22-25, 2017*, pages 471–475. IEEE Computer Society, 2017.

[27] Pan Hui, Yong Li, Jorg Ott, Steve Uhlig, Bo Han, and Kun Tan. Mobile big data for urban analytics. *IEEE Communications Magazine*, 56(11):12–12, 2018.

[28] Simon L. Jones, Denzil Ferreira, Simo Hosio, Jorge Gonçalves, and Vassilis Kostakos. Revisitation analysis of smartphone app use. In Kenji Mase, Marc

Langheinrich, Daniel Gatica-Perez, Hans Gellersen, Tanzeem Choudhury, and Koji Yatani, editors, *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp 2015, Osaka, Japan, September 7-11, 2015*, pages 1197–1208. ACM, 2015.

[29] Afaq H Khan, Mohammed A Qadeer, Juned A Ansari, and Sariya Waheed. 4G as a next generation wireless network. In *Proceedings of 2009 International Conference on Future Computer and Communication*, pages 334–338. IEEE, 2009.

[30] David G Kleinbaum, K Dietz, M Gail, Mitchel Klein, and Mitchell Klein. *Logistic regression*. Springer, 2002.

[31] Tong Li, Ahmad Alhilal, Anlan Zhang, Mohammad Ashraful Hoque, Dimitris Chatzopoulos, Zhu Xiao, Yong Li, and Pan Hui. Driving Big Data: A First Look at Driving Behavior via a Large-Scale Private Car Dataset. In *Proceedings of 35th IEEE International Conference on Data Engineering Workshops, ICDE Workshops 2019, Macao, China, April 8-12, 2019*, pages 61–68. IEEE, 2019.

[32] Tong Li, Tristan Braud, Yong Li, and Pan Hui. Lifecycle-Aware Online Video Caching. *IEEE Transactions on Mobile Computing*, 20(8):2624–2636, 2021.

[33] Tong Li, Sylvia T. Kouyoumdjieva, Gunnar Karlsson, and Pan Hui. Data Collection and Node Counting by Opportunistic Communication. In *Proceedings of 2019 IFIP Networking Conference, Networking 2019, Warsaw, Poland, May 20-22, 2019*, pages 1–9. IEEE, 2019.

[34] Tong Li, Yong Li, Mohammad Ashraful Hoque, Tong Xia, Sasu Tarkoma, and Pan Hui. To What Extent We Repeat Ourselves? Discovering Daily Activity Patterns Across Mobile App Usage. *IEEE Transactions on Mobile Computing*, 21(4):1492–1507, 2022.

[35] Tong Li, Zhaoqi Yang, Yong Li, Benjamin Finley, Sasu Tarkoma, and Pan Hui. Revealing Urban Dynamic Functions with Mobile App Usage Behavior and POIs. *Submitted to IEEE Transactions on Mobile Computing*, 2021.

[36] Tong Li, Mingyang Zhang, Hancheng Cao, Yong Li, Sasu Tarkoma, and Pan Hui. "What Apps Did You Use?": Understanding the Long-term Evolution of Mobile App Usage. In Yennun Huang, Irwin King, Tie-Yan Liu, and

Maarten van Steen, editors, *Proceedings of The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, pages 66–76. ACM / IW3C2, 2020.

[37] Tong Li, Mingyang Zhang, Yong Li, Eemil Lagerspetz, Sasu Tarkoma, and Pan Hui. The Impact of Covid-19 on Smartphone Usage. *IEEE Internet of Things Journal*, 8(23):16723–16733, 2021.

[38] Benjamin F Maier and Dirk Brockmann. Effective containment explains subexponential growth in recent confirmed covid-19 cases in china. *Science*, 368(6492):742–746, 2020.

[39] Eric Malmi and Ingmar Weber. You are what apps you use: Demographic prediction based on user's apps. In *Proceedings of the Tenth International Conference on Web and Social Media, Cologne, Germany, May 17-20, 2016*, pages 635–638. AAAI Press, 2016.

[40] Alexandre De Masi and Katarzyna Wac. You're using this app for what?: A mqol living lab study. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, UbiComp/ISWC 2018 Adjunct, Singapore, October 08-12, 2018*, pages 612–617. ACM, 2018.

[41] Christian Montag, Paul Dagum, Jon D Elhai, et al. On the need for digital phenotyping to obtain insights into mental states in the covid-19 pandemic. *Digital Psychology*, 1(2):40–42, 2020.

[42] Gonzalo Navarro. A guided tour to approximate string matching. *ACM Computing Survey*, 33(1):31–88, 2001.

[43] Eric Neufeld. Simpson's paradox in artificial intelligence and in real life. *Computational Intelligence*, 11:1–10, 1995.

[44] Keiichi Ochiai, Fatina Putri, and Yusuke Fukazawa. Local app classification using deep neural network based on mobile app market data. In *Proceedings of 2019 IEEE International Conference on Pervasive Computing and Communications, PerCom, Kyoto, Japan, March 11-15, 2019*, pages 186–191. IEEE, 2019.

[45] Adam J. Oliner, Anand P. Iyer, Ion Stoica, Eemil Lagerspetz, and Sasu Tarkoma. Carat: collaborative energy diagnosis for mobile devices. In Chiara Petrioli, Landon P. Cox, and Kamin Whitehouse, editors, *Proceedings of The*

*11th ACM Conference on Embedded Network Sensor Systems, SenSys '13, Roma, Italy, November 11-15, 2013*, pages 10:1–10:14. ACM, 2013.

[46] Ella Peltonen, Eemil Lagerspetz, Jonatan Hamberg, Abhinav Mehrotra, Mirco Musolesi, Petteri Nurmi, and Sasu Tarkoma. The hidden image of mobile apps: geographic, demographic, and cultural factors in mobile usage. In Lynne Baillie and Nuria Oliver, editors, *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services, MobileHCI 2018, Barcelona, Spain, September 03-06, 2018*, pages 10:1–10:12. ACM, 2018.

[47] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: online learning of social representations. In Sofus A. Macskassy, Claudia Perlich, Jure Leskovec, Wei Wang, and Rayid Ghani, editors, *Proceedings of The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 701–710. ACM, 2014.

[48] Michal Rosen-Zvi, Thomas L. Griffiths, Mark Steyvers, and Padhraic Smyth. The author-topic model for authors and documents. In David Maxwell Chickering and Joseph Y. Halpern, editors, *Proceedings of the 20th Conference in Uncertainty in Artificial Intelligence, Banff, Canada, July 7-11, 2004*, pages 487–494. AUAI Press, 2004.

[49] Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[50] Suranga Seneviratne, Aruna Seneviratne, Prasant Mohapatra, and Anirban Mahanti. Your installed apps reveal your gender and more! In Ersin Uzun and Mohamed Ali Kâafar, editors, *Proceedings of the ACM MobiCom Workshop on Security and Privacy in Mobile Environments, SPME@MobiCom 2014, Maui, Hawaii, USA, September 11, 2014*, pages 1–6. ACM, 2014.

[51] M. Zubair Shafiq, Lusheng Ji, Alex X. Liu, Jeffrey Pang, and Jia Wang. Geospatial and temporal dynamics of application usage in cellular data networks. *IEEE Transactions on Mobile Computing*, 14(7):1369–1381, 2015.

[52] Alain Shema and Daniel E. Acuña. Show me your app usage and I will tell who your close friends are: Predicting user's context from simple cellphone activity. In Gloria Mark, Susan R. Fussell, Cliff Lampe, M. C. Schraefel, Juan Pablo

Hourcade, Caroline Appert, and Daniel Wigdor, editors, *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, Denver, CO, USA, May 06-11, 2017, Extended Abstracts*, pages 2929–2935. ACM, 2017.

[53] Xin Su, Dafang Zhang, Wenjia Li, and Xiaofei Wang. Androgenerator: An automated and configurable android app network traffic generation system. *Security and Communication Networks*, 8(18):4273–4288, 2015.

[54] Johan AK Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.

[55] Vincent F. Taylor and Ivan Martinovic. To update or not to update: Insights from a two-year study of android app evolution. In Ramesh Karri, Ozgur Sinanoglu, Ahmad-Reza Sadeghi, and Xun Yi, editors, *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, AsiaCCS 2017, Abu Dhabi, United Arab Emirates, April 2-6, 2017*, pages 45–57. ACM, 2017.

[56] Jay J Van Bavel, Katherine Baicker, Paulo S Boggio, Valerio Capraro, Aleksandra Cichocka, Mina Cikara, Molly J Crockett, Alia J Crum, Karen M Douglas, James N Druckman, et al. Using social and behavioural science to support covid-19 pandemic response. *Nature Human Behaviour*, 4(5):460–471, 2020.

[57] Haoyu Wang, Hao Li, and Yao Guo. Understanding the evolution of mobile app ecosystems: A longitudinal measurement study of google play. In Ling Liu, Ryen W. White, Amin Mantrach, Fabrizio Silvestri, Julian J. McAuley, Ricardo Baeza-Yates, and Leila Zia, editors, *Proceedings of The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, pages 1988–1999. ACM, 2019.

[58] Huandong Wang, Fengli Xu, Yong Li, Pengyu Zhang, and Depeng Jin. Understanding mobile traffic patterns of large scale cellular towers in urban environment. In Kenjiro Cho, Kensuke Fukuda, Vivek S. Pai, and Neil Spring, editors, *Proceedings of the 2015 ACM Internet Measurement Conference, IMC 2015, Tokyo, Japan, October 28-30, 2015*, pages 225–238. ACM, 2015.

[59] Qinglong Wang, Amir Yahyavi, Bettina Kemme, and Wenbo He. I know what you did on your smartphone: Inferring app usage over encrypted data

traffic. In *Proceedings of 2015 IEEE Conference on Communications and Network Security, CNS 2015, Florence, Italy, September 28-30, 2015*, pages 433–441. IEEE, 2015.

[60] Rui Wang, Fanglin Chen, Zhenyu Chen, Tianxing Li, Gabriella M. Harari, Stefanie Tignor, Xia Zhou, Dror Ben-Zeev, and Andrew T. Campbell. Studentlife: assessing mental health, academic performance and behavioral trends of college students using smartphones. In A. J. Brush, Adrian Friday, Julie A. Kientz, James Scott, and Junehwa Song, editors, *Proceedings of The 2014 ACM Conference on Ubiquitous Computing, UbiComp '14, Seattle, WA, USA, September 13-17, 2014*, pages 3–14. ACM, 2014.

[61] Xin Wang, Shuhui Chen, and Jinshu Su. Real network traffic collection and deep learning for mobile app identification. *Wireless Communications and Mobile Computing*, 2020:4707909:1–4707909:14, 2020.

[62] Tong Xia and Yong Li. Revealing urban dynamics by learning online and offline behaviours together. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(1):30:1–30:25, 2019.

[63] Qiang Xu, Jeffrey Erman, Alexandre Gerber, Zhuoqing Morley Mao, Jeffrey Pang, and Shobha Venkataraman. Identifying diverse usage behaviors of smartphone apps. In Patrick Thiran and Walter Willinger, editors, *Proceedings of the 11th ACM SIGCOMM Internet Measurement Conference, IMC '11, Berlin, Germany, November 2-, 2011*, pages 329–344. ACM, 2011.

[64] Qiang Xu, Yong Liao, Stanislav Miskovic, Zhuoqing Morley Mao, Mario Baldi, Antonio Nucci, and Thomas Andrews. Automatic generation of mobile app signatures from traffic observations. In *Proceedings of 2015 IEEE Conference on Computer Communications, INFOCOM 2015, Kowloon, Hong Kong, April 26 - May 1, 2015*, pages 1481–1489. IEEE, 2015.

[65] Jie Yang, Yuanyuan Qiao, Xinyu Zhang, Haiyang He, Fang Liu, and Gang Cheng. Characterizing user behavior in mobile internet. *IEEE Transactions on Emerging Topics in Computing*, 3(1):95–106, 2015.

[66] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology*, 10(2):12:1–12:19, 2019.

[67] Hongyi Yao, Gyan Ranjan, Alok Tongaonkar, Yong Liao, and Zhuoqing Morley Mao. SAMPLES: self adaptive mining of persistent lexical snippets for classifying mobile application traffic. In Serge Fdida, Giovanni Pau, Sneha Kumar Kasera, and Heather Zheng, editors, *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, MobiCom 2015, Paris, France, September 7-11, 2015*, pages 439–451. ACM, 2015.

[68] Demetrios Zeinalipour-Yazti and Shonali Krishnaswamy. Mobile big data analytics: Research, practice, and opportunities. In Arkady B. Zaslavsky, Panos K. Chrysanthis, Christian Becker, Jadwiga Indulska, Mohamed F. Mokbel, Daniela Nicklas, and Chi-Yin Chow, editors, *Proceedings of IEEE 15th International Conference on Mobile Data Management, MDM 2014, Brisbane, Australia, July 14-18, 2014 - Volume 1*, pages 1–2. IEEE Computer Society, 2014.

[69] Lei Zhang, Jiangchuan Liu, Hongbo Jiang, and Yong Guan. Senstrack: Energy-efficient location tracking with smartphone sensors. *IEEE Sensors Journal*, 13(10):3775–3784, 2013.

[70] Mingyang Zhang, Tong Li, Yong Li, and Pan Hui. Multi-View Joint Graph Representation Learning for Urban Region Embedding. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 4431–4437. ijcai.org, 2020.

[71] Mingyang Zhang, Tong Li, Hongzhi Shi, Yong Li, and Pan Hui. A Decomposition Approach for Urban Anomaly Detection Across Spatiotemporal Data. In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 6043–6049. ijcai.org, 2019.

[72] Mingyang Zhang, Tong Li, Yue Yu, Yong Li, Pan Hui, and Yu Zheng. Urban Anomaly Analytics: Description, Detection and Prediction. *IEEE Transactions on Big Data*, pages 1–1, 2020.

[73] Sha Zhao, Shijian Li, Julian Ramos, Zhiling Luo, Ziwen Jiang, Anind K. Dey, and Gang Pan. User profiling from their use of smartphone applications: A survey. *Pervasive and Mobile Computing*, 59, 2019.

[74] Sha Zhao, Julian Ramos, Jianrong Tao, Ziwen Jiang, Shijian Li, Zhaohui Wu, Gang Pan, and Anind K. Dey. Discovering different kinds of smartphone

users through their application usage behaviors. In Paul Lukowicz, Antonio Krüger, Andreas Bulling, Youn-Kyung Lim, and Shwetak N. Patel, editors, *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp 2016, Heidelberg, Germany, September 12-16, 2016*, pages 498–509. ACM, 2016.

[75] Sha Zhao, Yizhi Xu, Xiaojuan Ma, Ziwen Jiang, Zhiling Luo, Shijian Li, Laurence Tianruo Yang, Anind K. Dey, and Gang Pan. Gender profiling from a single snapshot of apps installed on a smartphone: An empirical study. *IEEE Transactions on Industrial Informatics*, 16(2):1330–1342, 2020.

[76] Agustin Zuniga, Huber Flores, Eemil Lagerspetz, Petteri Nurmi, Sasu Tarkoma, Pan Hui, and Jukka Manner. Tortoise or hare? quantifying the effects of performance on mobile app retention. In Ling Liu, Ryen W. White, Amin Mantrach, Fabrizio Silvestri, Julian J. McAuley, Ricardo Baeza-Yates, and Leila Zia, editors, *Proceedings in The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, pages 2517–2528. ACM, 2019.

# Paper I

Tong Li, Yong Li, Mohammad A. Hoque, Tong Xia, Sasu Tarkoma, and Pan Hui

**To What Extent We Repeat Ourselves? Discovering Daily Activity Patterns Across Mobile App Usage**

In *IEEE Transactions on Mobile Computing*,
21(4): 1492-1507, 2022.

# To What Extent We Repeat Ourselves? Discovering Daily Activity Patterns Across Mobile App Usage

Tong Li, *Student Member, IEEE,* Yong Li, *Senior Member, IEEE,*
Mohammad A. Hoque, Tong Xia, Sasu Tarkoma, *Senior Member, IEEE,* and Pan Hui, *Fellow, IEEE*

**Abstract**—With the prevalence of smartphones, people have left abundant behavior records in cyberspace. Discovering and understanding individuals' cyber activities can provide useful implications for policymakers, service providers, and app developers. In this paper, we propose a framework to discover daily cyber activity patterns across people's mobile app usage. The framework first segments app usage traces into short time windows and then applies a probabilistic topic model to infer users' cyber activities in each window. By constructing and exploring the coherence of users' activity sequences, the framework can identify individuals' daily patterns. Next, the framework uses a hierarchical clustering algorithm to recognize the common patterns across diverse groups of individuals. We apply the framework on a large-scale and real-world dataset, consisting of 653,092 users with 971,818,946 usage records of 2,000 popular mobile apps. Our analysis shows that people usually follow yesterday's activity patterns, but the patterns tend to deviate as the time-lapse increases. We also discover five common daily cyber activity patterns, including afternoon reading, nightly entertainment, pervasive socializing, commuting, and nightly socializing. Our findings have profound implications on identifying the demographics of users and their lifestyles, habits, service requirements, and further detecting other disrupting trends such as working overtime and addiction to the game and social media.

**Index Terms**—Mobile app usage, activity pattern, pattern discovery, clustering.

## 1 Introduction

Individuals exhibit diverse activities both in physical and cyberspace. Discovering and understanding patterns of such activities are fundamental to promote and support healthier lifestyles for individuals and further improve people's well-being. Although prior research has made significant progress towards understanding activity patterns, they only focus on small groups or specific activities. With limited data and controlled studies, it is only possible to characterize the patterns of an individual or a few individuals. In most cases, the investigated groups are either narrowly focused or so diverse that it is difficult to find their common activity patterns. The common patterns among large-scale individuals may not only hint at their demographics and lifestyles but also expose disrupting trends in our society. However, this requires exploring the activity patterns of a large-scale population, e.g., millions of people.

- *T. Li and P. Hui are with the System and Media Laboratory (SyMLab), Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong. They are also with the Department of Computer Science, University of Helsinki, Helsinki, Finland. E-mail: t.li@connect.ust.hk, panhui@cse.ust.hk*
- *Y. Li and T. Xia are with Beijing National Research Center for Information Science and Technology (BNRist), Department of Electronic Engineering, Tsinghua University, Beijing, 100084, China. E-mail: liyong07@tsinghua.edu.cn, xia-t17@mails.tsinghua.edu.cn*
- *M. A. Hoque and S. Tarkoma are with the Department of Computer Science, University of Helsinki, Helsinki, Finland. E-mail: mohammad.a.hoque@helsinki.fi, sasu.tarkoma@helsinki.fi*

*Manuscript received 12 June 2019; revised 8 Aug. 2020; accepted 31 Aug. 2020.*

The difficulties in accessing data from many individuals are one of the main obstacles in large-scale pattern studies. Traditional data collection methods, like surveys and questionnaires, need a lot of human effort, which is usually inefficient and delayed. In recent years, the multifaceted usage of smartphones in daily lives has established them as a necessity, which records various users' cyber activities. In addition to traditional uses, e.g., communication and web browsing, people use smartphones in more complex activities such as ordering food, shopping online, and managing health [1]. A great number of research studies in recent years have applied app usage data to investigate user behavior [2]–[5]. These studies include app energy drain [6], app signatures [3], how app usage varies with different kinds of users [4], and how unusual events disrupt the app engagement [5].

In this work, we focus on discovering daily activity patterns in cyberspace from large-scale app usage data. More specifically, we study the following research problems.

- What activities can be discovered from app usage data?
- What common patterns we share with others in our daily cyber activities?

There are three challenges to address these two problems. **1)** The complexity of mobile app usage behavior. App usage behavior varies across app categories, usage time, and users. Due to the unbalanced usage of different categories, some useful and critical app usage, like ordering food, may be overwhelmed by popular categories and cannot be detected. Besides, the same app usage may imply different

activities for different users in different contexts. Hence, how to extract user activity features from such complicated data is fundamental but difficult. **2)** The lack of ground truth labels of user cyber activities. All large-scale app usage datasets lack meaningful labels to map user activities to app usage records. Finding the appropriate activity labels is challenging, and annotating the extracted cyber activities with semantic terms is also tricky. **3)** The complexity of user activities in cyberspace. Activity patterns can be diverse in both individual and group granularity. For individuals, her/his cyber activity patterns may change over days. How to identify a typical daily cyber activity pattern for each individual is complicated. Also, whether there are any common patterns across individuals and discovering these common patterns is another challenge.

To overcome the above challenges, we propose a framework to discover daily activity patterns in cyberspace by leveraging the following three key designs. **First**, we devise a probabilistic topic model based method to identify the activity features of app usage windows obtained using a time-based segmentation approach. The proposed method regards a window as a document, a user as an author, an activity as a topic, and app usage logs related to the window as words. In this way, the probabilistic topic model can characterize the relationships between users, app usage, and cyber activities in a cohesive manner. **Second**, to annotate extracted cyber activities with semantic terms, we explore the semantic information of app categories and prior knowledge of activity temporal patterns. More specifically, we compute an average app category feature vector for each extracted activity. In terms of the density in the calculated vector, we rank app categories in an activity termed as internal ranking. Besides, we also rank all activities for each app category termed as external ranking. By jointly considering internal ranking, external ranking, and temporal distribution of activities, we determine each extracted cyber activity with a meaningful label. **Third**, we look into the regularity of individuals' cyber activity patterns by measuring their average intra-distance of patterns across different days. We then identify a daily cyber activity pattern for every individual. By applying the agglomerative hierarchical clustering with edit distance, we recognize the common patterns across diverse groups, which address the third challenge.

The contributions of our work can be summarized as follows.

- We investigate the problem of discovering daily activity patterns in cyberspace of a large-scale population by using their mobile app usage data. We propose a novel activity discovery framework based on a probabilistic topic model, which can characterize the relationships between users, apps, and activities in a cohesive manner.
- We apply our framework on a large-scale and real-world app usage dataset. We discover people prefer to imitate yesterday's activity patterns and provide evidence that the population follows five common patterns in their daily cyber activities, including afternoon reading (8.50%), nightly entertainment (18.45%), pervasive socializing (7.56%), commuting (29.29%), and nightly socializing (36.20%).
- We verify our findings via a small-scale dataset with users' occupation information. We find that different common patterns correspond to different demographic groups. More specifically, freelancers have a nightly socializing pattern. White-collar workers have a commuting pattern. Advertisers and socializers have a pattern of pervasive socializing. Also, we infer that senior citizens are mostly involved in afternoon reading, and the younger generation is primarily involved in nightly entertainment.
- Based on the discovered cyber activity patterns, we detect several social issues, e.g., around 35% of workers always work overtime, and nearly 42% of the younger generation is addicted to gaming applications. We further explore the implications of our framework and findings for policymakers and government, researchers, service providers, and app developers.

The rest of this paper is organized as follows. In section 2, we present an overview of our dataset. In section 3, we present how to discover activities from app usage traces. In section 4, we elaborate on the schemes of identifying the common daily patterns. We then discuss the implications and limitations of our work in section 5. Related works are presented in section 6. Finally, we briefly conclude the paper in section 7.

## 2 Dataset Overview

To understand the daily activity patterns of people in a large city, we explore a city-scale app usage dataset provided by a primary Internet Service Provider (ISP) in China. The dataset was collected during one week in April 2016, covering the whole metropolitan area of Shanghai, one of the world's largest cities. The dataset includes over 2 million users and their app usage records during the data collection period. The app usage dataset is characterized by the ISP with an anonymized user ID, timestamp, and app ID.

In detail, the ISP identified app usage traces based on users' network access records collected from gateways, generated when users issue network connection requests. In terms of the data format, each network access record contains a user ID, timestamp, and the connection's metadata. To determine the corresponding app ID for each network access record, the ISP inspected the HTTP head and used the destination domain and user-agent as the app identifier. By adopting a systematic tool, SAMPLE [7], the ISP constructed conjunctive rules to match specific apps. SAMPLE applies a supervised learning algorithm over a small set of labeled data streams to automatically generate the conjunctive rules, which can identify over 90% of apps with an average accuracy of 99% [7]. In practice, the ISP built the conjunctive rules by manually operating a small set of apps to generate data streams. They then crawled the 2,000 most popular apps across app stores and matched network traffic records to these apps. Also, the ISP manually verified the correctness of the matched apps. In terms of the statistics from the ISP, more than 95% of network traffic used HTTP at the time of data collection, and they could map up to 90% of the network traffic to specific apps. During data collection, although some apps used HTTPS for critical

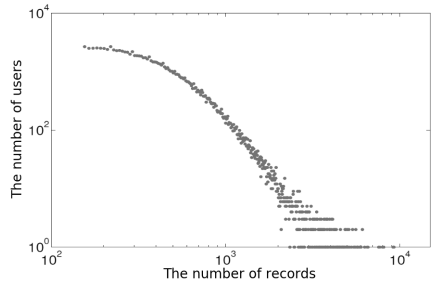
TABLE 1
The numuber of apps and usage records for each app category.

| No. | Category | # of apps | # of usage records | No. | Category | # of apps | # of usage records |
|---|---|---|---|---|---|---|---|
| 1 | Game | 342 | 44,553,941 | 13 | Photography | 26 | 2,082,726 |
| 2 | Finance | 27 | 22,421,634 | 14 | Lifestyle | 79 | 55,949,873 |
| 3 | Stock | 106 | 24,724,632 | 15 | Health & fitness | 73 | 4,616,153 |
| 4 | Shopping | 152 | 48,973,459 | 16 | Sports | 33 | 7,385,860 |
| 5 | Parent & child | 34 | 631,729 | 17 | News | 86 | 79,210,799 |
| 6 | Education | 67 | 7,135,838 | 18 | Reading | 98 | 17,182,331 |
| 7 | Weather | 34 | 4,181,546 | 19 | Media & video | 105 | 45,994,839 |
| 8 | Travel | 66 | 5,353,283 | 20 | Music & audio | 86 | 70,127,620 |
| 9 | Navigation | 68 | 89,816,498 | 21 | Business | 59 | 5,116,745 |
| 10 | Transportation | 78 | 39,915,957 | 22 | House & home | 26 | 1,412,869 |
| 11 | SON & IM | 185 | 827,784,029 | 23 | Car | 39 | 4,156,542 |
| 12 | Food & drink | 53 | 69,783,067 | 24 | Tools & others | 78 | 14,337,945 |

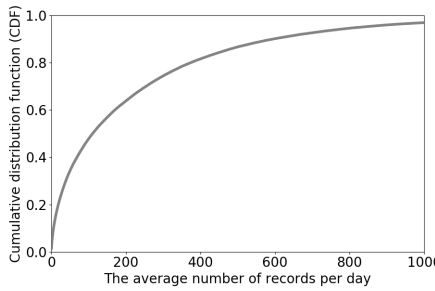
functions, e.g., log-in, most parts of their traffic still used HTTP. Also, we notice most apps use HTTPS in recent years. Some existing studies [8], [9] have demonstrated that app usage traces can also be identified from encrypted data traffic. Overall, the app usage dataset provided by the ISP, although not covering all traffic, is sufficient for our analysis of mainstreams of usage behavior modeling.

Next, we classify the 2,000 most popular apps into different categories. An app category has an inherent semantic meaning, and apps in the same category are more likely to be involved in a similar cyber activity [5]. As our data is collected from network operators, the dataset includes both Android and iOS users. Due to the difference in app categorization systems of Apple Store (iOS apps of 25 categories) and Google Play (Android apps of 30 categories), we cannot directly adopt the category information of app stores. For some apps, they may belong to different categories in different app stores. For example, Firefox belongs to utilities in Apple Store while it is in the communication category in Google Play. Therefore, we re-categorize apps. We first crawl app descriptions from app stores and generate an app-description matrix using Jieba [10]. With the app-description matrix, we apply the Latent Dirichlet Allocation [11] model to cluster apps and extract their category information. To determine the optimal categories, we gradually vary it from 20 to 30 and manually check the coherence of the outputs. We find that 24 categories are optimal and with the minimum perplexity. We then count the apps and usage records for each app category and show them in Table 1. It is worth noting that, in practice, some apps may fall into multiple categories. For example, the specific category of YouTube may vary depending on the content viewed by users. It belongs to the news category when users watch news videos while belonging to the education category when users watch education videos. However, due to the privacy restrictions, we cannot obtain the information of the content accessed by users. Therefore, in this work, we only consider a hard categorization scheme, where one app solely belongs to one major category. For example, YouTube belongs to the category of media and video in terms of its app description.

Since our analysis focuses on the collected 2,000 most popular apps, we first filter out the outlier users who do not have any matched app usage. After the filtering, the remaining dataset contains 1,699,386 users and 1,492,849,915 usage records. Fig. 1 shows the statistics of the dataset without outlier users. In Fig. 1(a), we find that the records




(a)




(b)

Fig. 1. This figure shows (a) the records for each unique user. (b) The cumulative distribution function of the average number of records per day.

of each unique user still follows the power-law distribution, which means that the app usage behavior still has the long tail pattern even when we consider only 2,000 apps. Further, we solely keep those users who were always active during the data collection period. In terms of the statistics, more than 90% of users have average daily records exceeding five, as shown in Fig. 1(b). Hence, we define the active day for each user is the one with at least five app usage records. After these two-step filtering operations, we finally obtain 971,818,946 valid app usage records and 653,092 unique active users. Table 2 presents a summary of the filtered app usage dataset.

Meanwhile, we are very aware of the privacy implications of using the dataset for research and our research findings. We and the ISP have taken adequate measures

TABLE 2
Dataset summary.

| # of Records | # of Users | # of Identified Apps | # of App Categories |
|---|---|---|---|
| 971,818,946 | 653,092 | 2,000 | 24 |

to safeguard the privacy of mobile users according to [12]. **First**, the ISP has the consent to collect mobile data, and stripped all the personally identifiable information from the traces and enforced strict non-disclosure agreements for all researchers. The dataset is stored in a server protected by authentication mechanisms and firewalls in the ISP network. **Second**, the ISP only gave us the anonymized user IDs. We never had access to their actual identifiers. Also, we did not have location information of users. The discovered pattern of an individual alone does not leak the privacy, as it cannot be associated with the user's actual ID. **Finally**, this work has received approval from both the ISP and the authors' local institute.

## 3 Discovery of User Activities

In this section, we identify users' cyber activities based on their app usage traces logged over a whole week. For each user, we first divide her/his app usage traces into several small time windows to capture short-term activities. By using a probabilistic topic model, we obtain the activity label for each window. We then determine the semantic terms of each cyber activity in terms of internal ranking and external ranking of app categories. Finally, we recognize seven unique activities across large-scale app usage data.

### 3.1 App Usage Trace Representation

In this paper, we assume that users' app usage records reflect their cyber activities. To capture users' short-term activities, we first divide app usage traces of each day into multiple small time windows. In practice, we use a time-based segmentation mechanism. Considering the diurnal pattern of app usage [13], we make a trade-off between non-uniform time grids and uniform time grids. In our case, a window refers to an app usage block composed of app usage records during a specific time period.

More specifically, we first look into the cumulative distributions of window size, the number of app usage records for different candidate time grids, as shown in Fig. 2. With half an hour and one-hour time grids, 20% of windows have less than 15 records, which are too short of capturing user activities. On the other hand, setting time grids as four and six hours, the windows become too large where multiple activities will be mixed. Hence, in terms of Fig. 2, the time grid of two hours strikes a balance between having enough app usage records within each window and having more stationary windows during the usual active hours of the day, i.e., 5.00 to 21.00.

Moreover, app usage has a typical diurnal pattern, as shown in Fig. 3. The app usage sharply declines after 21.00, and there is very limited app usage during the usual inactivity period, i.e., 0.00 - 5.00. This is expected, as people usually begin to take rest and sleep during these hours. To get a more stable size for windows, similar to [14], we aggregate

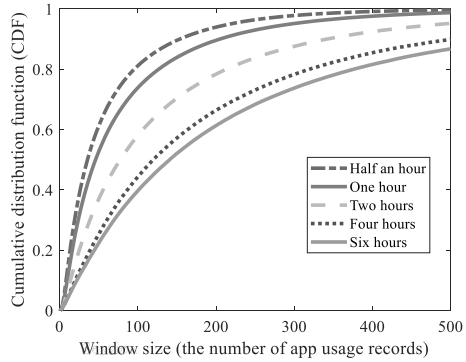



Fig. 2. Quantity of windows with respect to the number of app usage records.

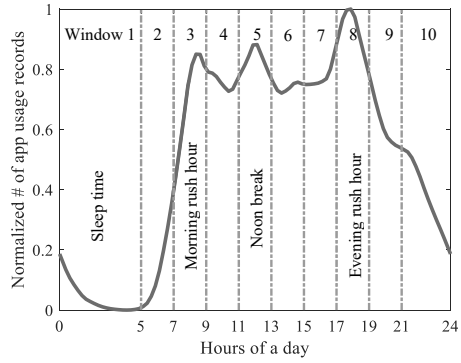



Fig. 3. Min-max normalized number of app usage records during one day.

the windows of low app usage during these two periods. In the end, we divide app usage traces of each day for each user into ten windows. Therefore, in our dataset, each user has 70 windows, i.e., 7 (# of days) $\times$ 10 (# of windows for each day). Also, we can apply the usual notions to some time slots, such as morning rush hour (7.00 to 9.00), noon break (11.00 to 13.00), and evening rush hour (17.00 to 19.00). After segmentation, we get 45,716,440 windows for the 653,092 unique active users. Among all these windows, only 9,196,661 windows are unique.

### 3.2 Activity Discovery

To characterize the activities of windows, we explore the power of the author-topic model [15]. As a probabilistic topic model, the author-topic model has been successfully used for discovering the hidden topic structure in large documents [15]–[17]. In this model, each document exhibits multiple topic features. Each word of a document supports topics in probability, and the authors of the document determine the mixture weights for different topics as well. Given all words and authors of each document as observations, the author-topic model is trained to infer the hidden topic of each document.

In the author-topic model, the generative model for documents is depicted in Fig. 4. There is one latent variable $z$ for topics. Assuming that there are $D$ documents in the

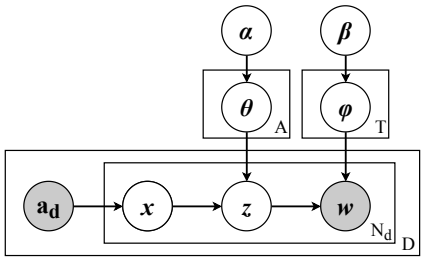



Fig. 4. Generative model for documents. Observations are represented by shadowed nodes.

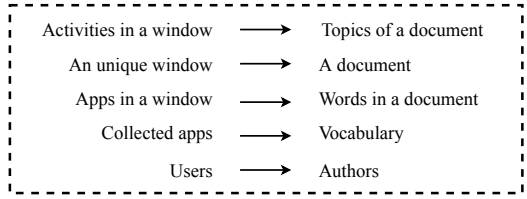



Fig. 5. Analogy between window-activities to document-topics.

corpus, document $d$ is a sequence of $N_d$ words and has a group of authors $\boldsymbol{a_d}$. $x$ indicates the author responsible for a given word $w_d^i$ and chosen from $\boldsymbol{a_d}$. Let $\alpha$ and $\beta$ be the prior parameters for the Dirichlet author-topic distribution and topic-word distribution. $\varphi$ denotes a $T \times V$ matrix of topic distributions, with a multinomial distribution over $V$ vocabulary items, i.e., all words in $D$ documents for each of $T$ topics. Each $\varphi_t$ is drawn independently from the symmetric Dirichlet($\beta$) prior. $\theta_a$ stands for the topic proportion for author $a$. $z$ denotes the topic responsible for generating the word $w$. By using the above notations, we illustrated the generative process as follows.

1) For each author in the $A$ authors, choose the vector of topic proportions $\theta_a \sim$ Dirichlet($\alpha$).
2) For each topic in the $T$ topics, choose the vector of word proportions $\varphi_t \sim$ Dirichlet($\beta$).
3) For each of the word $w_d^i$ in document $d$,
   - choose an author $x \sim$ Uniform($\boldsymbol{a_d}$);
   - choose a topic $z \sim$ Multinomial($\theta_a$);
   - choose a word $w_d^i \sim$ Multinomial($\varphi_z$).

Dirichlet($\cdot$) means the Dirichlet distribution. Uniform($\cdot$) represents the uniform distribution, and Multinomial($\cdot$) is the multinomial distribution. The exact inference of the hidden variables is computationally intractable. Therefore, in practice, approximate inference algorithms are commonly used, such as Laplace approximation [18], Variational approximation [19], and Markov chain Monte Carlo (MCMC) [20]. In our work, we use the Variational Bayes inference algorithm [21], a variant of Variational approximation method.

We aim to find the hidden cyber activity structure of app usage windows for the problem of activity discovery. Specifically, each window is a block of app usage traces, represented as a sequence of app IDs. Each window has multiple activity features, and each app usage of a window supports hidden activities in probability. Hence, the relationships among activities, apps, and windows, are highly similar to the relationships among topics, words, and documents. By considering they also have similar objectives, we build an analogy between the activity discovery of windows and the topic discovery of documents. As shown in Fig. 5, a unique window represents a document, the users of a window represent the authors of a document, and an activity represents a topic. A window has multiple activity features, which is just like a document that has various topics. Apps in one window are deemed as words in a document. The vocabulary is the set of all words in documents, while the collected apps are the set of all apps in windows. Thus, the collected apps are regarded as vocabulary. In practice, the collected apps stand for the app IDs of the 2,000 most popular apps.

Specifically, we applied the author-topic model to extract hidden activity features of app usage windows due to the following three reasons. 1). As shown in Fig. 2, app usage windows vary from 20 records to 500 records. Traditional clustering algorithms, like K-means and hierarchical clustering, have difficulty dealing with inputs of different sizes. Alternatively, the author-topic model has no input size limitation and has proven to perform well for both short texts [22], e.g., twitters, and long texts [16], e.g., articles. 2). As for traditional clustering algorithms, we need to empirically and manually extract features from app usage windows to identify activities. Nevertheless, the author-topic model can characterize the relationships between users, app usage, and cyber activities cohesively and automatically identify activities of app usage windows. 3). The same app usage may imply different activities in different contexts. As a probabilistic topic model, the author-topic model enables app usage to support multiple hidden activities in probability, which solves the semantic ambiguity of app usage.

Like the other topic models [23], [24], the author-topic model requires us to specify the number of topics in advance. Although determining the most appropriate number of topics remains an open issue, we can evaluate the model by measuring how perplexity scores vary with the number of topics. Perplexity is a measurement of how well a probability distribution or probability model predicts samples [25]. Perplexity is a common metric for estimating topic model performance [26]. Mathematically, the perplexity of a set of words, $W_d$ for document $d$, is defined as follows,

$$\text{Perplexity}(W_d) = \exp\left[-\frac{\ln P(W_d)}{N_d}\right]. \quad (1)$$

Intuitively, perplexity stands for the confusion of the model about its decision. More accurately, perplexity expresses the average number of words that have to be picked to get a correct one, when we randomly choose words from the probability distribution calculated by the author-topic model at each time step [27].

To determine the most appropriate number of topics, we vary it ranging from 2 to 30 and compute perplexity. Fig. 6 shows perplexity relative to the number of topics. The lower value of perplexity implies the better performance of the model. We then find the perplexity score is lowest when the number of topics is 12, while the knee of the curve is where the number of topics is 6. Empirically, the knee of the curve is better. Subsequently, we obtain the cyber activity features

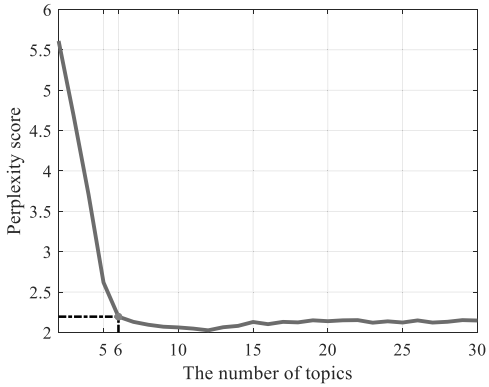



Fig. 6. Perplexity score versus the number of topics. The lower value of perplexity implies the better performance of the model. The knee of the curve is denoted as the red point where the number of topics is 6.

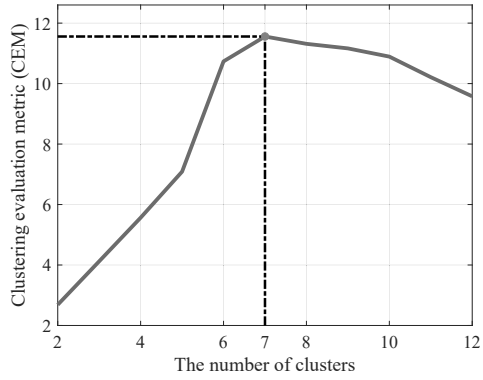



Fig. 7. Clustering evaluation metric versus the number of clusters. The higher the value implies the better the clustering. The optimal clustering is performed with 7 clusters.

of each window by applying the author-topic model and setting the predefined number of topics as 6.

### 3.3 Window Aggregation

To facilitate finding the semantic terms of each window, we take a further step, aggregating similar unique windows in terms of their activity features. Windows from the same cluster have similar activity features, and different clusters represent different activities. For an arbitrary window $s$, its activity feature is a 6-dimensional vector, i.e., $\theta_s = (\theta_{s,1},\ \theta_{s,2},\ ...,\ \theta_{s,6})$, where $\theta_{s,t}$ is the proportion of activity $t$ for window $s$.

We perform the Bisecting K-means clustering algorithm on the activity feature vectors of the 9,196,661 unique windows. Compared with other clustering algorithms, such as hierarchical clustering and spectral clustering, only k-means and its variants can handle such large scale windows. Different from basic K-means, Bisecting K-means as a variant that can overcome the problem of getting caught in a local minimum by minimizing the Sum of Squared Errors (SSE) of split clusters [28], [29]. Karypis *et al.* showed that Bisecting K-means outperforms basic K-means in terms of entropy, F measure, and overall similarity [30].

The number of clusters for the Bisecting K-means algorithm needs to be predefined according to the application and determined by measuring the quality of clustering [31]. To evaluate the quality of clustering for different numbers of clusters, $K$, we defined a clustering evaluation metric (CEM),

$$\text{CEM} = \frac{SP}{CP}, \tag{2}$$

where $SP$ and $CP$ indicate the average separation and compactness of clusters, respectively.

$$\begin{aligned} SP &= \frac{2}{K^2 - K} \sum_{i=1}^{K-1} \sum_{j=i+1}^{K} |c_i - c_j|, \\ CP &= \frac{1}{K} \sum_{i=1}^{K} \sum_{\theta \in \Omega_i} \frac{|\theta - c_i|}{|\Omega_i|}, \end{aligned} \tag{3}$$

where $\Omega_i$ is the set of members in cluster $i$, $c_i$ is the centroid of cluster $i$. $SP$ stands for the average inter-cluster distance. A small value for $SP$ indicates adjacent clusters are similar. Thus, the larger $SP$, the higher the quality of clustering. On the other hand, $CP$ stands for the average intra-cluster distance. A small value for $CP$ means the less scattering of clusters. As a result, the higher the CEM, the better the quality of clustering. To determine the most appropriate number of clusters, we run the Bisecting K-means algorithm on the Window-Activity matrix and vary the number of clusters ranging from 2 to 12 to compute CEM. Fig. 7 shows how CEM changes with the number of clusters. We then find the optimal number of clusters is 7.

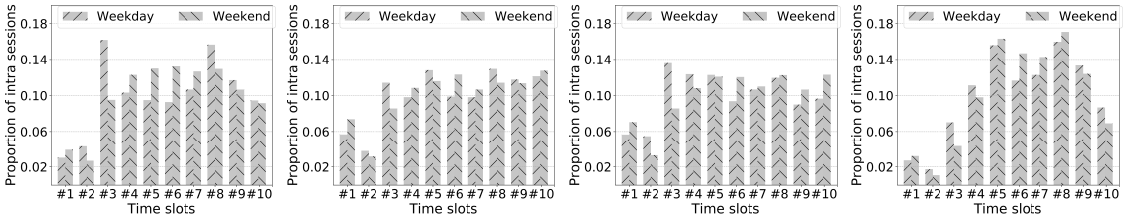
### 3.4 Activity Identification

After window aggregation, we obtain seven clusters. Windows in the same cluster have similar activities. In this step, we aim to identify each window cluster with semantic terms, i.e., activity labels. Note that activity identification is a very challenging problem. Unlike small scale datasets [32], the large-scale app usage datasets lack meaningful labels mapping user cyber activities to app usage records. Fortunately, we can explore the semantic information of app categories and prior knowledge of activity temporal patterns to identify activity labels. For example, if a user uses food & drink apps during lunchtime, it has a high probability of identifying the activity as exploring food.

In detail, we identify the cyber activity label of a window cluster by considering the following three aspects. 1) The app category configuration in a window cluster. We compute the average proportion of different app categories for each window cluster. According to the calculated proportion, we rank app categories in a window cluster, called *Internal Ranking (IR)*. 2) The window cluster configuration across different app categories. We also rank the window clusters for each app category, called *External Ranking (ER)*. 3) The temporal distribution of windows in each cluster. In Table 3, the top-3 internal app categories for each window cluster are colored by blue from darkest to lightest, while the top window cluster for each app category is colored by red. The seven typical window clusters are identified as follows.

TABLE 3
Proportion of app categories for window clusters and corresponding internal and external rankings. C: *cluster*, Pro: *proportion*, IR: *internal ranking*.

| | Commute & transportation (C1) | | Entertainment (C2) | | Shopping (C3) | | Socializing (C4) | | Reading & checking (C5) | | Life & health (C6) | | Exploring food (C7) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **App categories** | **Pro** | **IR** | **Pro** | **IR** | **Pro** | **IR** | **Pro** | **IR** | **Pro** | **IR** | **Pro** | **IR** | **Pro** | **IR** |
| Game | 0.0109 | 9 | 0.3435 | 2 | 0.0083 | 16 | 0.0093 | 5 | 0.0127 | 18 | 0.0076 | 10 | 0.0997 | 2 |
| Finance | 0.0063 | 13 | 0.0017 | 21 | 0.0127 | 12 | 0.0056 | 10 | 0.0087 | 19 | 0.0074 | 11 | 0.0254 | 6 |
| Stock | 0.0027 | 18 | 0.0059 | 12 | 0.0130 | 11 | 0.0024 | 14 | 0.0642 | 5 | 0.0028 | 17 | 0.0047 | 17 |
| Shopping | 0.0127 | 8 | 0.0077 | 11 | 0.1223 | 3 | 0.0141 | 3 | 0.0197 | 10 | 0.0171 | 5 | 0.0191 | 7 |
| Parent & child | 0.0009 | 24 | 0.0107 | 8 | 0.0093 | 15 | 0.0008 | 23 | 0.0049 | 24 | 0.0015 | 20 | 0.0049 | 16 |
| Education | 0.0026 | 19 | 0.0005 | 24 | 0.0048 | 22 | 0.0022 | 15 | 0.0151 | 13 | 0.0020 | 19 | 0.0015 | 23 |
| Weather | 0.0022 | 20 | 0.0017 | 22 | 0.0049 | 20 | 0.0016 | 18 | 0.0084 | 20 | 0.0014 | 21 | 0.0023 | 22 |
| Travel | 0.0016 | 22 | 0.0023 | 19 | 0.0049 | 21 | 0.0009 | 21 | 0.0133 | 17 | 0.0009 | 24 | 0.0038 | 18 |
| Navigation | 0.1812 | 3 | 0.0039 | 15 | 0.0387 | 5 | 0.0085 | 6 | 0.0715 | 4 | 0.0138 | 7 | 0.0346 | 5 |
| Transportation | 0.1322 | 4 | 0.0129 | 7 | 0.0269 | 7 | 0.0036 | 12 | 0.0199 | 9 | 0.0049 | 14 | 0.0148 | 8 |
| SON & IM | 0.2226 | 2 | 0.0551 | 5 | 0.4335 | 1 | 0.8294 | 1 | 0.1410 | 2 | 0.1141 | 2 | 0.0976 | 3 |
| Food & drink | 0.0480 | 5 | 0.0729 | 3 | 0.1374 | 2 | 0.0765 | 2 | 0.0432 | 6 | 0.0224 | 4 | 0.5825 | 1 |
| Photography | 0.0062 | 14 | 0.0041 | 14 | 0.0071 | 18 | 0.0016 | 16 | 0.0064 | 23 | 0.0010 | 23 | 0.0416 | 4 |
| Lifestyle | 0.0080 | 12 | 0.0032 | 17 | 0.0484 | 4 | 0.0060 | 7 | 0.0150 | 14 | 0.6962 | 1 | 0.0113 | 9 |
| Health & fitness | 0.0031 | 17 | 0.0098 | 9 | 0.0050 | 19 | 0.0016 | 19 | 0.0140 | 16 | 0.0478 | 3 | 0.0052 | 14 |
| Sports | 0.0017 | 21 | 0.0026 | 18 | 0.0073 | 17 | 0.0009 | 20 | 0.0406 | 7 | 0.0142 | 6 | 0.0027 | 21 |
| News | 0.0056 | 15 | 0.0052 | 13 | 0.0237 | 8 | 0.0056 | 9 | 0.3280 | 1 | 0.0038 | 16 | 0.0051 | 15 |
| Reading | 0.0046 | 16 | 0.0082 | 10 | 0.0098 | 14 | 0.0016 | 17 | 0.0878 | 3 | 0.0064 | 13 | 0.0071 | 13 |
| Media & video | 0.0103 | 10 | 0.3632 | 1 | 0.0342 | 6 | 0.0052 | 11 | 0.0236 | 8 | 0.0069 | 12 | 0.0076 | 12 |
| Music & audio | 0.2693 | 1 | 0.0579 | 4 | 0.0138 | 10 | 0.0134 | 4 | 0.0173 | 11 | 0.0090 | 9 | 0.0103 | 11 |
| Business | 0.0302 | 6 | 0.0208 | 6 | 0.0108 | 13 | 0.0028 | 13 | 0.0149 | 15 | 0.0104 | 8 | 0.0103 | 10 |
| House & home | 0.0097 | 11 | 0.0035 | 16 | 0.0040 | 24 | 0.0008 | 22 | 0.0076 | 21 | 0.0024 | 18 | 0.0034 | 19 |
| Car | 0.0013 | 23 | 0.0010 | 23 | 0.0043 | 23 | 0.0005 | 24 | 0.0151 | 12 | 0.0014 | 22 | 0.0010 | 24 |
| Tools & others | 0.0260 | 7 | 0.0017 | 20 | 0.0149 | 9 | 0.0058 | 8 | 0.0071 | 22 | 0.0046 | 15 | 0.0033 | 20 |




(a) Commute and transportation (b) Entertainment (c) Life and health (d) Exploring food

Fig. 8. The temporal distribution of windows in clusters (C1) *Commute and Transportation*, (C2) *Entertainment*, (C6) *Life and health*, and (C7) *Exploring food*.

(C1) *Commute and Transportation*. The top-3 app categories in this window cluster are *Music & audio*, *SON & IM*, and *Navigation*. This cluster contains the maximum number of *Transportation* apps as well. We also investigate the distribution of these windows in the temporal domain. As shown in Fig. 8(a), on weekdays, the windows in this cluster mainly happen in the time slots 3 and 8, i.e., morning rush hour and evening rush hour. The specific time slot division scheme is presented in section 3.1. On weekdays, the number of windows occurring in the evening is higher than that in the afternoon, while it is contrary to weekends. This phenomenon corresponds to the fact that on weekdays people, especially white-collar workers, are 'bound' in workplaces during the afternoon and free from work on weekends. According to the app category features of this cluster, we also infer that people prefer to listen to music and check emails during commute time. It is not surprising that navigation apps are highly used during transportation activities as well.

(C2) *Entertainment*. This cluster contains typical entertainment activities with the highest both internal ranking and external ranking of *Media & video* and *Game*, as shown in Table 3. Fig. 8(b) shows the temporal features of the windows in this cluster. On weekdays, there are three time slots of the highest proportion of windows, namely time slots 3 (morning rush hour), 5 (noon break), and 8 (evening rush hour). These three time slots are the main 'free time' for people on weekdays. On the other hand, on weekends, the time slots in the evening account for the highest proportion. Especially for the time slot 1, from 0.00 to 5.00, the proportion on weekends is much larger than weekdays, because people do not need to get up early and go to work on weekends.

(C3) *Shopping*. This cluster mainly has shopping activities with the most *shopping* apps. There is no significant difference in its temporal features on weekdays and weekends. Like the general pattern of user behavior, the proportion increases during the day and decreases during the night. In the windows of this cluster, the socializing apps, i.e., *SON & IM* category, account for the highest proportion in the internal ranking. We infer that people usually browse the products in online shops and share them with friends via

socializing apps to ask for suggestions and comments.

(C4) *Socializing*. This cluster is identified as the social activity since not only the *SON & IM* category has the highest internal ranking and external ranking but also the other app categories are of lower proportions compared with other clusters (see Table 3). In other words, the windows in this cluster are 'pure' social states. Its temporal features are similar to the *Shopping* activity, peaking during the daytime while reducing over the night period.

(C5) *Reading and checking*. The most characteristic app categories in this window cluster are *News*, *Reading*, *Sport*, *Weather*, *Stock*, and *Education*, with a significant higher proportion than other window clusters. These app categories are all about reading and checking activities. It seems that people are habituated to using these kinds of apps in the same period. Besides, its temporal features are similar to the *Shopping* activity as well.

(C6) *Life and health*. In this window cluster, the typical app category features are *Lifestyle* and *Health & fitness*, which are with both high external ranking and internal ranking as presented in Table 3. The temporal distribution of this cluster's windows is shown in Fig. 8(c). It is clear that, on weekdays, the windows are concentrated in the morning, especially in the time slot 3, from 7.00 to 9.00. In contrast, on weekends, time slot 10, from 21.00 to 24.00, accounts for the most significant part of windows. Since apps of *Health & fitness* category are mainly used to record users' exercises such as jogging, swimming, and keep-fit, we infer that people prefer to take workouts in the morning on weekdays while in the evening on weekends.

(C7) *Exploring food*. Table 3 shows that the windows in this cluster are of the maximum proportion of *Food & drink* app usage both in external ranking and internal ranking. We then look into its temporal features, as shown in Fig. 8(d). We observe that both on weekdays and weekends, time slot 5 (lunchtime, from 11.00 to 13.00) and time slot 8 (dinner time, from 17.00 to 19.00) take up the largest part of windows. Also, *SON & IM* and *Game* are with a high ranking in this cluster. We guess this is because people share restaurant localization with friends via social apps and like playing mobile games while waiting for dishes. We also notice *Finance* in this cluster is with a significantly higher proportion than other clusters, which suggests that people pay their orders by electronic payment methods like Paypal.

## 4 Discovery of Activity Patterns

Activity patterns are of significant value for both individuals and society. For individuals, service providers can provide personalized service by exploring users' lifestyles, habits, occupations, and socio-economic status from their activity patterns. For society, the government can understand people's living status and detect disrupting trends from activity patterns and then make policies to improve people's well-being. Specifically, in this work, we sequence users' cyber activities[1] identified from their app usage traces and apply sequence analysis methods to extract activity patterns.

1. Apart from discovered seven activities, we add an Unknown label to denote silent time slots.

### 4.1 Similarity Measurement of Activity Sequences

After identifying users' activities, the next goal is to use the collected one-week individuals' behavior to determine daily cyber activity patterns. To do so, we treat users' app usage traces of one day as an incidence of sequential activities. Apart from the discovered seven activities, we add an *Unknown* label for those silent time slots during which no app usage records are observed and denote it as $C8$. For each user, each day app usage traces reveal a cyber activity sequence of length ten that contains combinations of the eight general activities including *Commute and transportation*, *Entertainment*, *Shopping*, *Socializing*, *Reading and checking*, *Life and health*, *Exploring food*, and *Unknown*. Hence, each user's activity sequence can be expressed as,

$$A_u = \{[a_1^{d_1}, a_2^{d_1}, ..., a_{10}^{d_1}], ..., [a_1^{d_7}, a_2^{d_7}, ..., a_{10}^{d_7}]\}, \quad (4)$$

where $A_u$ stands for the activity sequence of user $u$ and $a_m^{d_n}$ denotes the activity label of window $m$ in the $n$-$th$ day, $a \in \{C1, C2, C3, C4, C5, C6, C7, C8\}$.

To quantify the degree of similarity among activity sequences, we apply the string metric. Each user's activity sequence for one day is regarded as a string, which is a combination of eight kinds of characters, i.e., activities. Particularly, in our work, we use the Levenshtein distance metric [33] which has been widely used in the social analysis [34], information theory [35], linguistics [36], and computer science [37]. In [38], William W. C. *et al.* compared different string metrics and showed the Levenshtein distance is better than others. The Levenshtein distance is also be referred to as edit distance. Given two sequences, the Levenshtein distance is the minimum number of single-character edits, including insertions, deletions, and substitutions, required to transform one sequence into another.

Compared with another commonly used distance metric, i.e., Hamming distance, Levenshtein distance is more suitable for our measurement to capture the sequential patterns. Hamming distance does not use insertions and deletions [39]. It only uses substitutions and is only possible to compare when activities occur, while Levenshtein distance takes insertions and deletions into account and can capture the order in which user activities are organized over time. We then justify that with a practical example. We are given three sequences: 'C8C1C5C4C1C2', 'C2C8C1C5C4C1' and 'C8C2C2C2C2C2', where C denotes the discovered activities which are presented in Table 3. Since our goal is to investigate how cyber activities are sequenced throughout a day, i.e., sequential features, the distance between 'C8C1C5C4C1C2' and 'C2C8C1C5C4C1' should be smaller than the distance between 'C8C1C5C4C1C2' and 'C8C2C2C2C2C2'. Note that, for 'C8C1C5C4C1C2' and 'C2C8C1C5C4C1', there is only one time slot shift. The Hamming distance and Levenshtein distance between 'C8C1C5C4C1C2' and 'C2C8C1C5C4C1' are 6 and 2 respectively, while both Hamming distance and Levenshtein distance between 'C8C1C5C4C1C2' and 'C8C2C2C2C2C2' are 4. Hence, we apply Levenshtein distance to measure the similarity of activity sequences. Particularly, the two edits to change 'C8C1C5C4C1C2' into 'C2C8C1C5C4C1' for Levenshtein metric are shown as follows,

1) C8C1C5C4C1C2 $\Rightarrow$ C8C1C5C4C1 (deletion of 'C2'),

2) C8C1C5C4C1 $\Rightarrow$ C2C8C1C5C4C1 (insertion of 'C2' at the beginning).

### 4.2 Individuals' Activity Analysis

We first investigate the similarity of different days to examine the regularity of activities in days' scale. As shown in Table 4, we calculate the average Levenshtein distance between two days in pairwise. Given day $i$ and day $j$, their distance is computed as,

$$\frac{1}{U}\sum_{u=1}^{U}\text{lev}(\mathbf{A}_u^{d_i}, \mathbf{A}_u^{d_j}), \tag{5}$$

where $U$ is the number of unique users and $\mathbf{A}_u^{d_i}$ denotes the activity sequence of the $i$-$th$ day for user $u$.

We notice that there is an apparent difference between weekdays' sequences and weekends' sequences because the average distance between any two weekdays is less than that between any weekday and weekend. Besides, we discover an interesting appearance that the activity sequence of a weekday is more similar to yesterday's sequence. This implies that *people intentionally or unintentionally obey yesterday's activity sequence, and there should be a daily pattern of activities for individuals.*

We further look into the coherence of each user's activity sequences on weekdays. As shown in Fig. 9, we compute the average intra-distance for each user,

$$\frac{2}{D^2-D}\sum_{i=1}^{D-1}\sum_{j=i+1}^{D}\text{lev}(\mathbf{A}^{d_i}, \mathbf{A}^{d_j}), \tag{6}$$

where $D$ is the number of weekdays, here $D = 5$, and $\mathbf{A}^{d_i}$ stands for $[a_1^{d_i}, a_2^{d_i}, ..., a_{10}^{d_i}]$ namely the activity sequence of day $i$. We observe that nearly 74% of users have the average intra-distances less than 5, implying that they repeat at least half of their daily activities in cyberspace. Therefore, a large part of individuals' daily lives follows a regular daily activity pattern. We give the definition of the daily activity pattern of an individual as follows.

**Definition 1.** *Daily activity pattern of an individual. Given an individual's activity sequences on weekdays,* $\mathbf{A}^{d_1}$, $\mathbf{A}^{d_2}$, ..., $\mathbf{A}^{d_5}$, *the daily activity pattern of the individual,* $\mathbf{A}$, *has the minimum sum-distance between all pairs of* $\mathbf{A}$ *and* $\mathbf{A}^{d_i}$. *Mathematically, denoting* $\mathbf{A} = [\mathbf{a_1}, \mathbf{a_2}, ..., \mathbf{a_{10}}]$, *then*

$$\mathbf{A} \leftarrow \arg\min_{\mathbf{a}\in\{\mathbf{C1},\mathbf{C2},...,\mathbf{C8}\}}\sum_{\mathbf{i=1}}^{\mathbf{5}}\mathbf{lev}(\mathbf{A^{d_i}}, \mathbf{A}). \tag{7}$$

### 4.3 Identifying Common Activity Patterns

We have examined the existence of daily activity patterns in cyberspace for individuals. Next, we investigate whether there are common activity patterns for millions of users. To do this, we first quantify the distance between each pair of daily activity patterns for all users. Once the distance matrix is calculated, we apply the agglomerative hierarchical algorithm to identify homogeneous clusters of daily patterns. In terms of existing studies [30] conducted on labeled datasets, the agglomerative hierarchical algorithm usually has a better performance compared with bisecting K-means.

TABLE 4
Average Levenshtein distance between arbitrary two days. We round numbers to two decimals.

| | MON | TUE | WED | THU | FRI | SAT | SUN |
|---|---|---|---|---|---|---|---|
| MON | / | 4.26 | 4.37 | 4.42 | 4.58 | 5.26 | 5.18 |
| TUE | / | / | 4.37 | 4.48 | 4.59 | 5.29 | 5.21 |
| WED | / | / | / | 4.28 | 4.49 | 5.07 | 5.01 |
| THU | / | / | / | / | 4.47 | 5.27 | 5.29 |
| FRI | / | / | / | / | / | 5.27 | 5.31 |
| SAT | / | / | / | / | / | / | 4.70 |

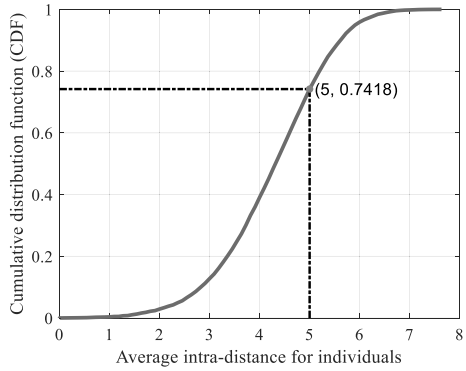



Fig. 9. Cumulative distribution of users with respect to the intra-distance. The average intra-distances of nearly 74% users are less than 5.

Also, there are 654,092 users, which satisfies the scalability limitation of hierarchical clustering algorithms. Hence, instead of bisecting K-means, we employ the agglomerative hierarchical algorithm for common pattern discovery. In detail, the agglomerative hierarchical algorithm is a 'bottom-up' approach in which each object starts in its own cluster, and the pairs of clusters are merged as one moves up the hierarchy [40].

To determine the most appropriate number of clusters, i.e., patterns, we apply the dendrogram to evaluate the agglomerative hierarchical clustering algorithm, as shown in Fig. 10. The dendrogram is a branching diagram representing the hierarchy of clusters based on the degree of similarity [41]. As Fig. 10 shows, the distance from the root to a subtree indicates the similarity of subtrees. Highly similar nodes or subtrees have joining points farther from the root. We know how the nodes are combined into larger parent clusters from the dendrogram, i.e., the detailed clustering process. In Fig. 10, we find five is the most appropriate number of clusters, where the clusters are of high intra-cluster and low inter-cluster distances. The five clusters are boxed using orange lines.

### 4.4 Pattern Annotation

Given the clustering results, we then annotate each cluster of daily activity patterns with semantic terms, which will contribute to understanding the hidden image of these patterns. We first visualize them by randomly selecting fifty users for each cluster and show how their cyber activities are sequenced. As shown in Fig. 11, the x-axis refers to the

TABLE 5
The number and proportion of users for each cluster.

| | Cluster A | Cluster B | Cluster C | Cluster D | Cluster E |
|---|---|---|---|---|---|
| Number of users | 55,513 | 120,495 | 49,374 | 191,291 | 236,419 |
| Proportion of users | 8.50% | 18.45% | 7.56% | 29.29% | 36.20% |

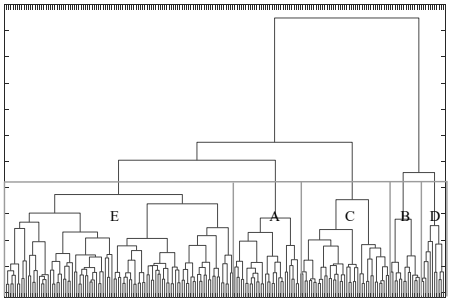



Fig. 10. Dendrogram of the hierarchical clustering where highly similar nodes or subtrees have joining points farther from the root. Five is the most appropriate number of clusters, where the clusters are of high intra-cluster and low inter-cluster distances.

windows, and the y-axis indicates the random fifty users. Each bin refers to the activity label of that window for that user, and we use different colors to distinguish different activities. The distribution of all users among clusters is shown in Table 5.

**Afternoon reading** (Cluster A). The users in this cluster are mostly involved in *Reading and Checking* during the day, as shown in Fig. 11(a). The users, on average, start to use apps from time slot 4, 9.00 to 11.00, and become to be inactive after time slot 8, around 19.00. Hence, we annotate this cluster as afternoon reading to reflect the main active periods and activity of this group. Although *Reading and Checking* activity dominates during time slots 4 to 7, there are still many users like *Shopping* during these hours. Generally, both two activities are leisure activities. We still notice that there are several *Commuting and Transportation* activities in this cluster. However, unlike Cluster D, *Commuting and Transportation* activities in this cluster are randomly distributed over time slots. Therefore, the users in this cluster do not have regular commute schedules, e.g., on and off work. Moreover, by considering the dominating pattern of reading and shopping activities, we infer the users in this group are senior citizens. Recalling the statistics in Table 5, they represent 8.5% of the total users, which is similar to the proportion of senior citizens, 10.1%, in Shanghai.

**Nightly entertainment** (Cluster B). Fig. 11(b) visualizes the daily patterns of this cluster. The users are engaged in *Entertainment* activities during evening and night, from 17.00 to 24.00. Hence, we annotate this cluster as nightly entertainment. Compared with cluster A, C, and E, the users have fewer activities during the usual active hours, i.e., from 7.00 to 15.00. Also, the users in this cluster are mostly nocturnal, and they are more than 18% of all the users. We infer that they are likely the younger generation. Due to the daytime classes, they only have free time in the evening and night, which may be why their app usage is so sparse during the day time. Besides, the daily patterns show a high number of users in this cluster sleep late, still active in the time slot 1, from 0.00 to 5.00. Most of their *Entertainment* activities last more than 6 hours, which indicates that the younger generation is addicted to the *Entertainment* activity, e.g., mobile games, and it is harmful to their health.

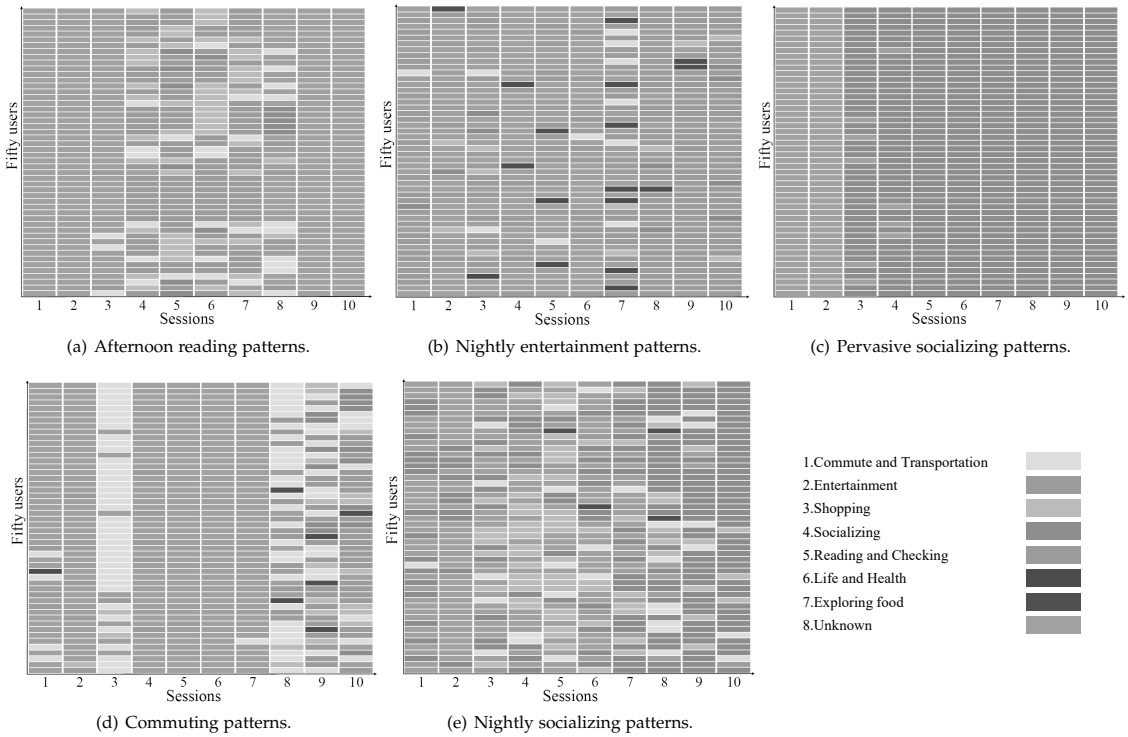
**Pervasive socializing** (Cluster C). As shown in Fig. 11(c), the users in this cluster are engaged in the *Socializing* activity from 7.00 till 24.00. Compared with the patterns of other clusters, the patterns of this cluster are more regular concerning the active time of users and the duration of the dominating activity. This unusual pattern of social activities of nearly 7.5% users can be explained as follows. With modern social networking apps, social activities are not limited only to known friends and families. Peoples are making friends and communicating with people from different social classes via social apps. The social platforms are not only for interaction but also for various businesses, such as advertising and self-media. For example, on WeChat, people advertise and sell their products via instant messages, video calls, group chats, and WeChat Moments. This method [42] utilizes the business relationship and friendship to maintain the customer relationship and is called the WeChat business. Therefore, we suspect that the users in this cluster work in call centers, customer services, or they are bloggers, cyberspace writers, and online shop owners.

**Commuting** (Cluster D). The patterns in this cluster shown in Fig. 11(d) are typical commuting patterns. The *Commuting and Transportation* activities are sequenced regularly and mainly occur during rush hours. Due to the regular commute patterns, we infer most people in this cluster are involved in white-collar jobs. Meanwhile, we find an important phenomenon. In the morning, nearly 90% of *Commute and Transportation* activities happen in the morning rush hour. However, these activities are spread over multiple time slots around the evening rush hour. This phenomenon implies that many workers cannot knock off on time, and even working overtime becomes a habitual pattern for them. Like cluster B, most of the users are with limited activities from 9.00 to 17.00, as they are busy at work and do not have time to use smartphones.

**Nightly socializing** (Cluster E). The patterns of this cluster are shown in Fig. 11(e). It is the largest cluster having 36.2% of users and the most diverse patterns as well. The dominating activity is *Socializing*, which mostly happens after the evening, i.e., after time slot 6. The users in this cluster are mostly active during the day and engaged in other activities, such as *Shopping*. This implies their time is more flexible compared to other cluster users. The lack of sufficient commuting suggests that people are mostly staying at or near home, involved in household work during the day time, and socializing in the evening. Hence, we infer the users in this cluster should be self-employed.

### 4.5 Verification through Controlled Study

To validate the detected activity patterns, we make a controlled study and collect a small-scale app usage dataset




(a) Afternoon reading patterns. (b) Nightly entertainment patterns. (c) Pervasive socializing patterns.

(d) Commuting patterns. (e) Nightly socializing patterns.

Fig. 11. Visualization of daily activity patterns by randomly selecting fifty users in each cluster. Each row represent the activity patterns for one user. The windows correspond to the time segments illustrated in Fig. 3.

TABLE 6
The average distance matrix between common daily activity patterns and occupations, where the corresponding occupations and clusters are highlighted.

| Distance: Cluster / Occupation | Afternoon Reading (Cluster A) | Nightly entertainment (Cluster B) | Pervasive socializing (Cluster C) | Commuting (Cluster D) | Nightly socializing (Cluster E) |
|---|---|---|---|---|---|
| White-collar workers | 6.6904 | 7.2043 | 7.4817 | 5.5983 | 6.6183 |
| Socializers | 7.5787 | 8.2933 | 3.5213 | 7.9347 | 5.7453 |
| Freelancers | 7.1092 | 7.2277 | 5.4939 | 6.7462 | 5.0953 |

from 100 users with occupation information. The small-scale app usage dataset is over one week, i.e., seven days and users are in the age range from 25 to 55. Users' occupations are in three categories, i.e., white-collar workers, socializers, and freelancers. Specifically, the white-collar workers include clerks, engineers, teachers, editors, lawyers, and others. The socializers consist of bloggers, advertisers, cyberspace writers, online shop owners, and others.

Applying our proposed cyber activity discovery approach, we obtain the activity sequences of that 100 users. We then compute the average distance matrix between common daily activity patterns and occupations, as shown in Table 6. Mathematically, the average distance is calculated as,

$$\frac{\sum_{u\in\mathbf{U}}\sum_{\hat{u}\in\hat{\mathbf{U}}_\mathbf{i}}\text{lev}(A_u, A_{\hat{u}})}{|\mathbf{U}|\cdot|\hat{\mathbf{U}}_\mathbf{i}|}, \tag{8}$$

where $A$ denotes the activity sequence, $\mathbf{U}$ is the set of users for each occupation category, and $\hat{\mathbf{U}}_\mathbf{i}$ is the set of users in cluster $i$.

From Table 6, we find that white-collar workers, socializers, and freelancers have the lowest average distance with cluster D, C, and E, respectively. Since white-collar workers go work and back home in the working days, it is reasonable that their activity sequences belong to the commuting pattern. Similarly, socializers are usually engaged in social-economic activities for a whole day and of the pervasive socializing pattern. As for freelancers, they have more freedom to arrange their time so that their activity pattern is the most diverse, similar to the nightly socializing pattern. Besides, due to the age bias of the small-scale dataset, we do not find the groups of small distances with cluster A (Afternoon reading) and cluster B (Nightly entertainment). This reflects the correctness of our former inferences to some extent as well.

In summary, these results are consistent with our annotation of the discovered activity pattern. These results also demonstrate the potential of our model to leverage the features of app usage in cyber activity profiling and social status inferring.

## 5 Discussion

We now revisit the research questions presented in the introductory part and present answers based on our methodology and empirical observations.

- What activities can be discovered from app usage data? *We have discovered a set of seven dominant activities and provided these seven activities with meaningful labels based on their temporal patterns and the semantic information of app categories. The discovered activities are commute and transportation, entertainment, shopping, socializing, reading and checking, life and health, and exploring food options.*
- What common patterns we share with others in our daily cyber activities? *We have examined the existence of daily activity patterns for individuals and found a set of five common regular activity patterns for different groups of people. More specifically, the afternoon reading pattern for senior citizens, the nightly entertainment pattern for the younger generation, the pervasive socializing pattern for socially active people, the commuting pattern for white-collar workers, and the nightly socializing for freelancers.*

In this study, we answered these two fundamental research questions. We also discuss how the discovery of such patterns may help to understand human behavior and further improve the quality of user experience and contribute to the well-being of people. In this section, based on our findings of both activities and daily patterns, we will discuss the implications for policymakers, researchers, service providers, and app developers.

### 5.1 Implications for Policymakers

Mobile app usage data and the insights on human behavior are relevant and vital for policymakers. The data analysis can provide valuable feedback regarding the activities and well-being of citizens. The insights are expected to contribute to better decisions on multiple scales: from cities and urban planning to the level of individual companies and environments. Traditional methods, like survey and questionnaire techniques require significant personnel resourcing and cannot address the need for timely data and insights. Thus, app data analysis provides an alternative methodology that can be near real-time and low cost.

Through smartphone app data analysis, social issues can be detected and traced at an early stage. For example, in March 2019, an 'anti-996' protest was launched via GitHub[2], followed by over 250 thousand people against working overtime.

We have analyzed working patterns and examined signs of overtime with our smartphone app dataset. We have detected that approx. 35% of workers tend to stay late at the office following a habitual pattern of working overtime. Similarly, we find that a significant proportion of the younger generation has addicted to the *Entertainment* activity. The observations indicate that policymakers and employers can use smartphone data for providing early advice and counseling when working overtime or becoming addicted to games is becoming an addiction.

2. https://996.icu/#/en_US

### 5.2 Implications for Researchers

The identified seven activities and five common daily cyber activity patterns indicate that app usage records reflect user activities in real life. The usage records provide a rich basis for research on multiple levels of abstraction from the analysis of specific apps and app categories to user activities and habits.

We show that demographics have a significant impact on users' app usage and activities. This may result in biased results and may explain the inability to replicate results across studies, as mentioned in [43]. For example: since our small-scale validation dataset does not cover senior citizens and teenagers, we could not discover the groups associated with the afternoon reading and nightly entertainment patterns that are presented in the large-scale dataset. This issue can be mitigated by understanding the demographics aspects of the study at hand and taking this information into account when designing the data gathering and analysis.

### 5.3 Implications for Service Providers

Smartphone app data analysis is expected to provide valuable insights for mobile service providers. They can configure and optimize their services in a dynamic manner according to the discovered patterns and behavior. For example, the usage patterns show when the high bandwidth apps are in use of the day, allowing the provider to optimize service delivery and reduce operating costs predictively. The data is also very important for generating recommendations to users based on their observed app usage.

As another example, the nightly entertainment pattern implies that low latency network service is essential in the evening and night since game and media & video apps dominate this pattern. However, in terms of our findings in section 4, there are nearly 42% of users in cluster B addicted to mobile games, still involved in entertainment activity after midnight. A recent report [44] suggests that gaming and smartphone addiction have significant detrimental effects on physical and mental well-being. The app data analysis can highlight these problems through early detection.

### 5.4 Implications for App Developers

Our work also provides useful information for app developers. For example, during the step of activity identification, we found several highly related app pairs, such as *Navigation* and *Transportation*, *Game* and *Media & video*, *Shopping* and *SON & IM*, *Food & drink* and *Photography*. These observations help application developers in making their apps more intuitive and user-friendly by grouping and merging functions of highly related applications. For example, WeChat, one of the most popular apps in China,

already supports such grouped functions through mini-programs[3]. Mini-programs are embedded in WeChat, and they enable the user to access other apps' services effectively.

### 5.5 Study Limitations

We have examined the smartphone app usage data gathered from the urban area. However, the activity patterns might be different for suburban and rural areas. The presented work is based on mobile app usage data; thus, it is only possible to detect cyber activities performed while using and carrying smartphones. It is also possible that the app usage records may not reflect the actual engagement of users' activities, as some apps may be executed automatically in the background. For example, an email app may download emails even when the user is not using that app. The basic model can be improved by introducing a user attention mechanism.

## 6 Related Work

### 6.1 App Usage Patterns

A lot of previous works or analyses focused on how individuals use their smartphones and their applications. For example, Falaki *et al.* [45] used detailed traces from 255 users to characterized user activities and found that users interact with their smartphones between 10 to 200 times per day on average and use 10-90 applications. Xu *et al.* [46] investigated the diverse usage patterns of smartphone apps via network measurements from a tier-1 cellular network provider in the US. They found that some apps have a high likelihood of co-occurrence across smartphones, that is, when a user uses one app, he or she is also likely to use another one. In our work, we also found similar results. For example, recalling *Commute and transportation* activity presented in section 3, we found that people listen to music and check emails along with the transportation apps while commuting. We also discovered that people are habituated to using news, reading, and weather apps during the same period in *Reading and checking* activity. Yang *et al.* [47] collected continuous cellular traffic over a week to characterize user behavior on mobile Internet. They showed that a user visits seven applications over a week and five categories of applications within a day on average. Canneyt *et al.* [5] collected a sample of Flurry data which consists of events from 600 million daily unique users and covers users from 221 countries. They showed how application usage behavior is disrupted through major political, social, and sports events.

Some existing studies clustered users into several particular groups and provide comprehensive descriptions for these groups. Jones *et al.* [2] analyzed users' application revisitation patterns based on three months of application launch logs from 165 users and identified three distinct user clusters, checkers, waiters, and responsives. Checkers refer to the users exhibit brief revisit patterns of fast revisitation (less than one hour). Waiters stand for the users who show longer revisit patterns, which are uniformly distributed between short-medium re-visitations (between 1min and 4hrs) and long re-visitations (from 2hrs to 3days). Responsives are the users who sometimes exhibit brief and occasionally long revisit patterns. Zhao *et al.* [4] analyzed one month of application usage from 106,762 users and discovered 382 distinct types of users based on their usage behaviors in app-category granularity. They also gave a meaningful label to the users in each cluster, such as Night communicators, Evening learners, and Financial users. In [48], Katevas *et al.* collected daily mobile phone activity data from 340 users and revealed five smartphone use profiles, i.e., limited use, business use, power use, and personality-& externally induced problematic use.

Unlike grouping users of similar application usage habits, Welke *et al.* [49] showed the significant diversity of application usage among users. They demonstrated that it is possible to differentiate users according to their application usage and found out that 500 most frequent applications are sufficient to identify 99.67% of the users. Further, Tu *et al.* [3] quantified the uniqueness of individual app usage and showed that the fingerprints of mobile app usage are highly unique. They also found user demographics, users' online, and offline behavior all influence the uniqueness level.

Since different demographic attributes can lead to differences in app usage behavior, many studies have sought to study the relationship between user personality traits and their app usage traces. For example, Seneviratne *et al.* [50] collected an app usage dataset from over 200 users and exploited linear support-vector machine to predict users' gender based on the apps used by users. Further, Malmi *et al.* [51] conducted a similar study on a more extensive app usage dataset covering 3,760 mobile users and demonstrated that, apart from gender, the app usage traces can be used to predict income as well. Zhao *et al.* [52] collected an app usage dataset from 15,000 mobile users. They extracted topic features from app descriptions and then applied the topic features of used apps to infer users' gender.

### 6.2 App Usage Prediction and Recommendation

Some scholars also worked on modeling mobile app usage and tried to predict which applications will be launched and recommend accordingly. Given the app usage sequences, it is often assumed that Markovian property stands. Zou *et al.* [53] proposed using Markovian models to learn the app usage sequences. They compared first and second-order Markov models with the weighted linear combination of these two models. The results showed that the combined model is the best model with an accuracy of 85% in predicting the top-5 apps each time of the sequence. Besides, Natarajan *et al.* [54] proposed a cluster-level Markov model to make personalized app usage prediction to a user. It first clusters users based on their usage patterns and then computes personalized PageRank for users corresponding cluster Markov graph.

Apart from only using app usage sequences, some works computed app usage features separately for different temporal contexts. In [55], Verkasalo et al. studied how mobile services are used in different contexts and found that time of day is the most useful context for app usage prediction. Liao *et al.* [56] proposed a temporal-based apps predictor to dynamically predict the apps that are most likely to be

3. https://wechatwiki.com/wechat-resources/wechat-mini-program-epic-tutorial-guide/

used. They extracted three Apps usage features, i.e., global, temporal, and periodical, from the Apps usage trace. Later they dynamically derive an app usage probability model from estimating the current usage probability of each app in each feature. Other studies [57], [58] applied similar approaches, i.e., taking time-based features into account to predict app usage dynamically. Like time features, the activity features derived in our work can be used as context, which helps to predict app usage.

With a large pool of apps currently available and the fast proliferation rate of new apps, app recommendation as an important topic also attracted many researchers. Shi *et al.* [59] proposed a similarity-based recommendation algorithm that measured the similarity of apps based on the usage patterns found among a group of users. Yan *et al.* [60] presented the AppJoy system to make personalized recommendations by picking the apps with similar usage patterns to a user's installed apps. Bae *et al.* [61] made app recommendations based on co-occurrence in usage behavior. Meanwhile, in our work, we also find several high related app pairs, like *Navigation* and *transportation*, *Game* and *Media & video*, *Shopping* and *SON & IM*, *Food & drink*, and *Photography*, which can provide prior knowledge for app recommendation systems. Besides, the contextual information, e.g., users' mobility status, location, and time of the day, is also useful for app recommendations. Davidsson *et al.* [62] combined both the context and user feedback to develop the app recommendation system. Kaji *et al.* [63] developed an app, AppLocky, which requests users to select their current context for context-aware app recommendations. The activity labels derived in our work can also be applied in recommendation systems as a kind of contextual information.

## 7 Conclusions

Although we are overwhelmed by the chaotic flow of everyday obligations, we have patterns in our digital activities that characterize our daily lives individually and collectively. In this paper, we designed a probabilistic topic model based activity detection framework for discovering daily activity patterns across mobile app usage data. By applying our framework on a large-scale and real-world dataset collected from Shanghai, one of the world's largest cities, we identified seven typical activities, i.e., commuting and transportation, entertainment, shopping, socializing, reading and checking, life and health, and exploring food. From users' activity sequences, we examined the regularity of individuals' daily activities and successfully extracted five common patterns among millions of people, including afternoon reading, nightly entertainment, pervasive socializing, commuting, and nightly socializing. We also showed that demographics have an important impact on users' daily lives. Finally, we demonstrated how our findings could be used by policymakers, government, researchers, service providers, and app developers. In this work, the patterns discovered are based on the dataset collected from only one city, i.e., Shanghai, which might not represent the whole community. Therefore, verifying the findings and comparing the differences across different cities is an exciting future direction.

## Acknowledgments

This research has been supported in part by project 16214817 from the Research Grants Council of Hong Kong, project FP805 from HKUST, the 5GEAR project, the FIT project and the CBAI (Crowdsourced Battery Optimization AI for a Connected World, grant No. 1319017) project from the Academy of Finland, the National Key Research and Development Program of China under grant 2018YFB1800804, the National Nature Science Foundation of China under U1936217, 61971267, 61972223, 61941117, 61861136003, Beijing Natural Science Foundation under L182038, Beijing National Research Center for Information Science and Technology under 20031887521, and research fund of Tsinghua University-Tencent Joint Laboratory for Internet Innovation Technology.

## References

[1] Tong Li, Mingyang Zhang, Hancheng Cao, Yong Li, Sasu Tarkoma, and Pan Hui. what apps did you use?: Understanding the long-term evolution of mobile app usage. In *Proceedings of The Web Conference 2020*, pages 66–76, 2020.

[2] Simon L. Jones, Denzil Ferreira, Simo Hosio, Jorge Goncalves, and Vassilis Kostakos. Revisitation analysis of smartphone app use. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '15, pages 1197–1208, New York, NY, USA, 2015. ACM.

[3] Zhen Tu, Runtong Li, Yong Li, Gang Wang, Di Wu, Pan Hui, Li Su, and Depeng Jin. Your apps give you away: Distinguishing mobile users by their app usage fingerprints. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(3):138, 2018.

[4] Sha Zhao, Julian Ramos, Jianrong Tao, Ziwen Jiang, Shijian Li, Zhaohui Wu, Gang Pan, and Anind K. Dey. Discovering different kinds of smartphone users through their application usage behaviors. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '16, pages 498–509, New York, NY, USA, 2016. ACM.

[5] Steven Van Canneyt, Marc Bron, Andy Haines, and Mounia Lalmas. Describing patterns and disruptions in large scale mobile app usage data. In *Proceedings of the 26th International Conference on World Wide Web Companion*, WWW '17 Companion, pages 1579–1584, Republic and Canton of Geneva, Switzerland, 2017. International World Wide Web Conferences Steering Committee.

[6] Adam J Oliner, Anand P Iyer, Ion Stoica, Eemil Lagerspetz, and Sasu Tarkoma. Carat: Collaborative energy diagnosis for mobile devices. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, page 10. ACM, 2013.

[7] Hongyi Yao, Gyan Ranjan, Alok Tongaonkar, Yong Liao, and Zhuoqing Morley Mao. Samples: Self adaptive mining of persistent lexical snippets for classifying mobile application traffic. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, MobiCom '15, pages 439–451, New York, NY, USA, 2015. ACM.

[8] Qinglong Wang, Amir Yahyavi, Bettina Kemme, and Wenbo He. I know what you did on your smartphone: Inferring app usage over encrypted data traffic. In *2015 IEEE Conference on Communications and Network Security (CNS)*, pages 433–441. IEEE, 2015.

[9] Xin Wang, Shuhui Chen, and Jinshu Su. Real network traffic collection and deep learning for mobile app identification. *Wireless Communications and Mobile Computing*, 2020.

[10] Jieba. Jieba chinese text segmentation. https://github.com/fxsjy/jieba, 2017.

[11] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.

[12] Marc Langheinrich. A privacy awareness system for ubiquitous computing environments. In *Proceedings of the 4th International Conference on Ubiquitous Computing*, UbiComp '02, pages 237–245, Berlin, Heidelberg, 2002. Springer-Verlag.

[13] Huandong Wang, Yong Li, Sihan Zeng, Gang Wang, Pengyu Zhang, Pan Hui, and Depeng Jin. Modeling spatio-temporal app usage for a large user population. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(1):27, 2019.

[14] George Hripcsak, David J Albers, and Adler Perotte. Parameterizing time in electronic health record studies. *Journal of the American Medical Informatics Association*, 22(4):794–804, 2015.

[15] Michal Rosen-Zvi, Thomas Griffiths, Mark Steyvers, and Padhraic Smyth. The author-topic model for authors and documents. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 487–494. AUAI Press, 2004.

[16] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[17] David M Blei. Probabilistic topic models. *Communications of the ACM*, 55(4):77–84, 2012.

[18] Nasser M. Nasrabadi. Pattern recognition and machine learning. *Journal of Electronic Imaging*, 16, 2007.

[19] Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. *An Introduction to Variational Methods for Graphical Models*, pages 105–161. Springer Netherlands, Dordrecht, 1998.

[20] Matthew Hoffman, Francis R. Bach, and David M. Blei. Online learning for latent dirichlet allocation. In J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 856–864. Curran Associates, Inc., 2010.

[21] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

[22] Liangjie Hong and Brian D Davison. Empirical study of topic modeling in twitter. In *Proceedings of the first workshop on social media analytics*, pages 80–88, 2010.

[23] Xuerui Wang and Andrew McCallum. Topics over time: a non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 424–433. ACM, 2006.

[24] Thomas Hofmann. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 289–296. Morgan Kaufmann Publishers Inc., 1999.

[25] L. R. Bahl, F. Jelinek, and R. L. Mercer. A maximum likelihood approach to continuous speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-5(2):179–190, March 1983.

[26] Peter F Brown, Vincent J Della Pietra, Robert L Mercer, Stephen A Della Pietra, and Jennifer C Lai. An estimate of an upper bound for the entropy of english. *Computational Linguistics*, 18(1):31–40, 1992.

[27] Graham Neubig. Neural machine translation and sequence-to-sequence models: A tutorial. *arXiv preprint arXiv:1703.01619*, 2017.

[28] Peter Harrington. Machine learning in action. *Shelter Island, NY: Manning Publications Co*, 2012.

[29] Anil K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651 – 666, 2010. Award winning papers from the 19th International Conference on Pattern Recognition (ICPR).

[30] George Karypis, Vipin Kumar, and Michael Steinbach. A comparison of document clustering techniques. In *KDD workshop on text mining*, volume 400, pages 525–526. Boston, 2000.

[31] Maria Halkidi, Yannis Batistakis, and Michalis Vazirgiannis. On clustering validation techniques. *Journal of intelligent information systems*, 17(2-3):107–145, 2001.

[32] Tâm Huynh, Mario Fritz, and Bernt Schiele. Discovery of activity patterns using topic models. In *UbiComp*, volume 8, pages 10–19, 2008.

[33] Li Yujian and Liu Bo. A normalized levenshtein distance metric. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1091–1095, June 2007.

[34] David Stark and Balázs Vedres. Social sequence analysis. *The Emergence of Organizations and Markets*, pages 347–64, 2012.

[35] Albertus SJ Helberg and Hendrik C Ferreira. On multiple insertion/deletion correcting codes. *IEEE Transactions on Information Theory*, 48(1):305–308, Jan 2002.

[36] M. Serva and F. Petroni. Indo-european languages tree by levenshtein distance. *EPL (Europhysics Letters)*, 81(6):68005, 2008.

[37] Robert A. Wagner and Michael J. Fischer. The string-to-string correction problem. *J. ACM*, 21(1):168–173, January 1974.

[38] William Cohen, Pradeep Ravikumar, and Stephen Fienberg. A comparison of string metrics for matching names and records. In *Kdd workshop on data cleaning and object consolidation*, volume 3, pages 73–78, 2003.

[39] Richard W Hamming. Error detecting and error correcting codes. *Bell System technical journal*, 29(2):147–160, 1950.

[40] Fionn Murtagh. A survey of recent advances in hierarchical clustering algorithms. *The Computer Journal*, 26(4):354–359, 1983.

[41] Jinwook Seo and B. Shneiderman. Interactively exploring hierarchical clustering results [gene identification]. *Computer*, 35(7):80–86, July 2002.

[42] Shuai Yang, Sixing Chen, and Bin Li. The role of business and friendships on wechat business: An emerging business model in china. *Journal of Global Marketing*, 29(4):174–187, 2016.

[43] Karen Church, Denzil Ferreira, Nikola Banovic, and Kent Lyons. Understanding the challenges of mobile phone usage data. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '15, pages 504–514, New York, NY, USA, 2015. ACM.

[44] Chun-Hao Liu, Sheng-Hsuan Lin, Yuan-Chien Pan, and Yu-Hsuan Lin. Smartphone gaming and frequent use pattern associated with smartphone addiction. *Medicine*, 95(28), 2016.

[45] Hossein Falaki, Ratul Mahajan, Srikanth Kandula, Dimitrios Lymberopoulos, Ramesh Govindan, and Deborah Estrin. Diversity in smartphone usage. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, MobiSys '10, pages 179–194, New York, NY, USA, 2010. ACM.

[46] Qiang Xu, Jeffrey Erman, Alexandre Gerber, Zhuoqing Mao, Jeffrey Pang, and Shobha Venkataraman. Identifying diverse usage behaviors of smartphone apps. In *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, IMC '11, pages 329–344, New York, NY, USA, 2011. ACM.

[47] Jie Yang, Yuanyuan Qiao, Xinyu Zhang, Haiyang He, Fang Liu, and Gang Cheng. Characterizing user behavior in mobile internet. *IEEE Transactions on Emerging Topics in Computing*, 3(1):95–106, March 2015.

[48] Kleomenis Katevas, Ioannis Arapakis, and Martin Pielot. Typical phone use habits: intense use does not predict negative well-being. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services*, page 11. ACM, 2018.

[49] Pascal Welke, Ionut Andone, Konrad Blaszkiewicz, and Alexander Markowetz. Differentiating smartphone users by app usage. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '16, pages 519–523, New York, NY, USA, 2016. ACM.

[50] Suranga Seneviratne, Aruna Seneviratne, Prasant Mohapatra, and Anirban Mahanti. Your installed apps reveal your gender and more! *SIGMOBILE Mob. Comput. Commun. Rev.*, 18(3):5561, January 2015.

[51] Eric Malmi and Ingmar Weber. You are what apps you use: Demographic prediction based on user's apps. In *International AAAI Conference on Web and Social Media*, ICWSM 16, 2016.

[52] Sha Zhao, Yizhi Xu, Xiaojuan Ma, Ziwen Jiang, Zhiling Luo, Shijian Li, Laurence Tianruo Yang, Anind Dey, and Gang Pan. Gender profiling from a single snapshot of apps installed on a smartphone: An empirical study. *IEEE Transactions on Industrial Informatics*, 16(2):1330–1342, Feb 2020.

[53] Xun Zou, Wangsheng Zhang, Shijian Li, and Gang Pan. Prophet: What app you wish to use next. In *Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication*, pages 167–170. ACM, 2013.

[54] Nagarajan Natarajan, Donghyuk Shin, and Inderjit S Dhillon. Which app will you use next?: collaborative filtering with interactional context. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 201–208. ACM, 2013.

[55] Hannu Verkasalo. Contextual patterns in mobile service usage. *Personal Ubiquitous Comput.*, 13(5):331–342, June 2009.

[56] Zhung-Xun Liao, Yi-Chin Pan, Wen-Chih Peng, and Po-Ruey Lei. On mining mobile apps usage behavior for predicting apps usage in smartphones. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 609–618. ACM, 2013.

[57] Tingxin Yan, David Chu, Deepak Ganesan, Aman Kansal, and Jie Liu. Fast app launching for mobile devices using predictive user

context. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*, pages 113–126. ACM, 2012.
[58] Chen Sun, Jun Zheng, Huiping Yao, Yang Wang, and D Frank Hsu. Apprush: using dynamic shortcuts to facilitate application launching on mobile devices. *Procedia Computer Science*, 19:445–452, 2013.
[59] Kent Shi and Kamal Ali. Getjar mobile application recommendations with very sparse datasets. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 204–212. ACM, 2012.
[60] Bo Yan and Guanling Chen. Appjoy: personalized mobile application discovery. In *Proceedings of the 9th international conference on Mobile systems, applications, and services*, pages 113–126. ACM, 2011.
[61] Donghwan Bae, Keejun Han, Juneyoung Park, and Mun Y Yi. Apptrends: A graph-based mobile app recommendation system using usage history. In *Big Data and Smart Computing (BigComp), 2015 International Conference on*, pages 210–216. IEEE, 2015.
[62] Christoffer Davidsson and Simon Moritz. Utilizing implicit feedback and context to recommend mobile applications from first use. In *Proceedings of the 2011 Workshop on Context-awareness in Retrieval and Recommendation*, pages 19–22. ACM, 2011.
[63] Katsuhiko Kaji, Motoki Yano, and Nobuo Kawaguchi. App. locky: users' context collecting platform for context-aware application recommendation. In *Proceedings of the 2nd international workshop on Ubiquitous crowdsouring*, pages 29–32. ACM, 2011.

**Tong Li** received the B.S. degree and M.S. degree in communication engineering from Hunan University, China, in 2014 and 2017. At present, he is a dual Ph.D. student at the Hong Kong University of Science and Technology and the University of Helsinki. His research interests include mobile computing, edge network, and mobile big data mining. He is an IEEE student member.

**Yong Li** (M'09-SM'16) is currently a Tenured Associate Professor of the Department of Electronic Engineering, Tsinghua University. He received the Ph.D. degree in electronic engineering from Tsinghua University in 2012. His research interests include machine learning and big data mining, particularly, automatic machine learning and spatial-temporal data mining for urban computing, recommender systems, and knowledge graphs. Dr. Li has served as General Chair, TPC Chair, SPC/TPC Member for several international workshops and conferences, and he is on the editorial board of two IEEE journals. He has published over 100 papers on first-tier international conferences and journals, including KDD, WWW, UbiComp, SIGIR, AAAI, TKDE, TMC etc, and his papers have total citations more than 8300. Among them, ten are ESI Highly Cited Papers in Computer Science, and five receive conference Best Paper (runup) Awards. He received IEEE 2016 ComSoc Asia-Pacific Outstanding Young Researchers, Young Talent Program of China Association for Science and Technology, and the National Youth Talent Support Program.

**Mohammad A. Hoque** obtained his M.Sc degree in Computer Science and Engineering in 2010, and Ph.D in 2013 from Aalto University. He is a postdoctoral researcher at the University of Helsinki. His research interests include energy efficient mobile computing, data analysis, distributed computing, and resource-aware scheduling.

**Tong Xia** received the B.S. degree in electrical engineering from School of Electrical Information, Wuhan University, Wuhan, China, in 2017. At present, she is studying for the M.S. degree in big data from Department of Electronic Engineering, Tsinghua University, Beijing, China. Her research interests include human mobility, mobile big data mining, user behavior modelling and urban computing.

**Sasu Tarkoma** (SM'12) received the MSc and PhD degrees in computer science from the Department of Computer Science, University of Helsinki. He is a Professor of Computer Science at the University of Helsinki, and Head of the Department of Computer Science. He has authored 4 textbooks and has published over 200 scientific articles. His research interests are Internet technology, distributed systems, data analytics, and mobile and ubiquitous computing. He is Fellow of IET and EAI. He has nine granted US Patents. His research has received several Best Paper awards and mentions, for example at IEEE PerCom, IEEE ICDCS, ACM CCR, and ACM OSR.

**Pan Hui** (SM'14-F'18) received his Ph.D. degree from the Computer Laboratory at University of Cambridge, and both his Bachelor and MPhil degrees from the University of Hong Kong.

He is the Nokia Chair Professor in Data Science and Professor of Computer Science at the University of Helsinki. He is also the director of the HKUST-DT Systems and Media Lab at the Hong Kong University of Science and Technology. He was a senior research scientist and then a Distinguished Scientist for Telekom Innovation Laboratories (T-labs) Germany and an adjunct Professor of social computing and networking at Aalto University. His industrial profile also includes his research at Intel Research Cambridge and Thomson Research Paris. He has published more than 300 research papers and with over 17,500 citations. He has 30 granted and filed European and US patents in the areas of augmented reality, data science, and mobile computing. He has been serving on the organising and technical program committee of numerous top international conferences including ACM SIGCOMM, MobiSys, IEEE Infocom, ICNP, SECON, IJCAI, AAAI, ICWSM and WWW. He is an associate editor for the leading journals IEEE Transactions on Mobile Computing and IEEE Transactions on Cloud Computing. He is an IEEE Fellow, an ACM Distinguished Scientist, and a member of the Academia Europaea.

# Paper II

II

Tong Li, Mingyang Zhang, Yong Li, Eemil Lagerspetz, Sasu Tarkoma, and Pan Hui

**The Impact of Covid-19 on Smartphone Usage**

In *IEEE Internet of Things Journal*,
8(23): 16723-16733, 2021.

# The Impact of Covid-19 on Smartphone Usage

Tong Li, *Student Member, IEEE,* Mingyang Zhang, Yong Li, *Senior Member, IEEE,*
Eemil Lagerspetz, Sasu Tarkoma, *Senior Member, IEEE,* and Pan Hui, *Fellow, IEEE*

***Abstract*—The outbreak of Covid-19 changed the world as well as human behavior. In this paper, we study the impact of Covid-19 on smartphone usage. We gather smartphone usage records from a global data collection platform called Carat, including the usage of mobile users in North America from November 2019 to April 2020. We then conduct the first study on the differences in smartphone usage across the outbreak of Covid-19. We discover that Covid-19 leads to a decrease in users' smartphone engagement and network switches, but an increase in WiFi usage. Also, its outbreak causes new typical diurnal patterns of both memory usage and WiFi usage. Additionally, we investigate the correlations between smartphone usage and daily confirmed cases of Covid-19. The results reveal that memory usage, WiFi usage, and network switches of smartphones have significant correlations, whose absolute values of Pearson coefficients are greater than 0.8. Moreover, smartphone usage behavior has the strongest correlation with the Covid-19 cases occurring after it, which exhibits the potential of inferring outbreak status. By conducting extensive experiments, we demonstrate that for the inference of outbreak stages, both Macro-F1 and Micro-F1 can achieve over 0.8. Our findings explore the values of smartphone usage data for fighting against the epidemic.**

***Index Terms*—Smartphone usage, Covid-19, correlations, outbreak stage inference.**

## I. Introduction

At the beginning of 2020, Covid-19 was identified and has spread globally [1]. The outbreak of Covid-19 has changed people's lives significantly. Countless efforts have been made to study the world after Covid-19 from different perspectives, ranging from world economy [2], personal mental health [3], to human mobility [4]. Meanwhile, since the first iPhone was released in 2007, smartphones have become a necessity in daily lives [5]. The number of smartphone users worldwide today has surpassed three billion [6]. However, up to now, the understanding of the impact of Covid-19 on smartphone usage is still inadequate. Specifically, studying how Covid-19 affects users' smartphone usage behavior can bring two-fold benefits. First, understanding smartphone usage differences across the Covid-19 outbreak is critical for the industry, e.g., smartphone manufacturers and network service providers, to dynamically adjust market strategies and enhance user experience. Second, smartphones are embedded with a set of sensors recording user activities in both cyber and physical spaces [7]. By exploring the impact, we can use such rich behavioral data to infer different Covid-19 outbreak stages and further contribute to the fight against Covid-19.

Meanwhile, some previous studies have introduced mobile sensing data to the public health field. For example, Yarkoni [8] proposed the concept of Psychoinformatics, using tools and techniques from information sciences to improve psychological research. Insel [9] and Baumeister *et al.* [10] introduced digital phenotyping that leverages digital behavior data logged on smartphone sensors to detect psychological states. Further, Markowetz *et al.* [11] proposed to explore big data technologies and conduct digital phenotyping on a large-scale. The above studies showed the correlation between smartphone usage and the psychological states of users. The Covid-19 pandemic represents a global health crisis, which will severely change psychological burdens and physical activities of individuals [12]. Such changes may be conveyed to and reflected in smartphone usage [13]. In this way, we are motivated to investigate how the outbreak of Covid-19 affects smartphone usage behavior.

In this work, we make an effort towards understanding the impact of Covid-19 on smartphone usage and explore the potential of smartphone usage data to fight against Covid-19. More specifically, we study the following research problems.

1) Does the outbreak of Covid-19 affect users' smartphone usage, and how?
2) Can we use smartphone usage data, e.g., CPU usage, memory usage, and network connections, to infer the outbreak stages of Covid-19?

To answer the above two questions, we reveal the correlations between smartphone usage and the outbreak of Covid-19 from both statistics and dynamic patterns. We first collect a large-scale smartphone usage dataset by leveraging a global crowdsourcing platform called Carat. The dataset covers users in North America and their smartphone usage records for six months from November 2019 to April 2020 (Section II). Next, we use the dataset to make a statistical analysis. The results demonstrate that the outbreak of Covid-19 has indeed impacted significantly on users' smartphone usage behavior in terms of CPU usage, memory usage, WiFi usage, and network switches. In our case, CPU and memory usage describe how much of the processor's and memory's capacity is in use, respectively. WiFi usage indicates the percentage of records under WiFi connection. Network switch refers to the change of network connection from WiFi to cellular network and vice versa. Specifically, the CPU usage and memory usage reflect the intensity of smartphone engagement of users. The WiFi usage and network switches reveal users' mobility intensity.

T. Li, M. Zhang and P. Hui are with the System and Media Laboratory (SyMLab), Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong. T. Li and P. Hui are also with the Department of Computer Science, University of Helsinki, Helsinki, Finland. (E-mail: t.li@connect.ust.hk, mzhangbj@ust.hk, panhui@cse.ust.hk)

Y. Li is with Beijing National Research Center for Information Science and Technology (BNRist), Department of Electronic Engineering, Tsinghua University, Beijing 100084, China. (E-mail: liyong07@tsinghua.edu.cn)

E. Lagerspetz and S. Tarkoma are with the Department of Computer Science, University of Helsinki, Helsinki, Finland. (E-mail: eemil.lagerspetz@cs.helsinki.fi, sasu.tarkoma@helsinki.fi)

TABLE I
SAMPLES OF THE COLLECTED SMARTPHONE USAGE DATA, WHERE LTE AND UTMS ARE SPECIFIC MODES OF CELLULAR NETWORK. USER IDS HAVE BEEN ANONYMIZED.

| User ID | Timestamp | CPU usage (%) | Active memory (KB) | Free memory (KB) | Network | Battery level (%) | Timezone | MCC |
|---|---|---|---|---|---|---|---|---|
| 1 | 2019-11-05 05:48:48 | 0.6992 | 3528 | 1875820 | WiFi | 66 | America/Denver | us |
| 2 | 2019-11-13 07:51:11 | 51.5152 | 429632 | 1746900 | LTE | 79 | America/New_York | us |
| 3 | 2019-11-16 12:31:51 | 53.1807 | 1006248 | 2853892 | UTMS | 94 | Europe/Helsinki | fi |

Further, we extend our analysis to dynamic patterns, i.e., diurnal patterns of smartphone usage. The results unveil how the outbreak of Covid-19 affects usage behavior during the time of one day. We also examine the correlations between smartphone usage and daily confirmed cases (Section III). Moreover, we investigate smartphone usage data's inference ability for Covid-19 outbreak stages using both statistical and deep learning methods. By comparing the performance and conducting importance analysis, we select the most potent smartphone usage features for the outbreak stage inference (Section IV).

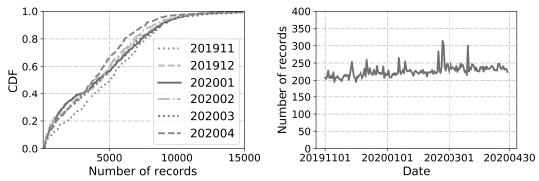
Among the many insightful results and observations, the following are the most prominent.

- The outbreak of Covid-19 causes a decrease in users' smartphone engagement in terms of both CPU usage and memory usage. However, it has different impacts on CPU and memory usage according to their diurnal patterns. Specifically, it leads to a new typical diurnal pattern of memory usage while it only changes the proportion of existing patterns of CPU usage.
- The outbreak of Covid-19 makes an increase in WiFi usage and a decrease in network switches, implying that users reduce their mobility intensity. Also, similar to memory usage, a new typical diurnal pattern of WiFi usage has emerged after the outbreak.
- Memory usage, WiFi usage, and network switches have significant correlations with the number of daily confirmed cases of Covid-19. Also, the correlation between smartphone usage behavior and Covid-19 daily cases has a time delay. Smartphone usage changes earlier than the number of cases. That is because the smartphone data can reflect the outbreak status in real-time. However, such reflection cannot be immediately expressed in daily cases due to the diagnosis delay.
- By using smartphone usage data to infer Covid-19 outbreak stages, we can achieve over 0.8 for both Macro-F1 and Micro-F1, which presents a promising application of smartphone usage data on fighting against Covid-19.

## II. DATASET OVERVIEW

### A. Data Collection

We leverage a crowdsourcing platform called Carat to collect smartphone usage data. Carat is a cross operating system mobile app, including both iOS[1] and Android[2], which can record users' smartphone usage traces automatically. Carat can monitor and record the working status of smartphones in detail. In practice, Carat informs of all data collection items in the End-user License Agreement (EULA) when users install Carat to alleviate user privacy concerns. Also, Carat users are anonymized, and the app does not collect any personal information. It is worth noting that Carat is live. Up to now, Carat has been downloaded over 100 thousand times. The number of downloads and installations is increasing every day.




(a) (b)

Fig. 1. This figure shows (a) The cumulative distribution function (CDF) of the number of records per month for each unique user. (b) The daily average number of collected records of users.

Specifically, Carat applies an event-triggered collection scheme, gathering a data sample every time the battery level changes by 1%. Each data sample contains a list of smartphone hardware status, including CPU, memory, battery, and network. Each sample also has several other features, including a user-specific identifier, timestamp, timezone, and mobile country code (MCC). The MCC is obtained from the cellular network and automatically converted to a two-character country code. Table I presents samples of collected smartphone usage data to show the data format.

### B. Basic Analysis

Since we focus on studying the impact of Covid-19, we select the records from November 2019 to April 2020. Also, we principally consider samples collected from North America. In total, we have 452 users with over 7,517,494 records. Since users involved may uninstall and reinstall Carat during the data collection period, the number of active users changes over time, i.e., November 2019 (293 users), December 2019 (295 users), January 2020 (251 users), February 2020 (224 users), March 2020 (198 users), and April 2020 (158 users). In our case, we use both timezones and MCC to determine the users' country, which increases the reliability of detection. Table II summarizes the dataset.

Next, we depict basic statistics to illustrate the quality and representativeness of the collected smartphone usage dataset. Fig. 1(a) presents the cumulative distribution function (CDF) of the number of records per month for each unique user. We observe that the involved users kept a high activeness level

[1]https://apps.apple.com/us/app/carat/id504771500
[2]https://play.google.com/store/search?q=carat

TABLE II
SUMMARY OF THE COLLECTED DATASET FROM THE USA.

| # Users | # Records | Attributes | Date | Area |
|---|---|---|---|---|
| 452 | 7,517,494 | User ID, timestamp, CPU usage, Memory usage, network status, battery level, timezone, MCC | 11/2019 - 04/2020 | North America |

during the data collection period. For each month, more than 20% of users have over 1,800 records. Moreover, we plot how the average number of users' records changes every day in Fig. 1(b). We can witness that there are around 220 records every day per user on average. Although there are some fluctuations, the curve is relatively stable. Such a high number of records per user demonstrates our dataset's effectiveness in capturing the smartphone usage behavior of users involved covering the entire six months, i.e., from November 2019 to April 2020. Also, the continuity of the data collection guarantees the representativeness of our study.

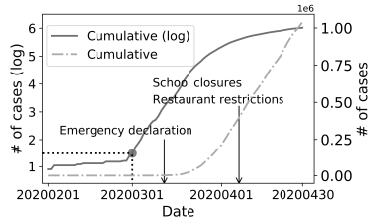



Fig. 2. The cumulative number of confirmed cases changes over time. The federal government issued an emergency declaration on March 13, 2020. Most states issued school closure rules and restaurant restrictions by April 7, 2020.

### C. Ethical Considerations

We are very aware of the privacy issues when using the collected data for research. We have taken adequate actions to safeguard the privacy of the involved mobile users. **First**, we do not collect any personal information from users. A user-specific identifier is randomly generated when the user first installs Carat. We only have users' country information rather than sensitive location information, like GPS data. Also, the data-gathering part of Carat is open-source[3]. Users can examine it easily. The users involved are informed of the data collection and management procedures in the End-user License Agreement (EULA) and grant their consent from their devices. In the EULA, we also point out that the data we collect may be used to improve products or for research purposes. **Second**, the dataset is stored in a secure local server protected by strict authentication mechanisms and firewalls. All researchers are regulated by a strict non-disclosure agreement to access the data. **Finally**, this work has received approval from all the authors' local institutions.

## III. DIFFERENCES IN SMARTPHONE USAGE

In this section, we aim to solve the first research problem, i.e., whether and how the outbreak of Covid-19 affects users' smartphone usage behavior. Specifically, we explore the impact on CPU usage, memory usage, and network status from statistical and dynamic pattern analysis. The data processing and analysis was conducted in Helsinki.

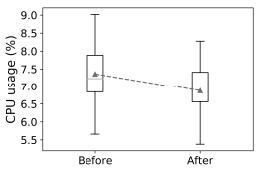
### A. Differences in Number and Distributions

To determine whether the outbreak of Covid-19 changes users' mobile engagement, first of all, we need to determine the outbreak date in North America. Fig. 2 shows the cumulative number of confirmed cases in North America from February 2020 to April 2020 and the governmental policies on the same timescale. The dashed curve is in the linear scale, while the solid curve depicts the cumulative number in the logarithmic scale. Notably, the propagation of Covid-19 is in exponential growth. Therefore, using the logarithmic scale curve makes it more accessible to detect the phase change of increase trend and determine the outbreak date accordingly [14]. In terms of Fig. 2, we can observe an apparent step-up around March 1, 2020, as denoted by the red point. Hence, we regard March 1, 2020, as the outbreak date of Covid-19 in North America.

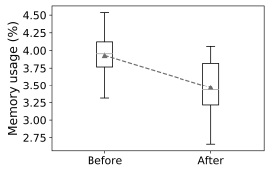
We then begin the analysis by comparing the distributions of smartphone usage variables before and after the outbreak of Covid-19. In Fig. 3, we use box-plots to depict the distributions of the percentages of CPU usage, memory usage, WiFi usage, and network switches, respectively. Specifically, the 'Before' set contains the samples from November 1, 2019, to February 29, 2020, while the 'After' set contains the samples from March 1, 2020, to April 30, 2020. The box-plots describe data distribution through quartiles. The candlesticks represent the minimum and the maximum values, while the boxed area contains the values between 25% and 75% quartiles. The horizontal line denotes the median, while the green upper triangle indicates the mean.

There is an apparent difference in smartphone usage across the outbreak in terms of all hardware variables. The mean values of CPU and memory usage drop from 7.36% and 3.93% to 6.87% and 3.47%, respectively. Their differences across the outbreak are significant under a two-sided $t$-test [15] with $p$ values of $5.239{\cdot}10^{-5} \ll 0.001$ and $7.383{\cdot}10^{-18} \ll 0.001$. The decreases imply that users' smartphone engagement becomes less active after the outbreak, i.e., March 1, 2020. Meanwhile, the WiFi usage percentage grows dramatically, where the mean value rises from 56.95% to 64.06%. The distribution difference is also significant under a two-sided $t$-test with a $p$ value of $2.585 \cdot 10^{-19} \ll 0.001$. Since WiFi access points are usually deployed indoors, we can conclude that people have more time to stay indoors instead of going outside after the outbreak of Covid-19. Moreover, we also notice that the percentage of network switches drops remarkably. The mean value declines from 3.98% to 2.85%, and the distribution difference is significant, with $p$ value $1.526 \cdot 10^{-23} \ll 0.001$. Similar to

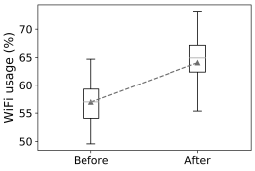
[3]The code is available at https://github.com/carat-project/carat/.



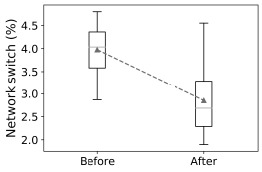

(a) The distributions of the percentage of CPU usage, $p = 5.239 \cdot 10^{-5}$.




(b) The distributions of the percentage of memory usage, $p = 7.383 \cdot 10^{-18}$.




(c) The distributions of the percentage of WiFi usage, $p = 2.585 \cdot 10^{-19}$.




(d) The distributions of the percentage of network switches, $p = 1.526 \cdot 10^{-23}$.

Fig. 3. The differences in smartphone usage before and after the outbreak of Covid-19.

TABLE III
CORRELATIONS BETWEEN MEMORY USAGE AND WIFI USAGE ACROSS DIFFERENT TIME PERIODS.

| Over complete time window | Before outbreak | After outbreak |
|---|---|---|
| -0.0689 | 0.3205 | -0.3240 |

WiFi usage, network switches also reflect the movement of mobile users. Since the WiFi network is commonly deployed indoors and limited by its coverage, network switches usually occur when mobile users go from indoors to outside and from outside to indoors. Consequently, the percentage of network switches can reveal the mobility intensity of smartphone users. In this way, the decreasing trend of network switches suggests users have less mobility after the outbreak.

As a result, based on the differences in number and distributions, we can conclude that the outbreak of Covid-19 causes a decrease in smartphone engagement in terms of both CPU and memory usage. Meanwhile, the outbreak causes an increase in users' intensity staying indoors in terms of WiFi usage. Further, we depict the correlation between WiFi usage and memory usage to investigate smartphone usage intensity when people stay indoors. Table III shows the correlations across different time periods, i.e., over the complete time windows, before the outbreak, and after the outbreak. As depicted in Table III, WiFi usage and memory usage have a weak positive correlation before the outbreak, which follows the commonly-held intuition. However, after the outbreak, the correlation becomes weak negative. We infer that the longer time to stay at home after the outbreak may cause such differences. When people have more time at home, they will prefer to use their computers and laptops for entertainment instead of smartphones.

### *B. Differences in Diurnal Patterns*

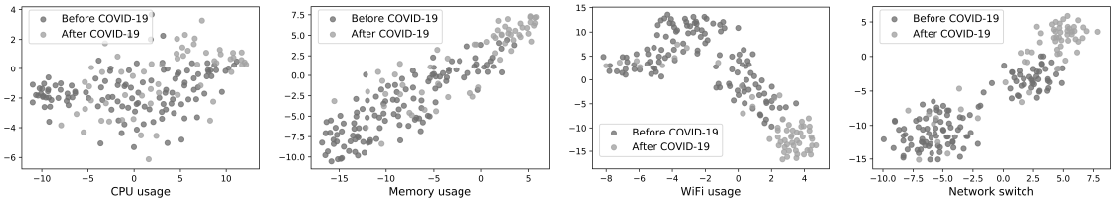
In terms of the above statistical analysis, we can conclude that the outbreak of Covid-19 has affected users' smartphone usage behavior. Next, we delve into the dynamic analysis, i.e., revealing the differences in diurnal patterns. The diurnal pattern depicts how users' smartphone usage behavior unfolds over the time of the day, which is an essential temporal pattern studied by many previous studies [16], [17].

We define each day's diurnal pattern by averaging the usage data over the day's active users. In our case, we evenly divide one day into 48 time-slots, where each time-slot represents half an hour. Therefore, each diurnal sequence is of 48 dimensions. Next, we compute smartphone usage data for each time-slot. In practice, as for CPU usage and memory usage behavior, we take the averages in that time slot. For WiFi usage, we calculate the proportion of WiFi connection records in that time slot. Besides, for network switches, we calculate the proportion of network type changes in the time slot. By doing so, given one day, each type of smartphone usage behavior will have a diurnal sequence with 48 dimensions. In total, we have 728 diurnal sequences, i.e., 182 (# of days) $\times$ 4 (# of usage types).

After obtaining the diurnal sequences, we use the t-SNE transformation [18] to visualize them, as shown in Fig. 4. t-SNE is a commonly used data transformation method that projects high-dimensional data to a low-dimensional space while keeping the similarity across objects. In Fig. 4, blue points represent the dates before the outbreak, while orange points represent the dates after the outbreak. We can observe that excluding CPU usage, the other types of smartphone usage behavior appear to be nicely separated by the outbreak. This shows the existence of differences in diurnal patterns of smartphone usage before and after the outbreak of Covid-19.

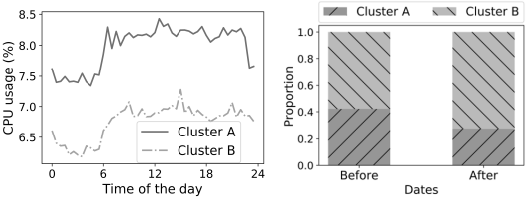
Based on the t-SNE visualization results, we propose a hypothesis that the outbreak of Covid-19 will lead to a new diurnal pattern for smartphone usage. In our case, the new pattern means that it does not or rarely appears before the outbreak but is popular on the dates after the outbreak. To test the hypothesis, we apply K-means to cluster diurnal sequences of the entire 182 days for each type of smartphone usage behavior and examine whether the cluster results can be distinguished by the outbreak date of Covid-19. Since there are only two situations for any date, i.e., before or after the outbreak, we set the number of clusters to two. The clustering results are presented in Fig. 5$\sim$8, where the cluster A and B refer to the two-cluster output of K-means. Also, we regard the centroid as the typical diurnal pattern of the cluster.

**Diurnal patterns of CPU usage.** As shown in Fig. 5(a), the obtained two typical diurnal patterns of CPU usage have the same trend but different values. Both of them decrease during the night and increase during the day, while cluster B's centroid is of lower numerical values. Fig. 5(b) shows that, compared to cluster A, cluster B accounts for a higher proportion of the dates after the outbreak, consistent with the dropping trend observed in Fig. 3(a). We also observe that Covid-19 only affects the proportion of two cluster labels, and both typical patterns frequently appear on the dates before the



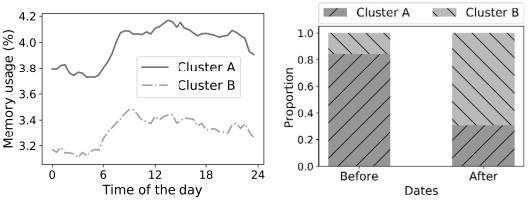

(a) t-SNE representation of CPU usage patterns. (b) t-SNE representation of memory usage patterns. (c) t-SNE representation of WiFi usage patterns. (d) t-SNE representation of network switch patterns.

Fig. 4. t-SNE representation of diurnal sequences of smartphone usage, projecting high-dimensional data to a 2-dimensional space while keeping the similarity across objects.



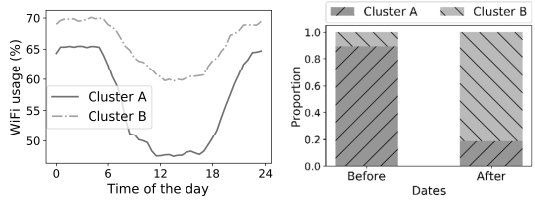

(a) The centroids of clusters. (b) Proportion of cluster labels.

Fig. 5. Cluster results of CPU usage diurnal patterns.



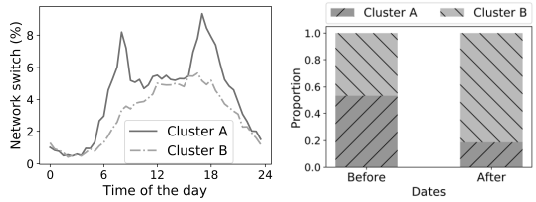

(a) The centroids of clusters. (b) Proportion of cluster labels.

Fig. 6. Cluster results of memory usage diurnal patterns.




(a) The centroids of clusters. (b) Proportion of cluster labels.

Fig. 7. Cluster results of WiFi usage diurnal patterns.




(a) The centroids of clusters. (b) Proportion of cluster labels.

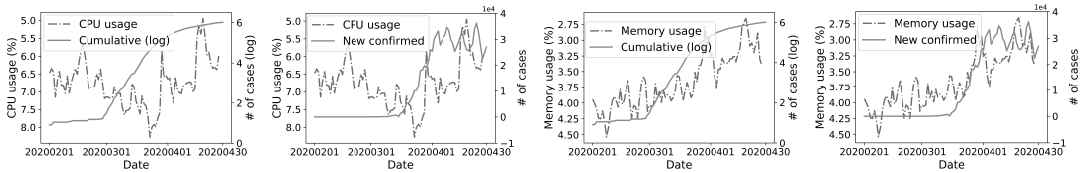
Fig. 8. Cluster results of network switch diurnal patterns.

outbreak. In other words, the outbreak did not create a new typical diurnal pattern of CPU usage. The t-SNE visualization in Fig. 4(a) also verifies this.

**Diurnal patterns of memory usage.** As depicted in Fig. 6(a), similar to CPU usage, two typical diurnal patterns obtained are also with the same trend but different numerical values. In terms of Fig. 6(b), over 80% of the dates before the outbreak belong to cluster A. Meanwhile, more than 65% of the dates after the outbreak belong to cluster B. Therefore, we can conclude that the cluster results can be distinguished by the outbreak date. Also, cluster B's centroid can be regarded as a new typical diurnal pattern because it rarely appears before the outbreak and becomes common after the outbreak. In summary, Covid-19 leads to the appearance of a new typical diurnal pattern of memory usage, corresponding to the t-SNE visualization in Fig. 4(b).

**Diurnal patterns of WiFi usage.** Fig. 7 displays the cluster results of WiFi usage. Unlike CPU and memory usage, apart from numerical differences, the centroids of WiFi usage clusters also have different changing trends. As depicted in Fig. 7(a), the centroid of cluster B has a higher percentage of WiFi usage throughout the day. Instead of a cliff-like drop shown in cluster A, cluster B has a slow-down after 6 am. This indicates that users need less mobile network support on the dates in cluster B. Moreover, similar to memory usage, the dates after the outbreak have a dominating cluster, i.e., cluster B. Therefore, Covid-19 also brings a new diurnal pattern of WiFi usage, leading users to use more WiFi connections.

**Diurnal patterns of network switches.** We exhibit the clustering results of network switch patterns in Fig. 8. As discussed in Section III-A, network switches can reflect the mobility intensity of smartphone users. In Fig. 8(a), the centroid of cluster A presents two peaks in the morning and evening rush hours, which verifies the above discussion. We notice that less than 18% of the dates after the outbreak belong to cluster A, indicating that users' mobility intensity drops significantly. Alternatively, cluster B has fewer network switches throughout the day and without bimodal patterns, indicating that users




(a) CPU usage and cumulative confirmed cases.
(b) CPU usage and new confirmed daily cases.
(c) Memory usage and cumulative confirmed cases.
(d) Memory usage and new confirmed daily cases.
(e) WiFi usage and cumulative confirmed cases.
(f) WiFi usage and new confirmed daily cases.
(g) Network switches and cumulative confirmed cases.
(h) Network switches and new confirmed daily cases.

Fig. 9. The daily patterns of smartphone usage and the number of daily confirmed cases of Covid-19.

TABLE IV
PEARSON CORRELATIONS BETWEEN SMARTPHONE USAGE AND COVID-19 CASES.

| Delay (day) | CPU usage | | Memory usage | | WiFi usage | | Network switch | |
|---|---|---|---|---|---|---|---|---|
| | Cumulative (log) | New confirmed | Cumulative (log) | New confirmed | Cumulative (log) | New confirmed | Cumulative (log) | New confirmed |
| 0 | -0.0632 | -0.2670 | -0.8163 | -0.8119 | 0.8364 | 0.8053 | -0.7649 | -0.7809 |
| 1 | -0.0383 | -0.2339 | -0.8119 | -0.8091 | 0.8754 | 0.8253 | -0.7838 | -0.8234 |
| 2 | -0.0541 | -0.2444 | -0.8315 | -0.8369 | 0.8667 | 0.8263 | -0.7830 | -0.8369 |
| 3 | -0.0240 | -0.2081 | -0.8277 | -0.8399 | 0.8523 | 0.8312 | -0.7772 | -0.8430 |

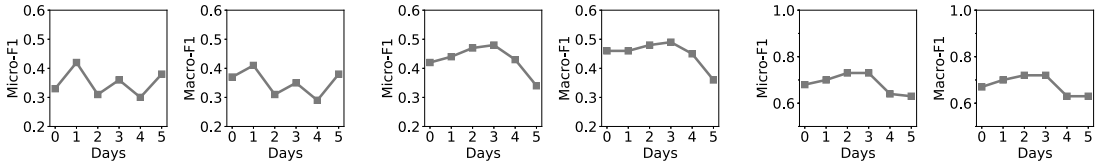
have less mobility on the dates in that cluster. Although cluster B dominates the dates after the outbreak, it also frequently appears before the outbreak. As a result, similar to CPU usage, Covid-19 only changes the proportion of different network switch patterns but does not trigger the appearance of new patterns.

Consequently, the outbreak of Covid-19 also profoundly affects diurnal patterns of smartphone usage behavior, implying that the diurnal sequences of smartphone usage can be used to reflect the outbreak status.

### *C. Correlations Between Smartphone Usage and Covid-19 Daily Cases*

We then analyze the correlations between smartphone usage and Covid-19 daily cases. Specifically, we take the average over the active users of each day and plot both the daily sequences of smartphone usage and the number of daily confirmed cases of Covid-19 in Fig. 9, from February 1, 2020 to March 30, 2020. For the figures of CPU usage, memory usage, and network switches, we inverse the y-axis for better visualization. From the results, we can observe that memory usage, WiFi usage, and network switches have strong correlations with both cumulative and new confirmed daily cases. That is because smartphone usage behavior reflects users' physical activities, e.g., staying at home and mobility intensity. Meanwhile, users' physical activities will influence and be affected by Covid-19. Therefore, smartphone usage behavior can indirectly reveal Covid-19 trends. Moreover, in Fig. 9(f) and Fig. 9(h), we discover a delay in the changing trends between smartphone usage behavior and new confirmed cases. In other words, smartphone usage changes earlier than the number of cases.

Further, to better explore the delay phenomenon, we put a set of delays on the daily sequences of smartphone usage behavior from 0 to 3 days. Then, we compute the Pearson correlation between shifted smartphone usage and Covid-19 sequences. The results are illustrated in Table IV. From the results, we discover that different smartphone usage features have various correlations with daily confirmed cases. Generally, memory usage, WiFi usage, and network switches have significant linear correlations with Covid-19 daily confirmed cases. The absolute values of their Pearson coefficients are greater than 0.8. However, CPU usage has a weak Pearson correlation, only around -0.26, with new confirmed cases of Covid-19. These observations are consistent with the findings in Section III-B. Moreover, when we delay usage behavior, it will have a higher correlation with Covid-19 cases, which corresponds to the observation that smartphone usage changes earlier than Covid-19 cases. Also, different smartphone usage variables show different typical time delays. In summary, the




(a) Performance with LR classifier. (b) Performance with SVM classifier. (c) Performance with Xgboost classifier.

Fig. 10. Covid-19 outbreak stage inferences with different time delays.

correlations between smartphone usage behavior and daily confirmed cases present a high potential of using smartphone usage for daily outbreak stage inference of Covid-19.

## IV. Inference of Outbreak Stages

In this section, we study the second research problem, i.e., whether we can use smartphone usage data, e.g., CPU usage, memory usage, and network connections, to infer the outbreak stages of Covid-19. The outbreak stages reflect different severities of the pandemic. Specifically, we try to determine two points, i.e., the typical time delay of stage inference using smartphone usage data and the performance of different smartphone usage features in Covid-19 stage inference. Also, to further improve inference performance, we propose an embedding mechanism to fuse different smartphone usage behavior features.

### A. Inference Settings

Recalling Fig. 2, we can witness that the outbreak of Covid-19 has shown three stages from March 1, 2020, to April 30, 2020. First, the dates from February 1, 2020, to March 1, 2020, are the early stage of Covid-19, with only a few cases appearing. Second, during the dates from March 1, 2020, to April 1, 2020, the daily confirmed cases increased dramatically. Third, on the dates after April 1, 2020, the increasing trend of Covid-19 cases is stable. Therefore, we label Covid-19 outbreak stages with three classes, i.e., early, dramatic, and stable. By doing so, the inference problem is converted into a 3-class classification problem. Specifically, we infer the outbreak stages of one day by using its diurnal sequences of different smartphone usage behavior, including CPU usage, memory usage, WiFi usage, and network switches. Also, to evaluate the performance, we use Macro-F1 and Micro-F1 as metrics. Macro-F1 treats all classes equally, computing the F1-score independently for each class and then taking the average. Alternatively, Micro-F1 aggregates the contributions of all classes to compute the average F1-score. The higher the value of Macro-F1 and Micro-F1, the better the performance. For all experiments, we obtain the results by employing a five-fold cross-validation policy on our dataset.

### B. Delay Analysis of Stage Inference

As we have discussed in Section III, users' smartphone usage behavior can reflect their physical activities and the outbreak stages of Covid-19. However, the reflection may not be immediately expressed by the daily cases of Covid-19 due to the incubation period and diagnosis delay. Hence, we explore the typical time delay of stage inference. Specifically, we infer the outbreak stage of one day by utilizing the smartphone usage features of the days before it. We use inference performance to evaluate the correlations between smartphone usage and Covid-19 trends. In other words, better performance indicates a higher correlation. Notably, different from the Pearson correlation, the task of inference can also reveal nonlinear correlations. In practice, we conduct the inference with the three most commonly used classification algorithms, logistic regression (LR) [19], support vector machine (SVM) [20] and Xgboost [21]. We infer the outbreak stages of one day by concatenating all behavior types' diurnal sequences, including CPU usage, memory usage, WiFi usage, and network switches.

We show the results in Fig. 10. The LR classifier has poor performance, and F1 scores fluctuate on different delays. That is because the LR classifier only uses a logistic function to model the correlation, which is more susceptible to outliers tampering with the performance. Therefore, it is hard to capture the relations between smartphone usage features and Covid-19 outbreak stages with the LR classifier using the real-word dataset that might have noisy data points. Alternatively, as shown in Fig. 10(b) and Fig. 10(c), SVM and Xgboost classifiers have better performance. Also, we can observe that F1 scores achieve the highest value under a delay of 2 or 3 days. This observation confirms that the reflection of users' smartphone usage behavior will emerge in Covid-19 trends with a time delay of a few days, further validating our analysis in Section III-C.

### C. Performance of Different Usage Features

Next, we evaluate the performance of different smartphone usage features and their combinations for the Covid-19 outbreak stage inference. Specifically, we explore four types of smartphone usage features, i.e., CPU usage (CPU), memory usage (Mem), WiFi usage (WiFi), and network switches (Net). We combine a set of features by concatenating them together. We perform the inference with the Xgboost classifier. The performance of different combinations of features is shown in Table V.

For the inference with a single feature, the performance of using network switches is the best, indicating that users' mobility intensity is most relevant to the Covid-19 status. Meanwhile, WiFi and memory usage achieve relatively good performance, implying that WiFi and memory usage also

TABLE V
INFERENCE PERFORMANCE WITH DIFFERENT FEATURES.

| Features | Macro-F1 | Micro-F1 |
|---|---|---|
| CPU | 0.590 | 0.584 |
| Mem | 0.702 | 0.697 |
| WiFi | 0.713 | 0.708 |
| Net | 0.757 | 0.753 |
| CPU+Mem | 0.666 | 0.663 |
| CPU+WiFi | 0.722 | 0.719 |
| CPU+Net | 0.609 | 0.685 |
| Mem+WiFi | 0.683 | 0.674 |
| Mem+Net | 0.735 | 0.730 |
| WiFi+Net | 0.707 | 0.696 |
| CPU+Mem+WiFi | 0.716 | 0.708 |
| CPU+WiFi+Net | 0.715 | 0.707 |
| Net+Mem+WiFi | 0.739 | 0.730 |
| CPU+Mem+Net | 0.766 | 0.764 |
| CPU+Mem+Net+WiFi | 0.733 | 0.721 |

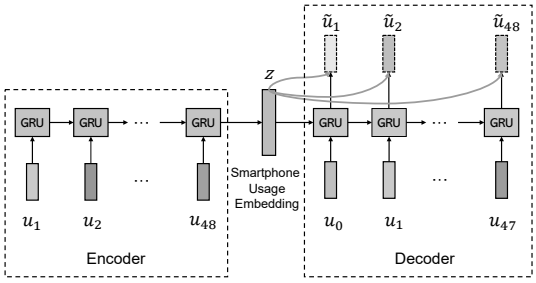



Fig. 11. Seq2Seq model for smartphone usage embedding.

reflect crucial human behavior related to Covid-19. In contrast, the CPU usage feature is less related and with the lowest inference performance. These inference results are consistent with our findings in Section III. As for the inference with multiple features, the performance is not simply a superposition of single features' performance. In terms of Table V, the best performance is achieved by using CPU, memory, and WiFi usage. However, it only achieves the F1 score of around 0.76, slightly higher than when merely using network switches. Also, most cases of using multiple features have lower performance than simply using network switches. These results reveal that simple concatenation is insufficient to fuse different behavior data, motivating us to develop a better fusion mechanism to explore different features effectively.

### *D. Smartphone Usage Behavior Embedding*

In this section, we propose an embedding model to fuse different smartphone usage behavior effectively. Given a day, we first construct a diurnal smartphone usage feature sequence $\{u_i\}_{i=1}^{48}$, where $u_i$ is a vector containing all four usage features in the $i$-th timeslot of the day. We then utilize a Seq2Seq [22] model to learn an embedding from the diurnal sequence. As shown in Fig. 11, the model consists of an encoder and a decoder, which are implemented with a GRU network [22]. The sequence $\{u_i\}_{i=1}^{48}$ is fed into the encoder to obtain an

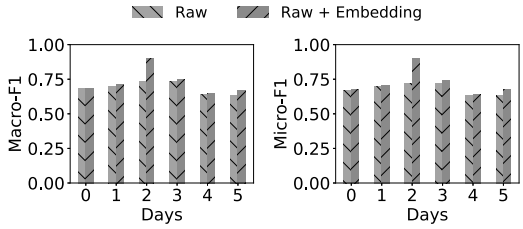



Fig. 12. Outbreak stage inferences with embeddings.

encoding vector of $z$. Then, $z$ and a shifted usage sequence $\{u_i\}_{i=0}^{47}$ are fed into the decoder to reconstruct the original sequence, where $u_0$ is a vector that contains all 1. Moreover, to encode comprehensive information in vector $z$, we engage $z$ in the reconstruction. Formally, the $i$-th unit of the decoder takes $u_{i-1}$ as input and outputs hidden state $\hat{h}_i$ , we infer $\hat{u}_i$ as,

$$\hat{u}_i = \sigma(W[\hat{h}_i, z] + b), \tag{1}$$

where $[,]$ is the concatenating operation, $\sigma$ is the sigmoid activating function, $W$ and $b$ are trainable parameters. Finally, we train the model by minimizing the reconstruction loss,

$$\mathcal{L} = \sum_{i=1}^{48} |\hat{u}_i - u_i|^2. \tag{2}$$

In our experiment, we train the model with the Adam optimizer with a learning rate of 0.0001. The batch size is set as the number of sequences, and we train the model for 200 epochs. By doing so, we obtain a usage embedding vector for each day. To evaluate whether the embedding fuses different usage features better, we conduct the inference on the original features (Raw) and the original features concatenated with the learned embeddings (Raw + Embedding). We again use the Xgboost classifier as the inference model.

We compare the performance with embeddings, as shown in Fig. 12. We can observe that, by combining with embeddings, we improve the entire performance under different delay settings. Especially when the delay is set as two days, the performance of raw features combined with embeddings reaches around 0.87 for both Macro-F1 and Micro-F1, which has an over 20% improvement compared with the best performance of only using raw features. These results demonstrate that the learned embeddings fuse multiple features more effectively indeed.

## V. DISCUSSION AND LIMITATION

In this paper, we have investigated the impact of Covid-19 on smartphone usage based on a real-world dataset. However, the number of users involved in our dataset is not very large, i.e., covering 425 users, which is a limitation of our work. The limited number of users involved may threaten the representativeness of our conclusion. To alleviate the influence caused by a limited number of users, we have taken several adequate measures in our work. For example, we compared the distribution of variables instead of the average and median. We also used the p-value to verify statistical significance.

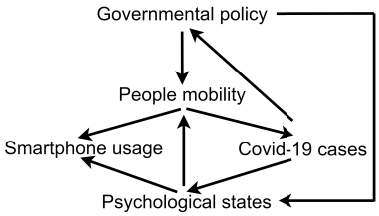



Fig. 13. A potential causality diagram of smartphone usage and Covid-19 cases.

Although we have examined the correlation between smartphone usage behavior and Covid-19 cases, their causality relationship still needs further exploration. In Fig. 13, we depict a potential causality diagram of smartphone usage and Covid-19 cases. People mobility and psychological state serve as a confounder and mediator connecting smartphone usage and Covid-19 cases, respectively. Smartphone usage is directly affected by mobility and can act as a mobility indicator. Also, smartphone usage is still affected by the psychological states of users [13]. Meanwhile, the causation between people mobility and Covid-19 cases is bidirectional. On the one hand, frequent people mobility will trigger new Covid-19 cases. On the other hand, Covid-19 will affect people's mobility through governmental policies and their psychological states. Therefore, the causation between smartphone usage and Covid-19 cases might be complex. As for checking the potential causality diagram we proposed, we leave it to future work.

## VI. Related Work

Many previous studies have focused on characterizing smartphone usage behavior. Shafiq *et al.* [23] presented the diurnal pattern of smartphone network usage from various granularities, i.e., bytes, packets, flows, and users. Peltonen *et al.* [24] collected a one-year smartphone usage dataset from 25,323 users distributed in 44 countries. They then studied how cultural features affect users' smartphone usage behavior. Srinivasan *et al.* [25] indicated that smartphone usage behavior profoundly depends on contextual information. For example, users use more WiFi connections at home. Moreover, Van Canneyt *et al.* [26] exhibited that the occurrence of special events, e.g., New year's day, UEFA European Championship, will disrupt users' normal smartphone usage patterns. These existing studies demonstrate that users' smartphone usage behavior will be sensitively impacted by diverse contextual factors, including time, locations, and big events, which inspired us to investigate how the outbreak of Covid-19 affects the smartphone usage behavior.

Also, some studies pointed out the strong link between smartphone usage behavior and users' physical attributes and activities. Zhao *et al.* [27] analyzed one month of smartphone usage data collected from 106,762 users. They then discovered 382 distinct types of users based on their usage behavior. Also, they gave each cluster a meaningful label, such as night communicators, evening learners, and financial users. Do *et al.* [28] represented users' smartphone usage traces in one day as a bag-of-words, where one word refers to a smartphone usage record with time features. They then applied an author-topic model to infer the underlying structure of users' physical activities. Similarly, Li *et al.* [29] leveraged smartphone app usage data to identify users' daily activities. These studies demonstrated that users' physical activities profoundly shape smartphone usage behavior, which shed light on using smartphone usage data to reflect human activities and further infer Covid-19 outbreak stages.

Smartphone usage behavior is still affected by users' psychological states. For example, Saeb *et al.* [30] explored smartphone sensors' data, like accelerometer, screen, GPS, and WiFi, which help estimate the depression and anxiety of users. Their methods can also be applied to our dataset, allowing us to detect the depression of users. During the Covid-19 crisis, we need to pay more attention to mental health in the population. Covid-19 may trigger psychiatric disorders of people [31]. Elhai *et al.* analyzed gaming disorder severity [32] and anxiety symptoms [33] during Covid-19. Moreover, Montag *et al.* [13] pointed out that we can leverage smartphone data to detect population mental states in real-time to help fight the Covid-19 pandemic. They also developed an app [34] for social scientists, which tracks smartphone usage data by combining self-report data with objectively recorded data. In practice, conducting population-scale digital phenotyping might be challenging due to the lack of sufficient labeled data. In that case, label-less learning should be a helpful technology. For example, Chen *et al.* [35] proposed a label-less learning for emotion cognition on a large-scale.

Some studies also analyzed physical activities during Covid-19 by using smartphone app usage and sensory data. Norbury *et al.* [36] discovered a positive relation between social app usage and total footsteps (obtained from sensory data) during the lockdown due to Covid-19. Couture *et al.* [37] investigated county-to-county movements based on the GPS data collected from smartphones. Unlike the above studies, our work directly investigates the relation between smartphone usage and the Covid-19 outbreak.

## VII. Conclusion

We conduct the first comprehensive study of the impact of Covid-19 on smartphone usage. Specifically, our analysis covers the mobile users in North America with six-month smartphone usage records from November 2019 to April 2020. Overall, our findings indicate that users' smartphone usage indeed changes across the outbreak of Covid-19. However, the outbreak has different effects on different usage behavior in terms of changing trends, diurnal patterns, and correlations. Also, we demonstrate the potential of using smartphone usage data to infer the outbreak stages, achieving over 0.8 for both Macro-F1 and Micro-F1. Our findings provide a novel application of smartphone usage data and explore their values for fighting against the epidemic.

## Acknowledgments

This research has been supported in part by project 16214817 from the Research Grants Council of Hong Kong, project FP805 from HKUST, the 5GEAR project, the FIT

project and the CBAI (Crowdsourced Battery Optimization AI for a Connected World, grant No. 1319017) project from the Academy of Finland, the National Natural Science Foundation of China under U1936217, 61971267, 61972223, 61941117, 61861136003, Beijing Natural Science Foundation under L182038, Beijing National Research Center for Information Science and Technology under 20031887521, and research fund of Tsinghua University - Tencent Joint Laboratory for Internet Innovation Technology.

## REFERENCES

[1] T. P. Velavan and C. G. Meyer, "The covid-19 epidemic," *Tropical medicine & international health*, vol. 25, no. 3, p. 278, 2020.

[2] W. McKibbin and R. Fernando, "The economic impact of covid-19," *Economics in the Time of COVID-19*, vol. 45, 2020.

[3] S. Li, Y. Wang, J. Xue, N. Zhao, and T. Zhu, "The impact of covid-19 epidemic declaration on psychological consequences: a study on active weibo users," *International journal of environmental research and public health*, vol. 17, no. 6, p. 2032, 2020.

[4] M. Chen, M. Li, Y. Hao, Z. Liu, L. Hu, and L. Wang, "The introduction of population migration to seiar for covid-19 epidemic modeling with an efficient intervention strategy," *Information Fusion*, vol. 64, pp. 252–258, 2020.

[5] T. Li, M. Zhang, H. Cao, Y. Li, S. Tarkoma, and P. Hui, "" what apps did you use?": Understanding the long-term evolution of mobile app usage," in *Proceedings of The Web Conference 2020*, 2020, pp. 66–76.

[6] Statista, "https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/," 2020.

[7] M. Chen, Y. Jiang, N. Guizani, J. Zhou, G. Tao, J. Yin, and K. Hwang, "Living with i-fabric: smart living powered by intelligent fabric and deep analytics," *IEEE Network*, vol. 34, no. 5, pp. 156–163, 2020.

[8] T. Yarkoni, "Psychoinformatics: New horizons at the interface of the psychological and computing sciences," *Current Directions in Psychological Science*, vol. 21, no. 6, pp. 391–397, 2012.

[9] T. R. Insel, "Digital phenotyping: technology for a new science of behavior," *Jama*, vol. 318, no. 13, pp. 1215–1216, 2017.

[10] H. Baumeister and C. Montag, *Digital Phenotyping and Mobile Sensing*. Springer, 2019.

[11] A. Markowetz, K. Błaszkiewicz, C. Montag, C. Switala, and T. E. Schlaepfer, "Psycho-informatics: big data shaping modern psychometrics," *Medical hypotheses*, vol. 82, no. 4, pp. 405–411, 2014.

[12] J. J. Van Bavel, K. Baicker, P. S. Boggio, V. Capraro, A. Cichocka, M. Cikara, M. J. Crockett, A. J. Crum, K. M. Douglas, J. N. Druckman *et al.*, "Using social and behavioural science to support covid-19 pandemic response," *Nature human behaviour*, vol. 4, no. 5, pp. 460–471, 2020.

[13] C. Montag, P. Dagum, J. D. Elhai *et al.*, "On the need for digital phenotyping to obtain insights into mental states in the covid-19 pandemic," *Digital Psychology*, vol. 1, no. 2, pp. 40–42, 2020.

[14] B. F. Maier and D. Brockmann, "Effective containment explains subexponential growth in recent confirmed covid-19 cases in china," *Science*, vol. 368, no. 6492, pp. 742–746, 2020.

[15] J. Neyman and E. S. Pearson, "The testing of statistical hypotheses in relation to probabilities a priori," in *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 29, no. 4. Cambridge University Press, 1933, pp. 492–510.

[16] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang, and J. Wang, "Characterizing geospatial dynamics of application usage in a 3g cellular data network," in *2012 Proceedings IEEE INFOCOM*. IEEE, 2012, pp. 1341–1349.

[17] N. Ghahramani and C. Brakewood, "Trends in mobile transit information utilization: An exploratory analysis of transit app in new york city," *Journal of Public Transportation*, vol. 19, no. 3, p. 9, 2016.

[18] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[19] D. G. Kleinbaum, K. Dietz, M. Gail, M. Klein, and M. Klein, *Logistic regression*. Springer, 2002.

[20] J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural processing letters*, vol. 9, no. 3, pp. 293–300, 1999.

[21] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[22] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[23] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang, and J. Wang, "Geospatial and temporal dynamics of application usage in cellular data networks," *IEEE Transactions on mobile computing*, vol. 14, no. 7, pp. 1369–1381, 2014.

[24] E. Peltonen, E. Lagerspetz, J. Hamberg, A. Mehrotra, M. Musolesi, P. Nurmi, and S. Tarkoma, "The hidden image of mobile apps: Geographic, demographic, and cultural factors in mobile usage," in *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services*, 2018, pp. 1–12.

[25] V. Srinivasan, S. Moghaddam, A. Mukherji, K. K. Rachuri, C. Xu, and E. M. Tapia, "Mobileminer: Mining your frequent patterns on your phone," in *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing*, 2014, pp. 389–400.

[26] S. Van Canneyt, M. Bron, A. Haines, and M. Lalmas, "Describing patterns and disruptions in large scale mobile app usage data," in *Proceedings of the 26th International Conference on World Wide Web Companion*, 2017, pp. 1579–1584.

[27] S. Zhao, J. Ramos, J. Tao, Z. Jiang, S. Li, Z. Wu, G. Pan, and A. K. Dey, "Discovering different kinds of smartphone users through their application usage behaviors," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2016, pp. 498–509.

[28] T.-M.-T. Do and D. Gatica-Perez, "By their apps you shall understand them: mining large-scale patterns of mobile phone usage," in *Proceedings of the 9th international conference on mobile and ubiquitous multimedia*, 2010, pp. 1–10.

[29] T. Li, Y. Li, M. A. Hoque, T. Xia, S. Tarkoma, and P. Hui, "To what extent we repeat ourselves? discovering daily activity patterns across mobile app usage," *IEEE Transactions on Mobile Computing*, 2020.

[30] S. Saeb, E. G. Lattie, K. P. Kording, and D. C. Mohr, "Mobile phone detection of semantic location and its relationship to depression and anxiety," *JMIR mHealth and uHealth*, vol. 5, no. 8, p. e112, 2017.

[31] A. Schimmenti, J. Billieux, and V. Starcevic, "The four horsemen of fear: An integrated model of understanding fear experiences during the covid-19 pandemic," *Clinical Neuropsychiatry*, vol. 17, no. 2, pp. 41–45, 2020.

[32] J. D. Elhai, D. McKay, H. Yang, C. Minaya, C. Montag, and G. J. Asmundson, "Health anxiety related to problematic smartphone use and gaming disorder severity during covid-19: Fear of missing out as a mediator," *Human Behavior and Emerging Technologies*, vol. 3, no. 1, pp. 137–146, 2021.

[33] J. D. Elhai, H. Yang, D. McKay, and G. J. Asmundson, "Covid-19 anxiety symptoms associated with problematic smartphone use severity in chinese adults," *Journal of Affective Disorders*, vol. 274, pp. 576–582, 2020.

[34] C. Montag, H. Baumeister, C. Kannen, R. Sariyska, E.-M. Meßner, and M. Brand, "Concept, possibilities and pilot-testing of a new smartphone application for the social and life sciences to study human behavior including validation data from personality psychology," *J—Multidisciplinary Scientific Journal*, vol. 2, no. 2, pp. 102–115, 2019.

[35] M. Chen and Y. Hao, "Label-less learning for emotion cognition," *IEEE transactions on neural networks and learning systems*, vol. 31, no. 7, pp. 2430–2440, 2019.

[36] A. Norbury, S. H. Liu, J. J. Campaña-Montes, L. Romero-Medrano, M. L. Barrigón, E. Smith, A. Artés-Rodríguez, E. Baca-Garcia, and M. M. Perez-Rodriguez, "Social media and smartphone app use predicts maintenance of physical activity during covid-19 enforced isolation in psychiatric outpatients," *Molecular psychiatry*, pp. 1–11, 2020.

[37] V. Couture, J. I. Dingel, A. E. Green, J. Handbury, and K. R. Williams, "Measuring movement and social contact with smartphone data: a real-time application to covid-19," National Bureau of Economic Research, Tech. Rep., 2020.

**Tong Li** received the B.S. degree and M.S. degree in communication engineering from Hunan University, China, in 2014 and 2017. At present, he is a dual Ph.D. student at the Hong Kong University of Science and Technology and the University of Helsinki. His research interests include data mining and machine learning, especially with applications to mobile big data and urban computing. He is an IEEE student member.

**Mingyang Zhang** is currently pursuing the Ph.D. degree with the department of Computer Science and Engineering, Hong Kong University of Science and Technology, within the System and Media Laboratory (SymLab). He received the B.S. degrees in electronic engineering from Tsinghua University, Beijing, China, in 2018. His research interests include spatiotemporal data mining, urban computing.

**Yong Li** (M'09-SM'16) is currently a Tenured Associate Professor of the Department of Electronic Engineering, Tsinghua University. He received the Ph.D. degree in electronic engineering from Tsinghua University in 2012. His research interests include machine learning and big data mining, particularly, automatic machine learning and spatial-temporal data mining for urban computing, recommender systems, and knowledge graphs. Dr. Li has served as General Chair, TPC Chair, SPC/TPC Member for several international workshops and conferences, and he is on the editorial board of two IEEE journals. He has published over 100 papers on first-tier international conferences and journals, including KDD, WWW, UbiComp, SIGIR, AAAI, TKDE, TMC etc, and his papers have total citations more than 8300. Among them, ten are ESI Highly Cited Papers in Computer Science, and five receive conference Best Paper (run-up) Awards. He received IEEE 2016 ComSoc Asia-Pacific Outstanding Young Researchers, Young Talent Program of China Association for Science and Technology, and the National Youth Talent Support Program.

**Eemil Lagerspetz** is a Postdoctoral Researcher at the Department of Computer Science, University of Helsinki. He completed his Ph.D. in Computer Science at the University of Helsinki in 2014. His research interests include large-scale data analysis (Big Data), edge computing, ubiquitous computing, and energy efficiency.

**Sasu Tarkoma** (SM'12) received the MSc and PhD degrees in computer science from the Department of Computer Science, University of Helsinki. He is a Professor of Computer Science at the University of Helsinki, and Head of the Department of Computer Science. He has authored 4 textbooks and has published over 200 scientific articles. His research interests are Internet technology, distributed systems, data analytics, and mobile and ubiquitous computing. He is Fellow of IET and EAI. He has nine granted US Patents. His research has received several Best Paper awards and mentions, for example at IEEE PerCom, IEEE ICDCS, ACM CCR, and ACM OSR.

**Pan Hui** (SM'14-F'18) received his Ph.D. degree from the Computer Laboratory at University of Cambridge, and both his Bachelor and MPhil degrees from the University of Hong Kong.

He is the Nokia Chair Professor in Data Science and Professor of Computer Science at the University of Helsinki. He is also the director of the HKUST-DT Systems and Media Lab at the Hong Kong University of Science and Technology. He was a senior research scientist and then a Distinguished Scientist for Telekom Innovation Laboratories (T-labs) Germany and an adjunct Professor of social computing and networking at Aalto University. His industrial profile also includes his research at Intel Research Cambridge and Thomson Research Paris. He has published more than 300 research papers and with over 17,500 citations. He has 30 granted and filed European and US patents in the areas of augmented reality, data science, and mobile computing. He has been serving on the organising and technical program committee of numerous top international conferences including ACM SIGCOMM, MobiSys, IEEE Infocom, ICNP, SECON, IJCAI, AAAI, ICWSM and WWW. He is an associate editor for the leading journals IEEE Transactions on Mobile Computing and IEEE Transactions on Cloud Computing. He is an IEEE Fellow, an ACM Distinguished Scientist, and a member of the Academia Europaea.

# Paper III

Tong Li, Mingyang Zhang, Hancheng Cao, Yong Li, Sasu Tarkoma, and Pan Hui

**"What Apps Did You Use?": Understanding the Long-term Evolution of Mobile App Usage**

In *Proceedings of The Web Conference 2020*, pp.66–76.

III

# "What Apps Did You Use?": Understanding the Long-term Evolution of Mobile App Usage

Tong Li
University of Helsinki, Finland
HKUST, Hong Kong
t.li@connect.ust.hk

Mingyang Zhang
HKUST, Hong Kong
mzhangbj@ust.hk

Hancheng Cao
Stanford University, United States
hanchcao@stanford.edu

Yong Li
Tsinghua University, China
liyong07@tsinghua.edu.cn

Sasu Tarkoma
University of Helsinki, Finland
sasu.tarkoma@cs.helsinki.fi

Pan Hui
University of Helsinki, Finland
HKUST, Hong Kong
panhui@cse.ust.hk

## ABSTRACT

The prevalence of smartphones has promoted the popularity of mobile apps in recent years. Although significant effort has been made to understand mobile app usage, existing studies are based primarily on short-term datasets with limited time span, e.g., a few months. Therefore, many basic facts about the long-term evolution of mobile app usage are unknown. In this paper, we study how mobile app usage evolves over a long-term period. We first introduce an app usage collection platform named carat, from which we have gathered app usage records of 1,465 users from 2012 to 2017. We then conduct the first study on the long-term evolution processes on a macro-level, i.e., app-category, and micro-level, i.e., individual app. We discover that, on both levels, there is a growth stage enabled by the introduction of new technologies. Then there is a plateau stage caused by high correlations between app categories and a pareto effect in individual app usage, respectively. Additionally, the evolution of individual app usage undergoes an elimination stage due to fierce intra-category competition. Nevertheless, the diverseness of app-category and individual app usage exhibit opposing trends: app-category usage assimilates while individual app usage diversifies. Our study provides useful implications for app developers, market intermediaries, and service providers.

## CCS CONCEPTS

• **Human-centered computing** → **Empirical studies in ubiquitous and mobile computing**.

## KEYWORDS

App usage; App categories; Google play; Long-term evolution

**ACM Reference Format:**
Tong Li, Mingyang Zhang, Hancheng Cao, Yong Li, Sasu Tarkoma, and Pan Hui. 2020. "What Apps Did You Use?": Understanding the Long-term Evolution of Mobile App Usage. In *Proceedings of The Web Conference 2020 (WWW '20), April 20–24, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3366423.3380095


*WWW '20, April 20–24, 2020, Taipei, Taiwan*

ACM ISBN 978-1-4503-7023-3/20/04.
https://doi.org/10.1145/3366423.3380095

## 1 INTRODUCTION

Since the introduction of the first Android-based smartphone the 'HTC Dream' in 2007 [28, 29], the usage of smartphones has significantly evolved over the last ten years, extending from essential communications to various applications, e.g., ordering food, shopping online, and managing health [1, 10, 19, 20]. Such diverse demands are supported by mobile apps, i.e., software applications designed to run on mobile devices [30]. To satisfy various user requirements, Google Play and Apple Store, i.e., the official Android and iOS app markets, provide a wide range of apps for mobile users. As of 2019, the number of apps in app markets has reached 2.7 million [21], and the app economy is estimated to grow to 6.3 trillion dollars by 2021 [25]. Such a vast market attracts and motivates app developers, market intermediaries, and service providers to better develop, disseminate, and deliver mobile apps.

In recent years, countless efforts have been made to study mobile app usage. Existing studies principally explore users' static behavior based on short-term datasets collected in a given time window ranging from one week [5, 23, 26], several months [6, 14, 15, 27, 32], and up to one year [18]. However, existing research falls short in studying the long-term evolution of users' app usage since they are limited by the short time span of their datasets.

Every year, mobile users will acquire new generations of smartphones, technologies, and apps. Both smartphone hardware and software are significantly advancing over time. As a result, users' mobile app usage will correspondingly evolve. The evolution of app usage makes some previous findings based on short-term datasets out-of-date and no longer applicable. Hence, in this dramatically changing world, studying evolutionary trends and extracting general laws behind mobile app usage enables us to gain insight beyond short-term observations. However, up to now, *many basic facts about the long-term dynamics of mobile app usage are unknown.* Therefore, exploring the long-term evolution of mobile app usage is essential.

Understanding the long-term evolution of mobile app usage is critical for industry because understanding such mechanisms can enable companies to effectively improve user experience, enhance apps' competitive power, and grasp market opportunities during development. For instance, for market intermediaries and service providers, analyzing the evolution processes can help with tracking app preferences of users, monitoring the maturity of different

app categories, and forecasting the future flourishing apps. They can further draw upon such insights to optimize the decisions for maintaining and improving the entire app market. Moreover, the long-term evolution study can help app developers grasp general laws behind the long-lived app categories and apps. In this way, app developers can make better decisions for developing and releasing apps and improving the competitive power of their apps.

In this paper, we make the first effort towards understanding the long-term evolution of mobile app usage. Specifically, our study details how users' usage changes over time at both s macro-level and micro-level, i.e., app categories and apps, respectively. To this end, we have collected a long-term app usage dataset by leveraging an Android-based platform called Carat. The dataset covers around 1,500 users in over 80 countries and their app usage records for six years from 2012 to 2017 (Section 2). We first use the dataset to make a macro-level analysis on the evolution of app-category usage in terms of four metrics, i.e., the number of used app categories, the diversity of app-category usage, the popularity of app categories and the correlations of app categories (Section 3). Next, we extend our analysis to the micro-level, i.e., individual app granularity. We characterize the evolution of app usage based on similar metrics. Comparing the evolving trends between the macro-level and micro-level, we delve into the reasons and summarize the general laws of long-term usage evolution (Section 4). At last, we explore the implications of our findings for app developers, market intermediaries, and service providers (Section 5). Among the many insightful results and observations, the following are the most prominent.

- The long-term usage evolution of app-categories and apps exhibits different processes. A complete usage evolution of an app-category undergoes two stages, i.e., a growth stage and a plateau stage. However, apart from the above two stages, apps have one more additional stage, i.e., an elimination stage.
- The diversity of app-category usage declines over time due to non-decreasing usage evolution processes. However, the diversity of app usage increases greatly, showing large differences between mobile users at the app level.
- The app usage shows a typical *Pareto effect*. A small group of apps dominate usage in both the entire app market and individual app categories. Also, we identify 12 essential apps of different functionality for smartphones.
- The release of new technologies will trigger the growth stage for both app categories and apps. This increasing trend will not be influenced by the maturity of app categories and the *Pareto effect*.
- The fierce intra-competition of apps results in an elimination stage of app usage and the decrease in correlations between apps in the same category. Also, the evolution of app usage will be affected by the degree of maturity of the app's category.

## 2 LONG-TERM MOBILE APP USAGE DATASET

### 2.1 Data Collection and Basic Analysis

It is difficult to collect a long-term app usage dataset for two main reasons. 1) For privacy and safety concerns, mobile users are hesitant to let a third party collect their data, especially for long-term collection. 2) In the research community scholars can recruit volunteers and use a monitoring app to collect app usage records. However, executing such a long-term study is costly in terms of both human labor and capital.

To overcome the above difficulties, we designed an Android-based long-term data collection platform called Carat. Carat is a mobile app that can record users' smartphone usage data automatically. First, to eliminate user privacy concerns, the user will be informed of all data collection items when installing Carat in the End-user License Agreement (EULA). We will not collect any personal information. Furthermore, the data-gathering part of the platform is open-source[1] thus users can examine it easily. Second, to reduce the expense of long-term data collection, we motivated users to keep using Carat for long time periods. To this end, we designed Carat as not only a simple data collection app but also a collaborative energy diagnosis app. Carat can provide personalized recommendations for improving smartphone battery life. Carat gathers a data sample every time the battery level changes by 1%, as allowed by the Android system. Each data sample contains a list of apps being used, a user-specific identifier, battery level, timestamp, time zone, mobile country code, and mobile network type. As of now, the Carat platform has gathered data from over 30,000 mobile users from over 100 countries[2].

As our focus is on studying the long-term evolution of mobile app usage, we select users with more than three years of records and define them as long-term users. In the end, we obtain 1,465 long-term users with 12,457,867 records starting from January 2012 to December 2017. Since the user may uninstall and reinstall Carat during the data collection period, the number of long-term users changes over time, i.e., 2012 (965 users), 2013 (836 users), 2014 (1,010 users), 2015 (1,197 users), 2016 (1,114 users), 2017 (916 users). Also, we crawl the apps' category information directly from Google Play. Table 1 summarizes the dataset used in our analysis.

Unlike previous works whose mobile app usage datasets are collected only from one city [8, 23, 32] or one country [31], our users are distributed across the world. The long-term users are from 87 countries. We use both time zones and mobile country codes to determine the country of users. The majority of the users are based in the USA (360 users). Also, there are many users in Finland (278 users), India (60 users), Germany (52 users), and the UK (49 users). The diversity of the Carat dataset enables us to capture the evolving trends of the worldwide app market, improving the representativeness of our analysis results.

### 2.2 Ethical Considerations

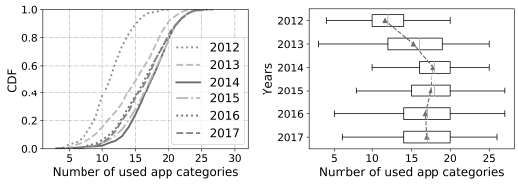
We are very aware of the privacy implications of using the collected data for research. We have taken adequate measures to safeguard the privacy of the involved mobile users. As mentioned, we do not collect any personal information from users. Also, the data-gathering part of Carat is open-source. The mobile users are informed of the data collection and management procedures and grant their consent from their devices. The dataset is stored in a secure local server protected by strict authentication mechanisms

[1] http://carat.cs.helsinki.fi/

[2] Sample of our collected data available at https://www.cs.helsinki.fi/group/carat/data-sharing/.

**Table 1: Summary of our dataset.**

| # Users | # Records | # Apps | # App Categories | Attributes | Date | Area |
|---|---|---|---|---|---|---|
| 1,465 | 12,457,867 | 25,068 | 32 | Apps, time zone, timestamp, mobile network type | 01/2012 - 12/2017 | Over 80 countries |




(a) CDF of the number of app categories used by each unique user.

(b) Distribution of used app categories across different years.

**Figure 1: Evolution of app-category usage across six years.**

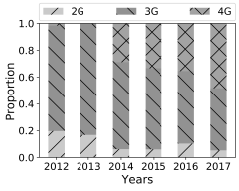



**Figure 2: Proportions of mobile network types.**

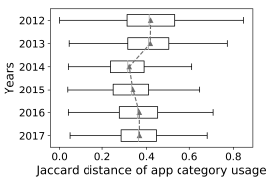



**Figure 3: Jaccard distance of app-category usage.**

and firewalls. A user-specific identifier is randomly generated when a user first installs Carat. We only have users' mobile country codes rather than sensitive location information. Hence, we cannot associate user-specific identifiers with physical users. All researchers are regulated by a strict non-disclosure agreement to access the data. This work has received approval from all authors' local institutions.

## 3 EVOLUTION OF APP-CATEGORY USAGE

### 3.1 Number of App Categories

We begin our analysis by investigating the most intuitive metric of app-category usage, i.e., the number of app categories used by each user during a given year. Figure 1(a) presents the Cumulative Distribution Function (CDF) of the number of used app categories for all long-term users from 2012 to 2017. We observe that the evolution of app-category usage undergoes two stages.

- **Stage one (2012 - 2014).** In this stage, the number of app categories used by each user increased significantly. The increasing trend suggests that during this stage, *smartphones were endowed with more functions, and people started using smartphones in more diverse activities.* In 2012, over 80% of users used less than 14 app categories, while the number increased to 20 by 2014. Moreover, the average number of used app categories expanded from 12 in 2012 to 17 in 2014.
- **Stage two (2014 - 2017).** During this stage, the number of used app categories remained stable over time, which implies that *both smartphones' functions and users' usage at the app-category granularity became steady.* As depicted in Figure 1(a), the CDF curves for years from 2014 to 2017 are very close to each other. The average number of used app categories was around 17.

Alternatively, to better illustrate the changes in the number of used app categories, we depict the distributions across different years using box-plots in Figure 1(b). Compared with CDF curves, box-plots describe data through their quartiles, enabling us to observe the changes in different groups of data [17]. In the box-plot, candlesticks represent the minimum and the maximum values of the data, while the boxed area contains the values between the 25% and 75% quatiles. The horizontal line depicts the median. The green upper triangle denotes the mean. From 2012 to 2014, the values in the interquartile range, i.e., the boxed area, increased significantly, reinforcing Figure 1(a). However, after 2014, the third quartile is constant, implying the group of users who use relatively more app categories remained stable. Although the first quartile dropped slightly until 2016, there was no discernible change in terms of the average value. Especially in 2016 and 2017, the interquartile range was the same, representing a steady state in users' app-category usage.

One possible reason for the increase in used app categories in stage one is the development of mobile networks. From 2012 to 2014, many countries, including the USA, Finland, the UK, etc., started to deploy fourth-generation mobile networks (4G) [16]. By 2014, 4G mobile networks had been commercialized and used on a large scale. In terms of the mobile network types in our dataset, we present how the proportions of different mobile network types changed from 2012 to 2017 in Figure 2. In our case, 2G and 3G refer to second-generation and third-generation mobile networks, respectively. We can observe that by 2014, around 30% of collected users were using 4G networks, and the fraction grew steadily after that, corresponding to the commercialization of 4G networks. Compared to 3G providing up to 21.6 Mbit/s download rate, 4G networks can support 1 Gbit/s or about 50 times that of 3G. As a result, mobile networks no longer inhibit the usage of latency-sensitive apps and data consuming apps, e.g., online gaming apps, online video apps, and map apps. Therefore, more app categories are widely used by mobile users to facilitate and color their lives. The details of the changes in popularity across different app categories will be discussed in Section 3.3.

### 3.2 Diversity of App-category Usage

We next study the diversity of app-category usage across different users. In 2010, Falaki *et al.* [7] first demonstrated the diversity of smartphone usage and strongly motivated the need for customizing smartphones to different mobile users. Zhao *et al.* [32] illustrated diversity in mobile app usage as well. Hence, we are interested in

analyzing how the diversity of app-category usage changes over time.

We apply Jaccard distance [13] to measure the difference in app-category usage between two users. Jaccard distance is a commonly used metric to measure the similarity between two sets. Denoting $C_a$ and $C_b$ as the sets of app categories used by user $A$ and user $B$, respectively, the Jaccard distance is computed as,

$$J(A,B) = \frac{|C_a \bigcup C_b| - |C_a \bigcap C_b|}{|C_a \bigcup C_b|}. \quad (1)$$

If the two users use the same app categories, i.e., $C_a = C_b$, $J(A,B) = 0$. On the contrary, if the two users use completely different app categories, i.e., $C_a \bigcap C_b = \quad$, $J(A,B) = 1$.

For each year, we compute the Jaccard distance between every two users and illustrate the distributions using box-plots, as shown in Figure 3. We notice that the average pairwise distance, denoted as the green triangle, shows a downtrend. Especially from 2013 to 2014, the average value dropped dramatically from 0.42 to 0.32. Although there was a slight increase after 2014, the average pairwise distance was still much lower than that of 2013. Also, the distribution did not significantly change from 2014 to 2017. The decaying distance reflects that *the diversity of app-category usage declined, and users' requirements for smartphone functions tend to be consistent.* We infer that two reasons led to a decrease in the diversity of app-category usage. First, the development of technologies, including in mobile networks, smartphone hardware, and software, etc., caused more app categories to become popular, and mobile users use similar app categories. For instance, because of the low network throughput and low quality of experience for online gaming, in 2012, only a small group of game fans would use online gaming apps. However, after the large-scale deployment of 4G networks, the quality of experience of mobile online games improved significantly. People become eager to download and play mobile online games. This inference is supported by the increasing popularity of game apps and will be discussed in detail in Section 3.3. Therefore, in this way, users tend to use similar app categories. Second, the app ecosystem pushes mobile users to use similar app categories. With the widespread adoption of mobile apps, a robust app ecosystem has formed, and the correlations of different app categories has become stronger (detailed in Section 3.4). For example, mobile users may share music, games, or books with their friends through social and communication apps. Therefore, their friends have to install the corresponding app categories if they want to open shared content. As time goes by, people will gradually use similar app categories.

## 3.3 Popularity of App Categories

To understand which app categories are more competitive and explore general laws in usage evolution, we next investigate how the popularity of each app category changes over time. In our case, we measure the popularity in terms of unique users, which is the ratio of the users who used that app category to all long-term users. For instance, if one app category has a popularity of 0.9, it means that 90% of long-term users have used at least one app belonging to that app category. Figure 4 shows the popularity of each app category across different years. From 2012 to 2014, there were 26 app categories. In 2015, three new app categories were introduced, i.e., 'Art and design', 'Food and drink', and 'Maps and navigation'.

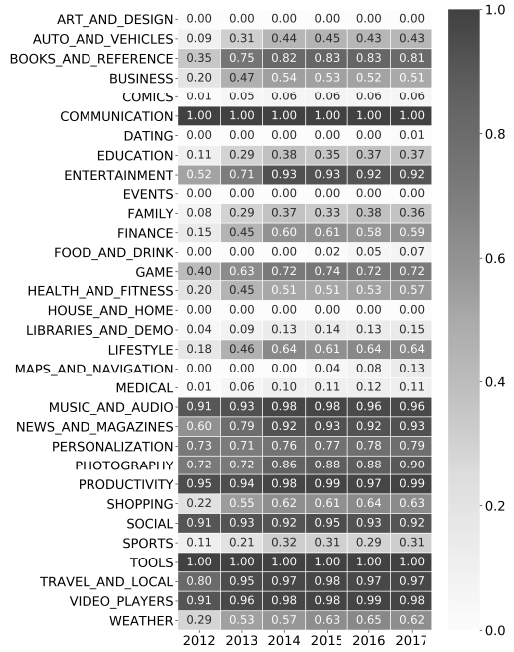



**Figure 4: App category popularity across different years.**

In 2016, there were two new categories, i.e., 'Events' and 'House and home'. In 2017, one new category, 'Dating' appeared. Therefore, in total, we have 32 app categories.

We first focus on the prevalent app categories. We define an app category as prevalent if its popularity is higher than 0.9. The prevalent app categories represent the critical requirements and preferences of mobile users. We discover that there are two types of prevalent app categories distinguished by their evolution processes.

- **Prior prevalent app category**. This type refers to the category whose popularity has exceeded 0.9 since 2012. There are six prior prevalent categories, including 'Communication', 'Music and audio', 'Productivity', 'Social', 'Tools', and 'Video players', which suggests smartphones have already acted as communication devices and multimedia players since 2012.
- **Posterior prevalent app category**. This type refers to the category whose popularity reached 0.9 after 2012, which suggests changes in smartphone roles. There are four posterior prevalent categories, i.e., 'Entertainment', 'News and magazines', 'Photography', and 'Travel and local'.

Compared to prior prevalent categories, posterior prevalent categories are more relevant to life services. The emerging of posterior prevalent categories implies smartphones changed from communication tools to life assistants coloring users' daily lives. This shifting may be caused by the development of technologies in mobile networks, smartphone hardware, and software. For example, as analyzed in Section 3.1, the prevalence of 'Entertainment' might

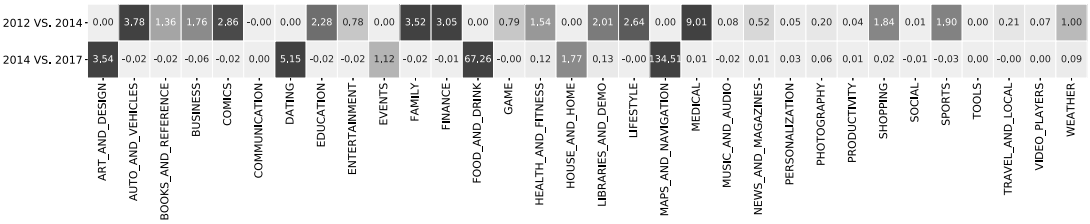
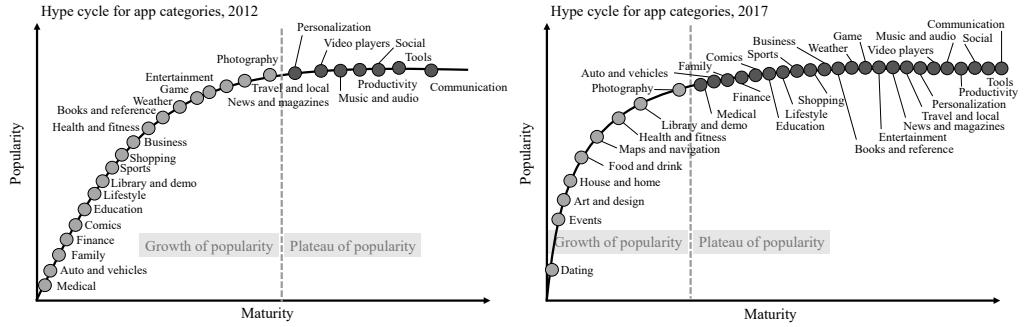



**Figure 5: The growth rates of popularity across different app categories.**




(a) The hype cycle of popularity for different app categories in 2012.

(b) The hype cycle of popularity for different app categories in 2017.

**Figure 6: The evolution of app category popularity.**

be caused by the upgrade of mobile networks. The increment in smartphone screen size may be responsible for the rise in the usage of 'News and magazines' due to the improved reading experience. 'Photography' apps also benefit from the upgrade of smartphone hardware. More powerful CPU and high-resolution cameras enable 'Photography' apps to process and render photos in real-time. Also, we infer that 'Travel and local' apps became prevalent due to the improvement in software services, like recommendations and visualizations.

It is of great importance to study the growth rates of popularity across different app categories for capturing users' preferences during the evolution of the app market. For each app category, we compute its growth rate of popularity during two stages, i.e., 2012-2014 and 2014-2017, respectively. Figure 5 shows the results. From 2012 to 2014, except for prior prevalent app categories, the popularity of other app categories increased. This trend suggests that *the app market for prior prevalent app categories has been mature since before 2012, and the entire app market experienced a boom period from 2012 to 2014.* The 'Medical' category had the highest growth rate during stage one, growing more than nine times. Such a high growth rate for the 'Medical' category verifies our previous inference that smartphones are turning into users' life assistants. Additionally, the popularity of other life-related app categories, like 'Finance', 'Family', 'Shopping', 'Education', and 'lifestyle', increased significantly as well. During stage two, i.e., from 2014 to 2017, newly added categories are concentrated in life services, and their popularity also underwent a significant increase. Especially for 'Food and drink' and 'Maps and navigation', their popularity grew over 67 times and 134 times, respectively. However, with the exception of the newly added categories, the popularity of other app categories stopped rising and became relatively stable during this stage. The stable popularity indicates the app category has become mature and also illustrates users' high reliance on that app category.

In terms of the popularity growth rates across diverse app categories, we present the hype cycles of popularity for app categories in Figure 6. The hype cycle shows the relationship between the maturity of app categories with their popularity. In the hype cycle, we only focus on depicting changes in popularity rather than exact values. Generally, if the app category is more mature then its popularity is more stable. As shown in Figure 6, the evolution of app category popularity undergoes two stages, i.e., growth of popularity and plateau of popularity.

- **Growth of popularity.** In this stage, the popularity of the app category increases. When an app category is newly introduced, it will be at this stage initially. Furthermore, as previously discussed, the development of technologies and smartphone designs, like 4G networks and larger screen sizes, will trigger an increase in multiple app categories' popularity.
- **Plateau of popularity.** In this stage, the popularity of the app category tends to be stable, which suggests that the


**Figure 7: The correlations of app categories across different years.**

market in this app category is mature. For different app categories, their steady popularity is different because they have different potential customers. For instance, the steady popularity for 'Communication', designed for almost all smartphone users, is around 1, while the steady popularity for 'Education' mainly used by students, is only 0.37.

Surprisingly, *there is no discernible decline stage during the popularity evolution of app categories*. We infer that there are three factors that inhibit the formation of a decline stage, i.e., user habits, user communities, and an app ecosystem. **First**, nowadays, people are accustomed to using diverse apps to facilitate their daily lives, e.g., ordering food and shopping online. Meanwhile, an app category contains a group of apps with similar functionality that typically differ from other app categories. Hence, it is hard for one app category to substitute for another. As a result, users' reliance and a category's irreplaceability will push users to continue to use that app category. **Second**, for one app category, its users will form a user community. The community will encourage users to keep using that app category. Taking 'Communication' as an example, if others are used to using 'Communication' apps, like Skype and Whatsapp, to contact you, it is difficult for you to get rid of 'Communication' apps and switch to make phone calls and sending SMS messages. **Third**, with the development of the app market, a stable and highly correlated app ecosystem has been formed (detailed in Section 3.4). Various app categories are connected with and reliant on others. Due to the high correlations among app categories, users have to keep using multiple app categories together. For example, for online shoppers, apart from 'Shopping' apps, they have to use 'Finance' apps for online payment as well.

## 3.4 Correlations of App Categories

To validate the previous inference about the app ecosystem, we then study the correlations of app categories. In our case, we use the co-usage of app categories for unique users to represent their correlations. Given two app categories $C_A$ and $C_B$, we denote the number of unique users using an app either in category $C_A$ or $C_B$ as $\mathscr{D}(C_A \bigcup C_B)$, and the number of unique users using apps from both categories $C_A$ and $C_B$ as $\mathscr{D}(C_A \bigcap C_B)$. The correlation between categories $C_A$ and $C_B$ is computed as,

$$Corr(C_A, C_B) = \frac{\mathscr{D}(C_A \bigcap C_B)}{\mathscr{D}(C_A \bigcup C_B)}. \quad (2)$$

The correlation $Corr(C_A, C_B)$ represents the probability that one user uses both categories $C_A$ and $C_B$.

Figure 7 displays the correlations of app categories in 2012, 2014, and 2017, respectively. In the heatmap, each row or column represents one app category. The categories are sorted in descending order by their first letter (the same as Figure 4). From Figure 7, we can observe that the strength of correlations between app categories generally increased from 2012 to 2014. Comparing the heatmaps in 2014 and 2017, *the correlations across various app categories were high and tended to be stable, suggesting that a robust app ecosystem had formed*. In that app ecosystem, all app categories are closely related to each other. 'Communication' and 'Social' have the highest correlations with other app categories. As the most popular app categories, 'Communication' and 'Social' act as the bases of the app ecosystem and the bridge to connect different categories. For example, users may recommend useful apps or share interesting content like news, music, and videos via 'Communication' and 'Social' apps to others. Meanwhile, others may try the recommended apps or use a viewer app to open the received content. Therefore, the fragmented and independent app categories are closely interconnected and form a robust ecosystem.

## 3.5 Summary

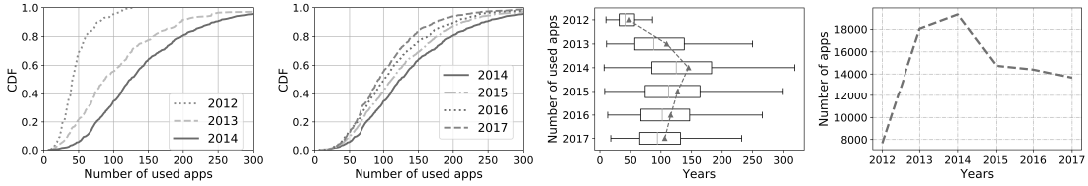
From the macro-level analysis, we can conclude that mobile app-category usage has indeed changed over the six years from 2012 to 2017. The functionality of smartphones has broadened from fundamental communication needs to life assistants. Generally, the evolution of app-category usage has two stages, i.e., a growth stage and plateau stage. The growth stage is triggered by the release of new technologies in multiple fields, including mobile networks, smartphone hardware, and software. The plateau stage is caused by both user factors, including user habits and user communities, and app factors, including the high correlated app ecosystem. Due to the stable evolution processes, users' app-category usage tends to be consistent, i.e., the diversity decreases, over time.

# 4 EVOLUTION OF APP USAGE

## 4.1 Number of Used Apps

We first analyze the number of apps used by each unique user. As shown in Figures. 8(a) and 8(b), similar to app categories, the evolution of app usage is also separated into two stages by the year 2014.

- **Stage one (2012 - 2014).** During this stage, *users increased the number of apps used on their smartphones*. In 2012, a user used up to 120 apps in one year, which is consistent with the




(a) CDF of the number of used apps for each user from 2012 to 2014.
(b) CDF of the number of used apps for each user from 2014 to 2017.
(c) Distribution of the number of used apps for each user across different years.
(d) The total number of used apps across different years.

**Figure 8: The evolution of app usage from 2012 to 2017.**

finding in previous work by Falaki *et al.* [7]. Nevertheless, in 2013, over 20% of users used at least 150 apps. In 2014, that proportion rose significantly to around 40%. This boosting period at the micro-level is consistent with the macro-level. As analyzed before, the occurrence of this stage should be motivated by the release of new technologies.

- **Stage two (2014 - 2017).** During this stage, *the number of apps used by each user decreased year by year, which is significantly different from the trend at the macro-level.* In 2017, the proportion of users who used over 150 apps fell to 20%.

To examine the changes in detail, we depict the distribution across different years using box-plots in Figure 8(c). We observe that the minimum number of used apps almost did not change over the six years and always stayed at around 12. The 12 app limit suggests that one smartphone has at least 12 essential functions. We will determine these 12 essential apps in Section 4.3. Moreover, from 2014 to 2017, compared with the minimum value and first quartile, the third quartile and maximum value dropped more sharply. That means the people who use many apps were significantly influenced and tended to use fewer apps. We then compute the total number of used apps across different years and present the results in Figure 8(d). However as opposed to Figures 8(a), 8(b), and 8(c), we aggregate all apps used in that year by all long-term users. As shown in Figure 8(d), the total number of apps used per year peaked in 2014 and then gradually declined. The decreasing trend implies that low-quality apps started to be discarded by users after the boosting period, i.e., stage one.

Because of the difference in trends during stage two at the macro-level and the micro-level, we next study the relationship between the numbers of apps and app categories used by each user. We show the data in Figure 9, where each dot represents one unique user. Generally, people who use more apps also use more app categories. From 2013 to 2014, the data points moved to the right and down, indicating that users started to use more apps and app categories simultaneously. Interestingly, we discover a phase change in Figure 9(b). When the user used more than 15 app categories, the number of used apps would increase dramatically. The different degrees of maturity across app categories may cause this phase change. In 2014, there were around 15 developed app categories with high degrees of maturity, and their markets were dominated by three to five apps in each category. As a result, users would focus on a small group of governing apps when they used these app categories. Conversely, when users utilized developing app categories lacking the governing apps, they would try many of the apps in that category and then select several high-quality apps. Thus, the number of used apps would increase suddenly. Compared with 2014, the data points in 2015 and 2017 moved to the left, suggesting users used fewer apps, but the distribution of the number of used app categories did not change. Moreover, in Figure 9(d), we discover the phase change faded, implying that governing apps have appeared in the previous developing app categories.

## 4.2 Diversity of App Usage

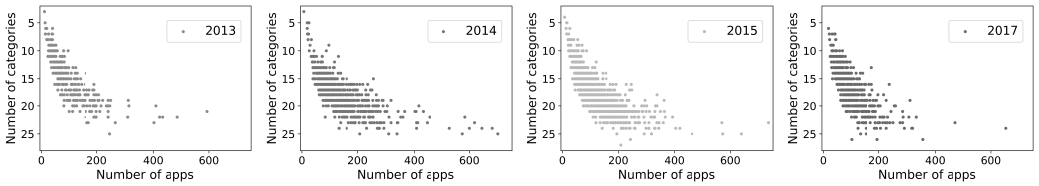
We next explore a question: how the diversity of app usage changes over time and whether the trend is consistent with app categories. By applying Jaccard distance to measure the difference of app usage between every two users, we depict the distribution of pairwise Jaccard distances across different years in Figure 10. From 2012 to 2013, the average distance between two users jumped from 0.75 to 0.85, implying the diversity of app usage increased. The trend is contrary to that at the macro-level in Figure 3. After 2013, the distribution became stable, i.e., the strength of diversity stopped increasing. However, users' used apps were still extremely different from others considering the minimum distance is nearly 0.7.

In summary, *the diversity between users exhibits two opposite evolutionary trends at the micro-level, i.e., apps, and the macro-level, i.e., app categories, respectively.* At the macro-level, mobile users fully explore the functionality of smartphones and tend to use more and similar app categories. On the other hand, at the micro-level, mobile users have different preferences and use a diverse array of apps.

## 4.3 Distribution of App Popularity

We further study the distributions of app popularity from 2012 to 2017. Figure 11 reports the CDF of app popularity (the ratio of app users to all users). Our results reveal a typical *Pareto effect* for app usage. Over 80% of apps have less than 0.01 popularity in 2012, while this number increased to 90% by 2017. The Pareto effect suggests that although the set of apps used by one user are quite different from others, the app market is still governed by a small number of dominating apps. This observation is consistent across all six years.

We next rank apps in terms of their popularity and select the top 20 apps for each year. We then discover that there are 16 dominating apps that repeatedly appeared in the top 20 list every year. We post the 16 apps and their rankings across different years in Figure




(a) The number of used app categories and apps across different users in 2013.

(b) The number of used app categories and apps across different users in 2014.

(c) The number of used app categories and apps across different users in 2015.

(d) The number of used app categories and apps across different users in 2017.

**Figure 9: The relationship between the number of used app categories and apps in different years.**

**Table 2: Twelve essential apps and their functionality.**

| App | Functionality | App | Functionality | App | Functionality |
|---|---|---|---|---|---|
| com.sec.android.inputmethod | Keyboard input | com.sec.android.gallery3d | Image and video viewing | com.sec.android.app.launcher | Home screen application |
| com.google.android.apps.plus | Google+, socializing | com.google.android.talk | Message contacts, video or voice calls | com.google.android.music | Music palyer |
| com.google.android.apps.maps | Maps and navigation | com.google.android.gms | Google play services | com.google.android.gm | Gmail |
| com.google.android.youtube | Watching videos | com.google.android.googlequicksearchbox | Google search | com.android.chrome | Web browsing |

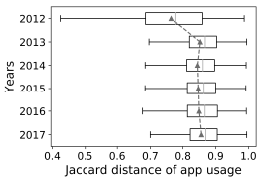



**Figure 10: Jaccard distance of app usage.**

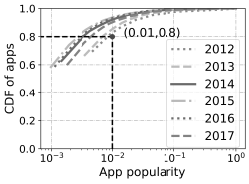



**Figure 11: CDF of the popularity of apps.**

12. Twelve out of the 16 dominating apps are part of the Android operating system, and correspond to the 12 essential apps observed in Figure 8(c). We list the 12 apps and their functionality in Table 2. Apart from these 12 essential apps, there are also four dominating apps from three prior prevalent app categories. Whatsapp and Push service are from the 'Communication' category. Facebook is from the 'Social' category, and Dropbox is from the 'Tools' category. The rankings of dominating apps did not vary significantly during the period. Google quick search box and Google Play services had the most number of users. Also, the popularity of Chrome and Whatsapp rose steadily every year.

## 4.4 App Usage Within App Categories

Up to now, we have discovered that the evolutionary processes at the macro-level and the micro-level show considerable differences, especially during stage two, i.e., from 2014 to 2017. Therefore, we next delve into the reasons behind this phenomenon and investigate how app usage changes in a particular app category. For the sake

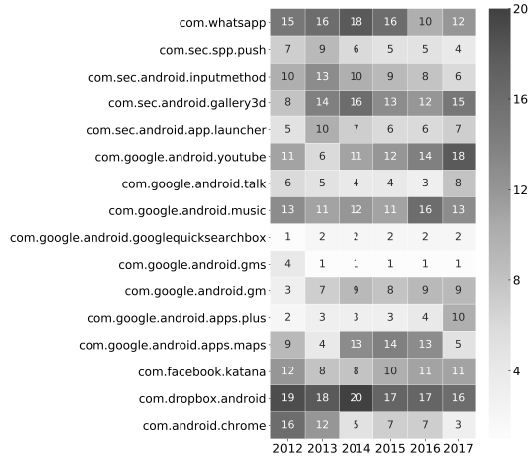
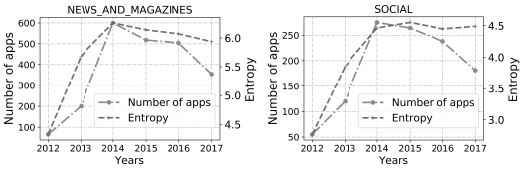



**Figure 12: Rank of popular apps across different years.**

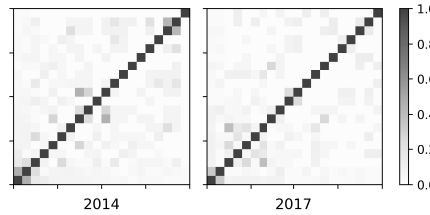
of representativeness, we actually select two typical app categories, i.e., 'News and magazine' representing a posterior prevalent app category and 'Social' representing a prior prevalent app category.

In our case, we apply the number of apps and app usage entropy to measure the evolution processes. Figure 13 shows the results. The entropy is a common metric to measure the randomness of a system [9]. We use entropy to measure the centralization of app usage in one specific app category, i.e., whether app usage in that

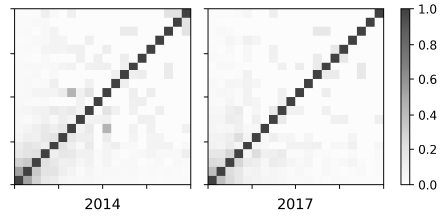



(a) 'News and magazine' category. (b) 'Social' category.

**Figure 13: Evolution of app usage in 'News and magazine' and 'Social' categories.**




(a) Correlations of apps in 'News and magazines' category.




(b) Correlations of apps in 'Social' category.

**Figure 14: Correlations of apps in 'News and magazine' and 'Social' categories.**

category concentrates on a few apps. The lower the entropy, the higher the centralization of app usage.

In terms of Figure 13, for both 'News and magazine' and 'Social' categories, the number of apps peaked in 2014 and then decreased. Their trends correspond to the trend for all apps, as shown in Figure 8(d). Additionally, we have also examined the other app categories and found their trends are consistent as well. Consequently, different degrees of maturity will not affect the evolution of the number of apps in different app categories. In terms of the number of intra-category apps, all app categories underwent two evolution stages, i.e., growth stage and elimination stage. We infer that the growth stage is caused by the release of new technologies, while the weeding-out of low-quality apps by users causes the elimination stage.

However, the evolution in entropy exhibits different trends in 'News and magazine' and 'Social' categories. For the 'Social' category, entropy first increased and then kept steady. The increase stage is caused by the growing number of apps in the category. New apps disperse users' concentration. On the other hand, the Pareto effect leads to the plateau stage. As a prior prevalent app category, 'Social' had a few governing apps dominating usage before 2012. Therefore, during the boosting period, the newly introduced apps would compete with these old governing apps, and some low-quality would be eliminated. Meanwhile, new governing apps would emerge. As a result, in 2014, apart from the increasing entropy, users' usage was also hugely dominated by both previous and new governing apps. Therefore, after 2014, the entropy did not change dramatically. For the 'News and magazine' category, the evolution in entropy still experienced the decrease stage. Since 'News and magazine' is a posterior prevalent app category, limited by its maturity, it had few governing apps before 2012. Hence, its entropy is deeply affected by the number of apps in the category.

In order to better understand the app elimination stage, we next investigate how the correlations of apps in the same app category changed from 2014 to 2017. Similar to Section 3.4, we use the co-usage of apps for unique users to represent their correlations. For consistency, we still use 'News and magazine' and 'Social' to represent posterior and prior prevalent app categories, respectively. In Figure 14, we depict the correlations of the top 20 popular apps in these two categories. In the heatmap, each row or column represents one app. The apps are listed in descending order in terms of their popularity. Compared with app categories, the correlations of apps in the same category is much lower, and most are below 0.2. Since the functionality of apps in the same category is similar, installing multiple apps from the same category is often redundant. Also, due to intra-category competition, the correlations of apps shrank over time in both categories. We still observe that in the 'News and magazine' category, apps' correlations nearly followed a uniform distribution in 2014, suggesting that at the beginning stage, the correlations of apps are independent of their popularity. In 2017, with the increase in the degree of the app category's maturity, the apps with high correlations tended to have high popularity. By comparing the top 20 popular apps in both 'News and magazine' and 'Social' categories from 2014 to 2017, we then discover the relationship between correlations and popularity of apps. The apps with high correlations have a greater chance of gaining popularity in the future.

## 4.5 Summary

In terms of the micro-level observations, users' mobile app usage exhibits different evolution processes from the macro-level. The fierce intra-category competition leads to the occurrence of an elimination stage and a decrease in the correlations of apps. Moreover, because of the high overlapping functionality across apps and their often perfect substitutability, mobile users have less reliance on an individual app. Therefore, users' app usage diversity is vast. Nevertheless, due to the Pareto effect, the most popular apps across users are still consistent. Also, in terms of the entropy metric, the degree of app category maturity will affect the evolution of app usage in the category.

## 5 RELATED WORK AND DISCUSSIONS

### 5.1 Related Work

*5.1.1 App Usage Analysis.* Many previous studies have focused on how individuals use their smartphones and mobile apps [11, 12, 18, 24, 32]. Generally, they discovered people's app usage patterns by clustering users into groups and providing comprehensive descriptions of those groups. Zhao *et al.* [32] analyzed a short-term app usage dataset of one month covering 106,762 users. They grouped users into 382 clusters and gave a meaningful label to each cluster, such as Night communicators, Evening learners, and Financial users. In [12], Katevas *et al.* collected a four-week usage dataset from 340 users and revealed five user profiles, including limited use, business use, power use, and personality use. Jones *et al.* and Cao *et al.* [3, 11] analyzed users' app re-visitation patterns based on a three-month dataset covering 165 users and identified three distinct user clusters, i.e., checkers, waiters, and responsives. In [18], Peltonen *et al.* collected an one-year app usage dataset from 25,323 users distributed in 44 countries. They clustered users based on their cultural features and investigated how their cultural affiliations affect their usage behavior. However, existing studies only concentrated on a limited time span ranging from one week to one year, and did not investigate the long-term evolution of mobile app usage.

*5.1.2 App Evolution Analysis.* Also, some scholars worked on analyzing app evolution [2, 4, 22, 25]. Carbunar *et al.* [4] crawled an app dataset from Google Play including 160,000 apps over six months. They studied the evolution of app properties, like downloads, price, and update frequency. In [2], Calciati *et al.* studied how apps evolve in their permission requests. They tracked over 14,000 releases of 227 Android apps and discovered a common trend of apps requiring more permissions over time. Alternatively, in [22], Taylor *et al.* also took quarterly snapshots of Google Play over two years and investigated how permissions requested by apps changed over time. They analyzed over 30,000 apps and discovered that Android apps are not getting safer as they are updated. Wang *et al.* [25] crawled three snapshots of Google Play in 2014, 2015, and 2017, and explored the evolution of various app properties, including permission usage, privacy policy declaration, advertising libraries, updates, and malicious behavior. However, these studies only consider the evolution of apps' inherent properties instead of users' actual usage. Due to the lack of user involvement, it is hard for them to capture the real trends of the app market and the preferences of users. In contrast, our work is the first attempt to understand the evolution of users' mobile app usage through data collected on smartphones over the years.

### 5.2 Discussions

The most prominent discovery in our paper is that the usage evolution at different levels exhibits different processes. The relevant stakeholders should note this difference because they play different roles at different levels of the app market. For example, Google is responsible for maintaining the Android operating system. Market intermediaries are in charge of managing app platforms, while app developers should provide high-quality apps. The relevant stakeholders should focus on the evolution of their corresponding level and dynamically adjust their strategies to improve their services. Also, we discover that the release of new technologies will trigger increasing usage in both app categories and individual apps. Hence, when a breakthrough occurs, all relevant stakeholders can grasp the valuable opportunity to extend their market shares. One potential opportunity is the deployment of 5G mobile networks.

We also discovered opposing trends in usage diversity between app categories and apps. App category usage tends to be consistent, while app usage across mobile users becomes quite different. Therefore, the app market intermediaries, at a higher level, should focus on the consistent requirements across mobile users instead of customized services. However, as for app developers, seeking to develop a commonly popular app for all mobile users may be difficult. Instead, focusing on small groups of users and meeting their personalized needs is a better choice.

We also notice the fierce intra-category competition between apps causes an elimination stage of app usage, and different degrees of app category maturity will affect this competition. Hence, app developers have to improve the competitiveness of their apps, especially during the elimination stage. Also, when they design new apps, they can take the maturity of app categories into account and choose a newly introduced or developing app category. Additionally, we notice that correlation plays a vital role in app usage. The apps with high correlations with others will become more popular in the future. Hence, app developers can leverage this finding to improve their apps' competitiveness by adding both inter- and intra-app category cooperation functions into their app designs.

App usage shows a typical *Pareto effect* at all times. A small group of apps dominate usage in both the entire app market and individual app categories. We also identify twelve essential apps of differing functionality for smartphones. In terms of these observations, the market intermediaries can recognize a small group of popular apps and put their installation packages as close as possible to end-users. For example, with the help of network service providers, they can cache the .apk files at the edges of networks.

## 6 CONCLUSION

We conduct the first comprehensive study of the long-term evolution of mobile app usage. Specifically, our analysis covers about 1,500 Android users with six-year app usage records from 2012 to 2017. Overall, our findings indicate that users' app usage indeed changes over time. However, the evolution processes in app-category usage and individual app usage are different in terms of popularity distribution, usage diversity, and correlations. Our findings provide insights for app developers to make better decisions on developing apps and improve competitiveness. Also, our study can help market intermediaries to manage app platforms and supply high-quality services.

## ACKNOWLEDGMENTS

This research has been supported in part by project 16214817 from the Research Grants Council of Hong Kong, project FP805 from HKUST, the 5GEAR project, the FIT project and the CBAI (Crowdsourced Battery Optimization AI for a Connected World, grant No. 1319017) project from the Academy of Finland, the National

Key Research and Development Program of China under grant 2018YFB1800804, the National Nature Science Foundation of China under U1836219, 61971267, 61972223, 61861136003, Beijing Natural Science Foundation under L182038, Beijing National Research Center for Information Science and Technology under 20031887521, and research fund of Tsinghua University-Tencent Joint Laboratory for Internet Innovation Technology.

## REFERENCES


[1] Matthias Böhmer, Christian Lander, and Antonio Krüger. 2013. What's in the Apps for Context? Extending a Sensor for Studying App Usage to Informing Context-Awareness. In *Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication (UbiComp '13 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 1423–1426. https://doi.org/10.1145/2494091.2496038

[2] Paolo Calciati and Alessandra Gorla. 2017. How Do Apps Evolve in Their Permission Requests? A Preliminary Study. In *Proceedings of the 14th International Conference on Mining Software Repositories (MSR '17)*. IEEE Press, 37–41. https://doi.org/10.1109/MSR.2017.64

[3] Hancheng Cao, Zhilong Chen, Fengli Xu, Yong Li, and Vassilis Kostakos. 2018. Revisitation in Urban Space vs. Online: A Comparison across POIs, Websites, and Smartphone Apps. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 4, Article Article 156 (Dec. 2018), 24 pages. https://doi.org/10.1145/3287034

[4] Bogdan Carbunar and Rahul Potharaju. 2015. A Longitudinal Study of the Google App Market. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015 (ASONAM '15)*. Association for Computing Machinery, New York, NY, USA, 242–249. https://doi.org/10.1145/2808797.2808823

[5] Xinlei Chen, Yu Wang, Jiayou He, Shijia Pan, Yong Li, and Pei Zhang. 2019. CAP: Context-Aware App Usage Prediction with Heterogeneous Graph Embedding. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 1, Article Article 4 (March 2019), 25 pages. https://doi.org/10.1145/3314391

[6] Trinh-Minh-Tri Do and Daniel Gatica-Perez. 2010. By Their Apps You Shall Understand Them: Mining Large-Scale Patterns of Mobile Phone Usage. In *Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia (MUM '10)*. Association for Computing Machinery, New York, NY, USA, Article Article 27, 10 pages. https://doi.org/10.1145/1899475.1899502

[7] Hossein Falaki, Ratul Mahajan, Srikanth Kandula, Dimitrios Lymberopoulos, Ramesh Govindan, and Deborah Estrin. 2010. Diversity in Smartphone Usage. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services (MobiSys '10)*. Association for Computing Machinery, New York, NY, USA, 179–194. https://doi.org/10.1145/1814433.1814453

[8] Eduardo Graells-Garrido, Diego Caro, Omar Miranda, Rossano Schifanella, and Oscar F. Peredo. 2018. The WWW (and an H) of Mobile Application Usage in the City: The What, Where, When, and How. In *Companion Proceedings of the The Web Conference 2018 (WWW '18)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 1221–1229. https://doi.org/10.1145/3184558.3191561

[9] Robert M Gray. 2011. *Entropy and Information Theory*. Springer Science & Business Media.

[10] Chakajkla Jesdabodi and Walid Maalej. 2015. Understanding Usage States on Mobile Devices. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '15)*. Association for Computing Machinery, New York, NY, USA, 1221–1225. https://doi.org/10.1145/2750858.2805837

[11] Simon L. Jones, Denzil Ferreira, Simo Hosio, Jorge Goncalves, and Vassilis Kostakos. 2015. Revisitation Analysis of Smartphone App Use. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '15)*. Association for Computing Machinery, New York, NY, USA, 1197–1208. https://doi.org/10.1145/2750858.2807542

[12] Kleomenis Katevas, Ioannis Arapakis, and Martin Pielot. 2018. Typical Phone Use Habits: Intense Use Does Not Predict Negative Well-Being. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '18)*. Association for Computing Machinery, New York, NY, USA, Article Article 11, 13 pages. https://doi.org/10.1145/3229434.3229441

[13] Sven Kosub. 2019. A note on the triangle inequality for the Jaccard distance. *Pattern Recognition Letters* 120 (2019), 36–38. https://doi.org/10.1016/j.patrec.2018.12.007

[14] Huoran Li, Xuan Lu, Xuanzhe Liu, Tao Xie, Kaigui Bian, Felix Xiaozhu Lin, Qiaozhu Mei, and Feng Feng. 2015. Characterizing Smartphone Usage Patterns from Millions of Android Users. In *Proceedings of the 2015 Internet Measurement Conference (IMC '15)*. Association for Computing Machinery, New York, NY, USA, 459–472. https://doi.org/10.1145/2815675.2815686

[15] Zhung-Xun Liao, Yi-Chin Pan, Wen-Chih Peng, and Po-Ruey Lei. 2013. On Mining Mobile Apps Usage Behavior for Predicting Apps Usage in Smartphones. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management (CIKM '13)*. Association for Computing Machinery, New York, NY, USA, 609–618. https://doi.org/10.1145/2505515.2505529

[16] Nigel Linge and Andy Sutton. 2014. The road to 4G. *The Journal of the Institute of Telecommunications Professionals* 8, 1 (2014).

[17] Robert McGill, John W. Tukey, and Wayne A. Larsen. 1978. Variations of Box Plots. *The American Statistician* 32, 1 (1978), 12–16. https://doi.org/10.1080/00031305.1978.10479236

[18] Ella Peltonen, Eemil Lagerspetz, Jonatan Hamberg, Abhinav Mehrotra, Mirco Musolesi, Petteri Nurmi, and Sasu Tarkoma. 2018. The Hidden Image of Mobile Apps: Geographic, Demographic, and Cultural Factors in Mobile Usage. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '18)*. Association for Computing Machinery, New York, NY, USA, Article Article 10, 12 pages. https://doi.org/10.1145/3229434.3229474

[19] Vladan Radosavljevic, Mihajlo Grbovic, Nemanja Djuric, Narayan Bhamidipati, Daneo Zhang, Jack Wang, Jiankai Dang, Haiying Huang, Ananth Nagarajan, and Peiji Chen. 2016. Smartphone App Categorization for Interest Targeting in Advertising Marketplace. In *Proceedings of the 25th International Conference Companion on World Wide Web (WWW '16 Companion)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 93–94. https://doi.org/10.1145/2872518.2889411

[20] Vijay Srinivasan, Saeed Moghaddam, Abhishek Mukherji, Kiran K. Rachuri, Chenren Xu, and Emmanuel Munguia Tapia. 2014. MobileMiner: Mining Your Frequent Patterns on Your Phone. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '14)*. Association for Computing Machinery, New York, NY, USA, 389–400. https://doi.org/10.1145/2632048.2632052

[21] Statista. 2019. https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/. (2019).

[22] Vincent F. Taylor and Ivan Martinovic. 2017. To Update or Not to Update: Insights From a Two-Year Study of Android App Evolution. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security (ASIA CCS '17)*. Association for Computing Machinery, New York, NY, USA, 45–57. https://doi.org/10.1145/3052973.3052990

[23] Zhen Tu, Runtong Li, Yong Li, Gang Wang, Di Wu, Pan Hui, Li Su, and Depeng Jin. 2018. Your Apps Give You Away: Distinguishing Mobile Users by Their App Usage Fingerprints. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 3, Article Article 138 (Sept. 2018), 23 pages. https://doi.org/10.1145/3264948

[24] Steven Van Canneyt, Marc Bron, Andy Haines, and Mounia Lalmas. 2017. Describing Patterns and Disruptions in Large Scale Mobile App Usage Data. In *Proceedings of the 26th International Conference on World Wide Web Companion (WWW '17 Companion)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 1579–1584. https://doi.org/10.1145/3041021.3051113

[25] Haoyu Wang, Hao Li, and Yao Guo. 2019. Understanding the Evolution of Mobile App Ecosystems: A Longitudinal Measurement Study of Google Play. In *The World Wide Web Conference (WWW '19)*. Association for Computing Machinery, New York, NY, USA, 1988–1999. https://doi.org/10.1145/3308558.3313611

[26] Huandong Wang, Yong Li, Sihan Zeng, Gang Wang, Pengyu Zhang, Pan Hui, and Depeng Jin. 2019. Modeling Spatio-Temporal App Usage for a Large User Population. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 1, Article Article 27 (March 2019), 23 pages. https://doi.org/10.1145/3314414

[27] Pascal Welke, Ionut Andone, Konrad Blaszkiewicz, and Alexander Markowetz. 2016. Differentiating Smartphone Users by App Usage. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '16)*. Association for Computing Machinery, New York, NY, USA, 519–523. https://doi.org/10.1145/2971648.2971707

[28] Wikipedia. 2019. https://en.wikipedia.org/wiki/Android_(operating_system). (2019).

[29] Wikipedia. 2019. https://en.wikipedia.org/wiki/HTC_Dream. (2019).

[30] Wikipedia. 2019. https://en.wikipedia.org/wiki/Mobile_app. (2019).

[31] Qiang Xu, Jeffrey Erman, Alexandre Gerber, Zhuoqing Mao, Jeffrey Pang, and Shobha Venkataraman. 2011. Identifying Diverse Usage Behaviors of Smartphone Apps. In *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference (IMC '11)*. Association for Computing Machinery, New York, NY, USA, 329–344. https://doi.org/10.1145/2068816.2068847

[32] Sha Zhao, Julian Ramos, Jianrong Tao, Ziwen Jiang, Shijian Li, Zhaohui Wu, Gang Pan, and Anind K. Dey. 2016. Discovering Different Kinds of Smartphone Users through Their Application Usage Behaviors. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '16)*. Association for Computing Machinery, New York, NY, USA, 498–509. https://doi.org/10.1145/2971648.2971696

# Paper IV

Tong Li, Zhaoqi Yang, Yong Li, Benjamin Finley, Sasu Tarkoma, and Pan Hui

**Revealing Urban Dynamic Functions with Mobile App Usage Behavior and POIs**

Submitted to IEEE Transactions on Mobile Computing.

# Revealing Urban Dynamic Functions with Mobile App Usage Behavior and POIs

Tong Li, *Student Member, IEEE,* Zhaoqi Yang, Yong Li, *Senior Member, IEEE,* Benjamin Finley, Sasu Tarkoma, *Senior Member, IEEE,* and Pan Hui, *Fellow, IEEE*

**Abstract**—A city is composed of many regions providing different functions for urban residents (for example residential regions and business regions). Additionally, due to daily urban dynamics, a region might provide different functions at different times of the day. In this work, we propose a graph-based representation learning framework that reveals urban dynamic functions using mobile app usage behavior and POIs. Specifically, we use a graph structure to model POIs and mobile app usage data jointly. In this graph, nodes represent users, apps, and time-enhanced locations, and edges represent the co-occurrence of entities in app usage records. POI distributions of regions are treated as location node features. Through leveraging meta-paths and graph neural networks, the proposed framework is able to map time-enhanced location nodes into the same latent space, which captures both graph structure (i.e., mobile app usage) and node feature (i.e., POIs) information. As a result, a region at a specific time interval is represented by an embedding vector. We further evaluate our framework through a series of experiments conducted on real-world datasets. Specifically, we use the learned region embeddings for the two distinct tasks of static land usage identification and regional economic level (GDP) prediction. Our method outperforms the state-of-the-art approaches by over 20%. Moreover, we present three case studies to illustrate how region functions change throughout the day by using learned dynamic region embeddings. Overall, the work not only lights the way for further urban-related applications, but also shows the significant potential of mobile app usage data in urban analytics.

**Index Terms**—Functional regions, dynamic functions, app usage, representation learning.

## 1 Introduction

Urbanization often leads to different regions of a city having different functional roles to support urban residents' diverse demands [1], such as working, residence, studying, and entertainment. These functional regions can be naturally formed according to the lifestyles of residents. Alternatively, the government or urban planners can artificially design them. Studying urban functions provides essential information useful in urban planning and management to help solve many urban challenges [2]. Therefore, such studies play a critical role in urban analysis.

Until very recently, most of our understanding of urban functions focuses on static land usage. However, in practice, a region often has multiple functions to meet the various needs of local residents. For example, shopping malls and residences are often co-located in a single region. Moreover, due to diurnal urban dynamics and the multiple functions, regions can exhibit different functions at different times of the day [3]. Thus, static land usage analysis often fails to cope with such complex and dynamic cases, and exploring the changes in regional functions during the day is essential. Studying dynamic functions enable us to gain insight beyond static analysis to reveal regional characteristics from multiple perspectives throughout the day.

In this work, we aim to reveal urban dynamic functions by using spatiotemporal app usage data and points of interests (POIs). Specifically, POIs implicitly reflect a regions' static functions, while spatiotemporal app usage data depict the dynamic app usage behavior across different regions. In particular, mobile app usage data has three advantages for understanding urban dynamic functions. **First**, many existing studies have shown that mobile app usage behavior in a region has a strong link with a region's features. For instance, mobile users prefer to use browser and multimedia apps in the airport and transit stations while waiting [4]. **Second**, mobile app usage behavior, as a kind of spatiotemporal data, can depict the dynamic relations across different regions and further help uncover dynamic functions. **Third**, compared with other data sources, like taxi trips, mobile app usage records can cover almost the entire urban area, instead of just certain transit hot spots.

Revealing urban dynamic functions with mobile app usage behavior and POIs, however, is non-trivial due to three **challenges**. **1)** As POIs and app usage behavior are cross-domain data, jointly leveraging and combining these datasets in a unified way is the first challenge. **2)** Mobile app usage data only explicitly includes relationships between entities of different types (e.g., region to app, app to user, etc.) but not between entities of the same type (e.g., region to region). However, as we want to uncover regional characteristics, we need to extract these hidden dynamic region to region relationships from the app usage behavior.


- *T. Li and P. Hui are with the System and Media Laboratory (SyMLab), Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong. They are also with the Department of Computer Science, University of Helsinki, Helsinki, Finland. E-mail: t.li@connect.ust.hk, panhui@cse.ust.hk*
- *Z. Yang and Y. Li are with the Beijing National Research Center for Information Science and Technology (BNRist), Department of Electronic Engineering, Tsinghua University, Beijing, China. E-mail: liyong07@tsinghua.edu.cn*
- *S. Tarkoma and B. Finley are with the Department of Computer Science, University of Helsinki, Helsinki, Finland. E-mail: sasu.tarkoma@cs.helsinki.fi*

This task is fundamental but difficult. **3**) In addition to qualitative analysis, we are also interested in quantitatively measuring the dynamic functions of regions. Therefore, devising a method for quantitative analysis and for depicting the intensities of different functions throughout the day is another challenge.

To solve the mentioned challenges, we propose a graph-based representation learning framework that reveals dynamic functions by leveraging three key designs. **First**, we use a graph structure to combine cross-domain data, including POIs and mobile app usage. Specifically, by introducing user, app, and time-enhanced location nodes, we build a heterogeneous graph representing dynamic app usage behavior. Each time-enhanced location node represents one region at a specific time interval. Also, the POI distribution of regions become node features. **Second**, to extract hidden inter-regional relations from app usage data, we designed a meta-path guided method to generate a relational location graph from the heterogeneous app usage graph. The relational location graph only contains time-enhanced location nodes, where edges represent a composite relation connecting the two regions. **Third**, to conduct quantitative analysis, we design a graph auto-encoder that leverages relational graph attention networks to learn the dynamic embeddings of regions, i.e., to map time-enhanced location nodes into the same latent space. These embeddings measure the relationship strengths between regions of different time slots and determine the intensities of different functions of regions.

In summary, the main contributions of this work can be summarized as follows.

- We investigated the problem of revealing urban dynamic functions by jointly using mobile app usage data and POIs. To the best of our knowledge, this is the first study to illustrate how region function changes throughout the day.
- We develop a graph-based framework to learn the dynamic embeddings of regions. Specifically, we model spatiotemporal app usage behavior by constructing a heterogeneous graph with the user, app, time-enhanced location nodes, and POI distributions of regions as location node features. By utilizing meta-paths and graph neural networks, the framework can sufficiently integrate graph structure (i.e., mobile app usage) and node features (i.e., POIs) information into region embeddings.
- We evaluate our proposed framework through a set of experiments conducted on real-world datasets. We first use region embeddings to identify static land usage. Our method outperforms state-of-the-art baselines by over 20%. Next, by using the learned dynamic region embeddings, we present three case studies to reveal how region functions change throughout the day. Eventually, we employ dynamic embeddings to predict district economic levels and achieve an accuracy of 84%, which illustrates the strong correlations between dynamic functions and the economic development of districts. Through a series of experiments, we show the superiority and the effectiveness of our framework.

## 2 PRELIMINARY

In this section, we present a set of important preliminaries for understanding our research problem and method.

### 2.1 Data Overview

#### 2.1.1 Mobile App Usage

Mobile app usage data refers to a set of cyber activity records generated by smartphone users using mobile apps. As ubiquitous data, mobile app usage records are generally collected using monitoring apps [5] or from network operators [6]. Such data has been essential in many ubiquitous computing problems like app ecosystem modeling [7], user profiling [8], and urban dynamics analysis [9].

Specifically, when a user launches or switches to a specific app, the app will move to the smartphone foreground; this behavior is regarded as a use of that app and generates an app usage record. In general, an app usage record includes 4W features, i.e., who, what, where, and when. In other words, a raw app usage record can be represented as 4-tuple, i.e., $r =< u, a, l, t >$, where $u$ represents the user (who), $a$ represents the app used (what), $l$ represents the location information (where), and $t$ represents the timestamp of that record (when). In terms of location information, monitoring apps can collect the location from the smartphone GPS sensor [10]. Alternatively, network operators can infer the location information from the locations of the based stations associated with the user [11].

In this work, we leverage a city-scale mobile app usage dataset provided by a primary Internet Service Provider (ISP) in China. The dataset was collected over one week in April 2016 and covers the whole metropolitan area of Shanghai, one of the world's largest cities. Each app usage record of the dataset contains an anonymized user ID, app ID, base station ID, and timestamp. The full dataset includes 1.7 million users, and their app usage records during the data collection period. Specifically, we utilize a subset containing the top 100000 users ranked by their total number of usage records. We also note that the data subset scale could be easily achieved using monitoring apps [5], [12] or collecting from network operators [6]. Table 1 presents a descriptive summary of the used dataset. As for location information, we use the GPS-based location of the associated base station in each record as an approximation. Fig. 1 depicts the distribution of these 9858 base stations in Shanghai. We note that such a dense deployment of base stations helps add confidence that the location approximation is fairly accurate thus bolstering our analysis.

#### 2.1.2 Road Network

The road network data consist of a set of road segments that naturally partition a city into multiple polygons that are thus regarded as regions. Compared to using uniform grid-based partition, a region bounded by major roads may be simply a block or could be a community with more semantic meanings, like residential, park, office, etc. [13].

In practice, we extracted the major road networks of Shanghai from OpenStreetMap[1], which provides free crowd-sourced geographic data. Similar to [1], by using major roads like highways and ring roads, we partitioned the whole metropolitan area of Shanghai into 1,595 disjoint regions. The skeleton of the road network is shown in Fig. 2. In our case, we use regions as the atom unit.

1. https://www.openstreetmap.org/

TABLE 1
Mobile app usage dataset descriptive summary.

| # of Users | # of Records | # of Identified apps | # of Base stations | Duration | Area |
|---|---|---|---|---|---|
| 100,000 | 270,802,027 | 2,000 | 9,858 | One week | Shanghai |

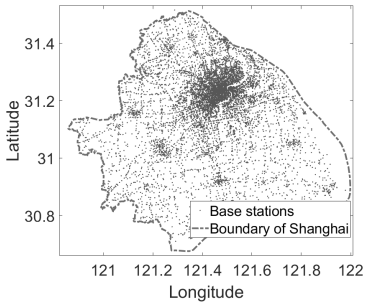



Fig. 1. Distribution of base stations of ISP in Shanghai.

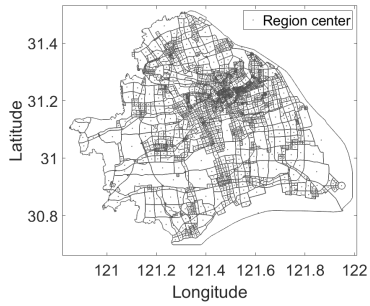



Fig. 2. Skeleton of road networks in Shanghai.

#### 2.1.3 Point of Interest

Point of interest (POI) data consists of a set of POIs, which represent various venues in the physical world, like shopping malls, restaurants, and theatres. POI data implicitly reflects the static functions of regions. For example, if a region contains many shopping malls, that region could be considered a commercial area. Generally, a POI has a name, category, location (coordinates), and other attributes. We crawled 782,528 POIs of Shanghai from the Baidu Map service [14] to create a POI dataset. There are 15 POI categories, including restaurant, hotel, entertainment, industry, residence, education, hospital, fitness center, shopping mall, scenic spot, transportation facility, financial service, life service, corporation & business, government & organization.

#### 2.1.4 Land Use Map

A land use map reflects the official ground-truth of the static functions of urban regions. In particular, the official land use map[2] published by the government of Shanghai uses six different land use designation types, i.e., residence, business, industry, public infrastructure, farming and forestry, and ecological restoration area. We use this official data as the land use dataset.

#### 2.1.5 Urban Economy

The urban economy reflects the development level of different urban regions, profoundly shaped by region functions and urban dynamics. In practice, Gross Domestic Product (GDP), the market value of all the final goods and services produced, is a commonly used measurement to quantify an area's economy [15]. Hence, in this work, we also use GDP to reflect the economic level of urban regions. Specifically, we obtained the official GDP data of 188 administrative districts in Shanghai from the *Shanghai Economy Almanac* (2017) [16]. The almanac is the most authoritative, complete, and systematic reference for Shanghai economic data.

### 2.2 Framework Overview

We present an overview of our proposed framework in Fig. 3. The complete process of the framework can be described as follows. First, based on road networks, we partition the city into multiple disjoint regions. These regions are treated as atomic units to study dynamic region functions. Then using this region data along with spatiotemporal app usage data, we construct a heterogeneous app usage graph. Next, we derive a relational location graph from the app usage graph with the POI distribution as region features. Then, we obtain dynamic region embeddings by feeding the attributed relational location graph into a graph auto-encoder model for training. The main purpose of the graph auto-encoder model is to fuse both graph structure and node feature information into the node embeddings. In our case, each region at a specific time interval has a corresponding embedding, representing the characteristics of that region in that time interval. Finally, we verify our model using three illustrative applications, including static land usage identification, dynamic region functions analysis, and economic level prediction. In practice, for static land usage identification, we use the official land use map as the ground truth, while for economic level prediction, we take GDP data of administrative districts as the ground truth.

2. http://www.shanghai.gov.cn/newshanghai/xxgkfj/2035003.pdf

## 3 METHOD

### 3.1 App Usage Based Graph

Mobile app usage data contains dynamic relationships between users, apps, and locations. To encode such connections between different entities, we first build a heterogeneous app usage graph and then formalize it as a meta-path guided homogeneous relational location graph.

#### 3.1.1 Heterogeneous App Usage Graph

The interactions in mobile app usage behavior can be abstracted as a heterogeneous graph containing three types of entities, i.e., users, apps, and locations. Fig. 4 shows the structure of the heterogeneous app usage graph, where $U$ refers to *user* nodes, $A$ refers to *app* nodes, $L$ refers to *time-enhanced location* nodes, and the edges reflect the co-occurrence of different objects in mobile app usage records. We note that since particular regions can exhibit different roles at different time slots, we use time-enhanced location

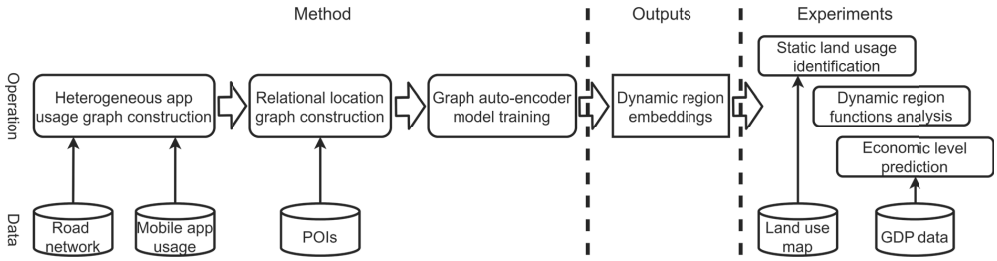



Fig. 3. Overview of our proposed framework.

nodes to represent these dynamic relationships. Specifically, each time-enhanced location node is denoted as $l_i^t$ representing the region $i$ at time slot $t$. Therefore, the set of time-enhanced location nodes contains $R \cdot T$ nodes, where $R$ is the number of regions, and $T$ is the number of time slots. For the sake of simplicity, in this paper, the term 'location node' refers to 'time-enhanced location node' by default.

To distinguish different connection strengths between nodes, we model the heterogeneous app usage graph as an undirected weighted graph. There are three types of edges, including *user app* edges $e(u, a)$ that reflect the usage of apps by users, *user time-enhanced location* edges $e(u, l^t)$ that reflect the trajectories of users, and *app time-enhanced location* edges $e(a, l^t)$ that reflect the spatiotemporal nature of app usage. In detail, we illustrate the method to compute edge weight as follows. We initially set all edge weights to zero, then for each app usage record $r =< u, a, l, t >$, we increment the edge weights $w(u, a)$, $w(u, l^t)$, $w(a, l^t)$. After traversing all records we obtain the final weights of all edges. Given the heterogeneity of the app usage graph, including multiple types of nodes and edges, we need to normalize the edge weights across different edge types. Thus, we normalize the weights using a co-occurrence count for each edge type separately. Specifically, we apply max-min normalization. For example, for *user time-enhanced location* edges $e(u, l^t)$, we set the normalized edge weight as,

$$\hat{w}(u, l^t) = \frac{w(u, l^t) - \min\limits_{u,\, l^t} \left(w(u, l^t)\right)}{\max\limits_{u,\, l^t} \left(w(u, l^t)\right) - \min\limits_{u,\, l^t} \left(w(u, l^t)\right)}, \quad \forall u \in U, l^t \in L, \quad (1)$$

where $\hat{w}(\cdot)$ denotes the normalized edge weight, $U$ is the set of user nodes, and $L$ is the set of time-enhanced location nodes. We then apply analogous normalizations to the other two types of edges, i.e., $e(u, a)$, and $e(a, l^t)$, and obtain the normalized weights $\hat{w}(u, a)$, and $\hat{w}(a, l^t)$, respectively.

### 3.1.2 Homogeneous Relational Location Graph

As we are interested in uncovering urban dynamics, i.e., learning representations of the time-enhanced location nodes, we next derive a homogeneous relational location graph from the heterogeneous app usage graph.

Inspired by PathSim [17], we apply a meta-path based method to the time-enhanced location nodes in the heterogeneous app usage graph. A meta-path defines a compositional relation connecting two entities while still accounting for the heterogeneity and semantics of nodes and edges between those entities. Such a structure is widely used to capture the similarity between nodes of the same type in a heterogeneous graph [18], [19].

**Definition 1.** ***Meta-path*** *[17]. A meta-path $\phi$ is defined as a path generation rule on a heterogeneous graph in the form of $V_1 \rightarrow V_2 \rightarrow \cdots \rightarrow V_l$, where $V$ denotes node types. In other words, a meta-path $\phi$ describes a composite connection relation between nodes of node types $V_1$ and $V_l$.*

**Definition 2.** ***Meta-path reachable nodes****. Given a meta-path $\phi$ and a node $v$, the meta-path reachable nodes $\mathcal{N}_v^{\phi}$ of node $v$ are a set of nodes connected with node $v$ through a path in the generated path set $P_\phi$ based on meta-path $\phi$.*

The key idea behind a meta-path is to generate a set of paths through the heterogeneous graph based on a semantic-aware relation. For example, considering the app usage graph, the meta-path of *Location-User-Location* (abbreviated as 'LUL') enables the system to start with a given location node and find other location nodes visited by the same user. In particular, as shown in Fig. 5, given the meta-path 'LUL', $L_1 \rightarrow U_2 \rightarrow L_2$ is an entity in the generated path set and $L_1$ and $L_2$ are meta-path reachable based on the meta-path 'LUL'.

Moreover, we observe that two nodes of the same type can be reachable via different meta-paths. By taking $L_1$ and $L_2$ in Fig. 5 as an example, apart from the meta-path 'LUL', they are also meta-path reachable based on the meta-path *Location-App-Location* (abbreviated as 'LAL') through the path $L_1 \rightarrow A_1 \rightarrow L_2$ which describes the co-app-usage relation between time-enhanced locations. Based on different meta-paths, the meta-path reachable connections reveal different semantic relations of nodes by exploiting the structural information in the heterogeneous graph.

Next, we employ the meta-path reachable connections to construct a homogeneous relational location graph, while retaining the structural information of the heterogeneous app usage graph. Specifically, there are two steps, $i$) path set generation, $ii$) relational connection construction.

1). **Path set generation**. Given a meta-path $\phi$, in this step, we generate a set of node paths $P_\phi$ guided by this meta-path. As the app usage graph is a weighted graph, we use weighted random walks [20] to generate node paths by considering the non-uniform preference of object selection. In particular, using **weighted** random walks provides two important benefits. First, it maintains the critical (high edge weight) connections between nodes in the heterogeneous graph. Second, it also avoids adding noise to the path sets by filtering out weak (low edge weight) connections. By using multiple meta-paths $\phi_1, \phi_2, ..., \phi_n$, we can generate corre-

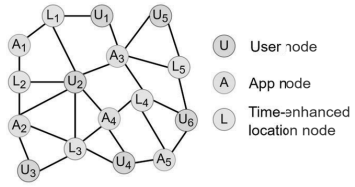



Fig. 4. An example of a heterogeneous app usage graph.

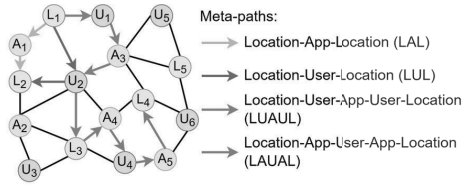



Fig. 5. An example of several meta-paths in an app usage graph.

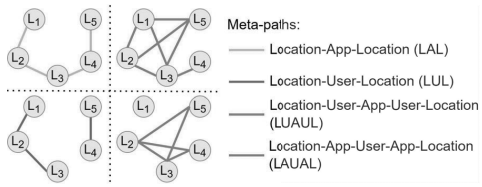



Fig. 6. The corresponding meta-path guided location graphs of the heterogeneous app usage graph from Fig. 4.

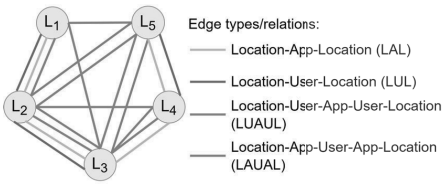



Fig. 7. The corresponding relational location graph containing all graph structures of the meta-path guided location graphs from Fig. 6.

sponding path sets $P_{\phi_1}, P_{\phi_2}, ..., P_{\phi_n}$ where $n$ is the number of meta-paths. Each path set has a semantic meaning and represents a specific structure in the heterogeneous graph.

2). **Relational connection construction**. Given multiple path sets $P_{\phi_1}, P_{\phi_2}, ..., P_{\phi_n}$ generated in step (1), in this step, we determine node connections in the location graph. Specifically, for each path set $P_\phi$, we build a meta-path guided location graph where location nodes will be connected if they are meta-path reachable. An example is depicted in Fig. 6. For meta-paths 'LAL', 'LUL', 'LUAUL', and 'LAUAL', we construct four meta-path guided homogeneous location graphs that correspond to those meta-paths, respectively. For different location graphs, their edges reflect different semantic meanings and relations. As all the location graphs have the same node sets, i.e., the set of time-enhanced location nodes, we can merge them using a relational graph to distinguish edges with different semantic meanings. Unlike a conventional graph, in a relational graph, edges have type attributes, where different edge types can represent different relations. An example is shown in Fig. 7 in which we construct a corresponding relational location graph that contains all connection structures of the meta-path guided location graphs in Fig. 6. Specifically, we distinguish different types of edges by different colors. In this way, we construct a homogeneous relational location graph in terms of the path sets $P_{\phi_1}, P_{\phi_2}, ..., P_{\phi_n}$ generated in step (1).

Using the above steps, we can derive a homogeneous relational location graph from the heterogeneous app usage graph. In particular, we denote the relational location graph as $G_{hom} = (L, E_\Phi, \Phi, H)$, where $L$ and $E_\Phi$ denote the sets of time-enhanced location nodes and relational edges, respectively. $\Phi$ is the set of relation types (i.e., meta-paths), and $H$ is the set of node features.

### 3.1.3 Node Features

In order to leverage the POI data of regions, we assign a feature vector $\boldsymbol{h}$ to each time-enhanced location node $l \in L$ in the homogeneous relational location graph $G_{hom}$. Specifically, the time-enhanced location node features contain two parts: static components $\boldsymbol{h_s}$ and dynamic components $\boldsymbol{h_d}$, where $\boldsymbol{h} = [\boldsymbol{h_s}, \boldsymbol{h_d}]$.

**Static components $\boldsymbol{h_s}$.** We use the density of nearby POIs to represent the static components of location node features. As illustrated in section 2.1.3, POI data depict various venues located in the region, such as shopping malls, theaters, parks, and office buildings. Thus, nearby POIs describe the inherent characteristics of that region. Since a region's POI distribution does not change dynamically over a single day, different time-enhanced location nodes that represent the same region have the same POI features. Namely, the POI features are static for individual regions.

In particular, we use the distribution of POI categories within a region as the numerical features of corresponding location nodes. Assuming the number of POI categories is $C$, for an arbitrary time-enhanced location node $l_i^t \in L$ standing for the region $i$ at time slot $t$, it has a POI category distribution vector $\boldsymbol{h^i_{POI}} = [h^i_{POI_1}, h^i_{POI_2}, ..., h^i_{POI_c}]$, where $h^i_{POI_c}$ is the number of POIs of category $c$ within region $i$. Moreover, for different POI categories, their popularity varies dramatically. Thus, POIs are not uniformly distributed across different categories. For instance, the number of restaurant POIs is much higher than that of education POIs. Therefore, to eliminate the imbalance among different POI categories, we normalize the POI category distribution vectors using the term frequency-inverse document frequency (TF-IDF) [21]. Mathematically, for a time-enhanced location node $l_i^t$, its normalized POI feature vector $\hat{\boldsymbol{h}}^{\boldsymbol{i}}_{\boldsymbol{POI}} = [\hat{h}^i_{POI_1}, \hat{h}^i_{POI_2}, ..., \hat{h}^i_{POI_c}]$ can be computed as,

$$\hat{h}^i_{POI_c} = \frac{h^i_{POI_c}}{\sum_{c=1}^{C} h^i_{POI_c}} \cdot \log \frac{R}{|\{\boldsymbol{h^i_{POI}} : h^i_{POI_c} > 0\}| + 1}, \forall c = 1, ..., C, \tag{2}$$

where $R$ is the number of regions, $\frac{h^i_{POI_c}}{\sum_{c=1}^{C} h^i_{POI_c}}$ represents the term frequency and $\frac{R}{|\{\boldsymbol{h^i_{POI}} : h^i_{POI_c} > 0\}|+1}$ represents the inverse document frequency. For each time-enhanced location node, the static component of the node's features $\boldsymbol{h_s}$ is the normalized POI feature vector, i.e., $\boldsymbol{h_s} = \hat{\boldsymbol{h}}_{\boldsymbol{POI}}$.

**Dynamic components $\boldsymbol{h_d}$.** We use the human mobility flows in a region within a time slot to represent the dynamic components of location node features. In particular, human

mobility flows describe people's arrive-stay-leave behavior. In detail, people arrive at a specific region and stay for a certain period, and then leave that region. Many previous studies have shown that human mobility flows within a region reflect that region's dynamic characteristics [22]. Specifically, areas with similar flow patterns have similar functions. We note that since human mobility flows in a region change over time in a day, the mobility flow features are dynamic for individual regions.

In our case, we use spatiotemporal mobile app data to infer human mobility flows. Given a specific region and a particular time slot, we use the difference in the number of active users between adjacent time slots to describe the mobility flows in that region during that time slot. Specifically, we compute the number of active users who arrive at, stay in, and leave from the region $i$ in time slot $t$ and denote them as $AR_i^t$, $ST_i^t$, $LV_i^t$, respectively. Next, we normalize the mobility flow features to the range of 0 to 1 by using the max-min normalization method. Mathematically, given a time-enhanced location node $l_i^t$, its normalized mobility flow features of $\hat{AR}_i^t$ is computed as,

$$\hat{AR}_i^t = \frac{AR_i^t - \min\limits_{i,\ t}\left(AR_i^t\right)}{\max\limits_{i,\ t}\left(AR_i^t\right) - \min\limits_{i,\ t}\left(AR_i^t\right)}, \qquad \forall i = 1, 2, ..., R, t = 1, 2, ..., T, \tag{3}$$

where $R$ and $T$ are the number of regions and time slots respectively. With analogous normalization operation, we then obtain normalized features of $\hat{ST}$ and $\hat{LV}$. In summary, for each time-enhanced location node, the dynamic component of the node's features $\boldsymbol{h_d}$ is the normalized mobility flow features, i.e., $\boldsymbol{h_d} = [\hat{AR}, \hat{ST}, \hat{LV}]$.

### 3.2 Auto-Encoder for Relational Location Graph

The information in the relational location graph $G_{hom} = (L, E_\Phi, \Phi, H)$ is contained in both the network structure and node features. Expressly, the node features represent the internal characteristics of time-enhanced locations, while the network structure depicts their relationships. In this section, we aim to learn a numerical representation for each time-enhanced location node by simultaneously considering both node features and network structure.

#### 3.2.1 Graph Auto-encoder

Specifically, we utilize a deep auto-encoder framework for learning time-enhanced location embeddings. An auto-encoder is an unsupervised neural network model, which consists of two parts: a graph encoder and a graph decoder. The whole architecture of the framework is shown in Fig. 8. The encoder projects the original node feature matrix $H$ to a hidden representation $Z$. While the decoder attempts to reconstruct the node feature matrix $H'$ from the generated hidden representation $Z$. The auto-encoder framework aims to guarantee that the reconstructed node feature matrix $H'$ is as similar to the original feature matrix $H$ as possible. Also, in order to introduce network structure information into the hidden representation $Z$, both graph encoder and decoder characterize node features over the relational location graph $G_{hom}$ by using relational graph attention networks (i.e., Rel-GAT).

#### 3.2.2 Relational Graph Attention Network

In this section, we detail the implementation of the relational graph attention network (i.e., Rel-GAT). In particular, the relational graph attention network is derived from a graph neural network [23] that leverages both the local graph structure and node features for node embeddings. The critical idea of a graph neural network is to aggregate and propagate node features in terms of the graph structure. In detail, the computation of graph neural networks is carried out in two steps: (i) message propagating, (ii) aggregating and updating. In the message propagating step, each node passes its representation vector to its neighbors. In the aggregating and updating step, each node first aggregates the received representation vectors and then updates its representation with the aggregation.

Since the homogeneous relational location graph carries multiple edge types, i.e., relations, we enhance the conventional propagation and aggregation operations to relation-specific operations and further introduce the attention mechanism [24] to effectively measure the aggregation weight between two nodes. We name the newly designed graph neural network as the relational graph attention network, Rel-GAT. Given a time-enhanced location node $l_i^t$ representing location $i$ at time slot $t$, after a single Rel-GAT layer, its representation is computed as,

$$\boldsymbol{h}_{l_i^t}^{(k+1)} = \sigma\left(\sum_{\phi\in\Phi}\sum_{j\in\mathcal{N}_{l_i^t}^{\phi}}\left(\alpha_{\phi,j,l_i^t}^{(k)} \cdot W_\phi^{(k)} \cdot \boldsymbol{h}_j^{(k)}\right) + W_0^{(k)} \cdot \boldsymbol{h}_{l_i^t}^{(k)}\right), \tag{4}$$

where $\boldsymbol{h}_{l_i^t}^{(k)}$ is the representation vector of node $l_i^t$ in the $k$-th layer of the neural network, $\mathcal{N}_{l_i^t}^{\phi}$ is the set of neighbors of node $l_i^t$ under the edge relation $\phi$, $\alpha_{\phi,j,l_i^t}^{(k)}$ is the aggregation weight indicating the importance of node $j$'s representation to the node $l_i^t$ under relation $\phi$ in the $k$-th layer of the neural network, $W_\phi^{(k)}$ is a relation-specific transformation matrix of relation $\phi$, $\sigma(\cdot)$ is an activation function. Intuitively, (4) aggregates relation-specific transformed representations of neighbors through a set of corresponding relational edges. Additionally, to ensure that the representation of a node at the $(k+1)$-th layer is informed by its previous representation at $k$-th layer, we add a self-loop to each node and introduce a relation-specific transformation matrix $W_0$ for self-loop connections.

Inspired by the graph attention network [25], we adopt a shared attention mechanism to determine the aggregation weight $\alpha_{\phi,j,l_i^t}^{(k)}$ with node representations as inputs. The attention mechanism can be expressed as,

$$\alpha_{\phi,j,l_i^t}^{(k)} = \frac{\exp\left(\sigma\left(\boldsymbol{c}_{\boldsymbol{\phi}}^{\mathbf{T}}\left[W_\phi^{(k)} \cdot \boldsymbol{h}_{l_i^t}^{(k)} \| W_\phi^{(k)} \cdot \boldsymbol{h}_j^{(k)}\right]\right)\right)}{\sum_{m\in\mathcal{N}_{l_i^t}^{\phi}}\exp\left(\sigma\left(\boldsymbol{c}_{\boldsymbol{\phi}}^{\mathbf{T}}\left[W_\phi^{(k)} \cdot \boldsymbol{h}_{l_i^t}^{(k)} \| W_\phi^{(k)} \cdot \boldsymbol{h}_m^{(k)}\right]\right)\right)}, \tag{5}$$

where $\boldsymbol{c}_\phi$ is the attention vector under relation $\phi$. $\cdot^{\mathbf{T}}$ and $\|$ represent transposition and concatenation operations, respectively. Intuitively, the attention mechanism captures the feature proximity between pairs of nodes. In other words, two nodes with higher representational similarity will have a larger aggregation weight.

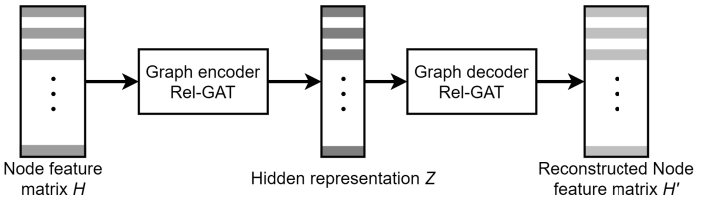



Fig. 8. The architecture of the graph auto-encoder. The graph encoder and decoder capture node features from the relational location graph by using relational graph attention networks, i.e., Rel-GAT.

### 3.3 Learning and Training

As mentioned the graph encoder consists of a stack of relational graph attention networks. Furthermore, following the auto-encoder architecture, the graph decoder consists of the same number of relational graph attention networks, and the decoder's hidden units are symmetric to the graph encoder's hidden units. Therefore, by providing node feature matrix $H$ to the graph encoder, we will obtain a reconstructed feature matrix $H'$, i.e., the graph decoder's output. In this work, we take the Euclidean distance between $H$ and $H'$ as the reconstruction loss, which is computed as,

$$\mathcal{L}\left(H, H'\right) = \left\| H - H' \right\|^2 . \tag{6}$$

Guided by this loss function, we can optimize the graph auto-encoder via the back-propagation method [26]. Specifically, in the graph auto-encoder model, the relation-specific transformation matrix $W_\phi$ and attention vector $\boldsymbol{c}_\phi$ of each network layer are trainable parameters. After training, we take the hidden representation $Z$, i.e., the output of the graph encoder, as the embedding of the time-enhanced location nodes, which represents information of both node features and network structure. In this way, all time-enhanced location nodes are projected into the same latent space. By investigating how a region embedding changes over time, we can reveal the urban dynamics accordingly.

## 4 Experiment

In this section, we evaluate our proposed model through a set of experiments conducted on city-scale real-world datasets. We first introduce the experimental setup, including data preprocessing, baselines, and parameter settings. Next, we analyze the learned dynamic region embeddings and their semantic meanings in detail. We finally reveal the relationship between regions' dynamic functions and their economic development.

### 4.1 Experiment Setup

#### 4.1.1 Data Preprocessing

In this work, we focus on the whole metropolitan area of Shanghai, one of the world's largest cities. In particular, we utilize five kinds of ubiquitous data, including mobile app usage, road network, POI data, land use map, and urban economic data. The details of the above datasets are explained in section 2.1.

We first use the road network to partition the metropolitan area of Shanghai into 1,595 disjoint regions. Next, we employ a large-scale mobile app usage dataset collected in Shanghai to construct the heterogeneous app usage graph. As stated in section 2.1, we utilize a subset containing the top 100000 users ranked by their total number of usage records. Each app usage record contains an anonymized user ID, app ID, base station ID, and timestamp. In practice, we evenly divide one day into 12 time-slots. Through mapping base stations to regions and timestamps to time-slots, we build a heterogeneous app usage graph by following the procedure from section 3.1.1. The heterogeneous app usage graph has 100000 user nodes, 2000 app nodes, and 19140 ($1595 \times 12$) time-enhanced location nodes.

#### 4.1.2 Baselines

We compare our model with four commonly used and state-of-the-art approaches for urban exploration.

• **POI**. An intuitive approach is to represent a region using intra-region POI data. We use TF-IDF to measure different POI categories' importance to a region. Specifically, each region can be represented by a $C$-dimensional vector, where $C$ is the total number of unique POI categories. This baseline only considers the static features of regions.

• **Hidden Markov model (HMM)** [9]. HMM is a state-of-the-art method for modeling urban dynamics with app usage data. In this method, we build a state-sharing hidden Markov model by jointly using intra-region app usage features and human mobility flows. For one region at a time slot, it endows a state for that region. Each region can be represented by a state sequence across all time slots. However, this baseline cannot represent or model urban dynamics precisely because of the limited number of states.

• **DeepWalk** [27]. DeepWalk extends the word2vec [28] model to the scenario of network embedding. DeepWalk uses local information obtained from truncated random walks and the skip-gram model to learn node embeddings. In the experiments, we employ DeepWalk on the heterogeneous app usage graph and obtain the embeddings of time-enhanced location nodes. Specifically, each region can be represented by a vector, which is the average of its embeddings in all time slots. In this case, the embedding only reflects the graph structure and neglects the heterogeneity of the graph.

• **Metapath2Vec** [18]. Metapath2Vec employs meta-path based random walks to construct the heterogeneous neighborhood of nodes and then leverages the skip-gram model to perform node embeddings. In the experiments, we take the heterogeneous app usage graph as input and use 'LAL', 'LUL', 'LUAUL', and 'LAUAL' as meta-path schemes. Like DeepWalk, we represent each region by using the average of its embeddings in all time slots. Although Metapath2Vec takes the graph's heterogeneity into account, the method is still limited by the inability to leverage node features.

TABLE 2
Performance of Graph Auto-encoder (our proposed model) and baseline methods for static land use identification. NMI refers to normalized mutual information, ARI refers for adjusted rand index, and Imp. refers to improvement.

| Model | NMI | Imp. on NMI | ARI | Imp. on ARI | F-score | Imp. on F-score |
|---|---|---|---|---|---|---|
| POI | 0.3359 | 103.22% | 0.2926 | 122.52% | 0.3505 | 120.14% |
| DeepWalk | 0.4459 | 53.08% | 0.3937 | 65.38% | 0.4971 | 55.22% |
| Metapath2Vec | 0.5121 | 33.29% | 0.4332 | 50.30% | 0.5673 | 36.01% |
| HMM | 0.5749 | **18.73%** | 0.5183 | **25.62%** | 0.6460 | **19.44%** |
| Graph Auto-encoder | **0.6826** | - | **0.6511** | - | **0.7716** | - |

• **Graph Auto-encoder**. Our proposed method. Given the heterogeneous app usage graph, we construct the corresponding relational location graph guided by meta-paths 'LAL', 'LUL', 'LUAUL', and 'LAUAL'. By feeding the relational location graph into the Rel-GAT-based graph auto-encoder, we obtain the embeddings of time-enhanced location nodes. Our model considers the graph's heterogeneity and uses meta-paths to leverage the semantics of different types of edges and nodes. Also, we use graph neural networks and an auto-encoder framework to fuse both node features and graph structure information into node embeddings.

#### 4.1.3 Implementation Details and Parameter Settings

We implement our model with Pytorch[3] and Deep Graph Library[4] and train on a NVIDIA GTX 2080Ti GPU. In the training procedure, we randomly initialize parameters and use Adam [29] to optimize the model with a learning rate of 0.001. The dimension of the attention vector $\boldsymbol{c}$ is 128. Also, for the sake of fair comparison, we set the dimension of the node embeddings to 64 for DeepWalk, Metapath2Vec, as well as, Graph Auto-encoder.

### 4.2 Identifying Static Land Usage

For the task of identifying static land usage, we perform k-means clustering on region representations to partition regions into $k$ clusters. Regions with similar static land usage should, in theory, be assigned to the same cluster. Specifically, for the HMM baseline each region is represented by a state sequence across all time slots. While for the graph embedding methods, including DeepWalk, Metapath2Vec, and our model (i.e., Graph Auto-encoder), we represent each region by using the average of its embeddings in all time slots, which is computed as,

$$z_{l_i} = \frac{1}{T}\sum_{t=1}^{T} z_{l_i^t}, \tag{7}$$

where $z_{l_i^t}$ is the embedding of time-enhanced location node $l_i^t$ representing the region $i$ at time slot $t$, and $T$ is the total number of time slots. We note that $z_{l_i}$ merges all dynamic embeddings together and represents the typical embedding of the region $i$ across all time slots.

To validate identification performance, we use the official land use map of Shanghai as the ground-truth. As depicted in Fig. 9(a), the map classifies land use into 6 categories, i.e., residence, business, industry, public infrastructure, farming and forestry, and ecological restoration area. Therefore, we partition regions into 6 clusters by using k-means and setting $k = 6$. Next, we use the following metrics to evaluate the region clustering results of our proposed method and baselines.

• **Normalized Mutual Information (NMI)**. NMI is a widely used metric to measure the purity of clustering results from an information-theoretic perspective. NMI is computed as,

$$\text{NMI} = \frac{I(L,C)}{[H(L)+H(C)]/2}, \tag{8}$$

where $L$ is the set of ground-truth labels, and $C$ is the set of clustering labels. $I(L,C)$ denotes the mutual information between ground-truth and clustering labels. $H(L)$ and $H(C)$ represent the entropy of ground-truth and clustering label sets, respectively. The scale of NMI ranges from 0 (no mutual information) to 1 (perfect correlation). Thus, a higher NMI indicates that the clustering results are closer to the ground-truth.

• **Adjusted Rand Index (ARI)**. ARI is the corrected-for-chance version of the Rand index [30]. First, the Rand Index (RI) computes a similarity measure between clustering labels and ground-truth labels, which is computed as,

$$\text{RI} = \frac{TP+TN}{TP+FP+TN+FN}, \tag{9}$$

where $TP$ is the number of true positives, $TN$ is the number of true negatives, $FP$ is the number of false positives, and $FN$ is the number of false negatives. ARI makes a correction for chance by establishing a baseline, i.e., random labeling, which is defined as,

$$\text{ARI} = \frac{RI - Expected\_RI}{max(RI) - Expected\_RI}. \tag{10}$$

Thus, ARI is ensured to have a value close to 0 for random labeling and equal to 1.0 when the clustering results match ground-truth perfectly. That is, the higher the ARI, the better the clustering performance.

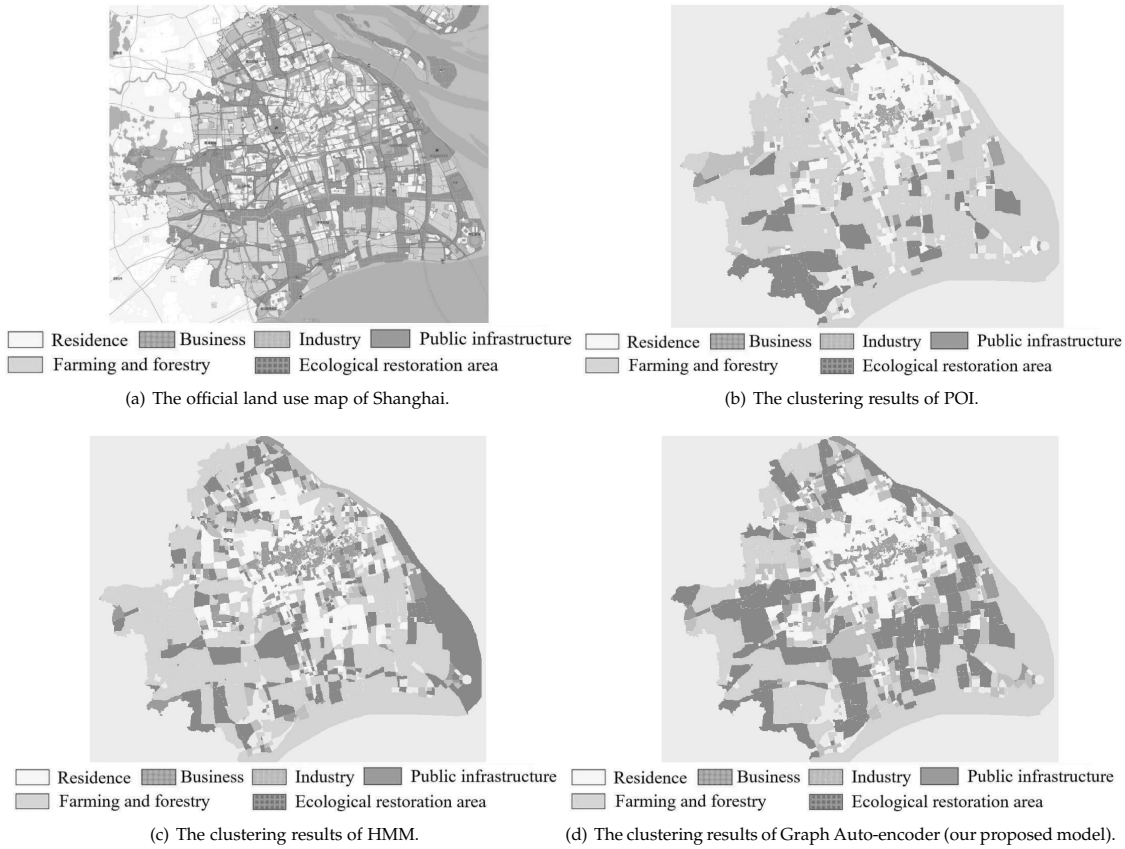
• **F-score**. F-score is a measure of clustering accuracy, which is calculated from precision and recall.

$$\text{F-score} = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}, \tag{11}$$

Specifically, the higher the F-score, the better the clustering results. The maximum F-score is 1 and minimum is 0.

The evaluation results are shown in Table 2. From the results, we have the following key observations. 1). Graph Auto-encoder performs the best among all methods by a large margin. Compared with the best baseline, Graph Auto-encoder shows an improvement of 18.73%, 25.62%, and 19.44%, in terms of NMI, ARI, and F-score, respectively. 2). The network embedding methods, including DeepWalk and Metapath2Vec, show better performance than the POI

3. https://pytorch.org/.
4. https://www.dgl.ai/.




(a) The official land use map of Shanghai.

(b) The clustering results of POI.

(c) The clustering results of HMM.

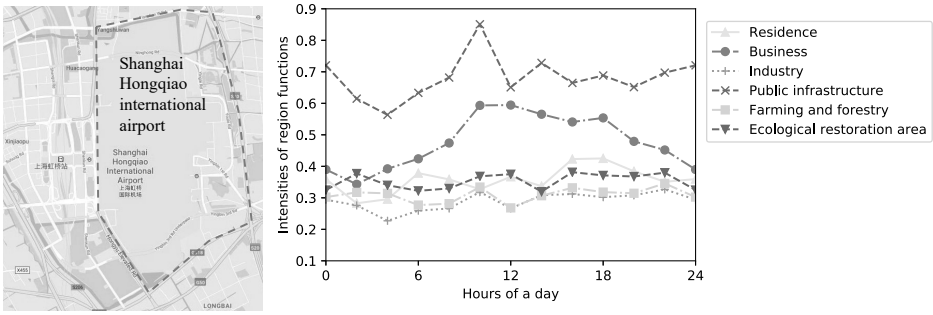
(d) The clustering results of Graph Auto-encoder (our proposed model).

Fig. 9. The official land use map of Shanghai and region clustering results of POI, HMM, and Graph Auto-encoder. Each cluster is denoted by a unique color.

method, implying that mobile app usage data are more informative for region profiling compared with POI distribution. Moreover, the results demonstrate the effectiveness of using graph structures to model app usage behavior and the inter-relations between regions. Also, likely due to the use of meta-path, the performance of Metapath2Vec is slightly better than DeepWalk. 3). HMM shows the best performance among all baselines. The main reason is that HMM jointly uses mobile app usage and human mobility flows as region features. However, compared with Graph Auto-encoder, HMM only leverages individual region features and neglects interactions between regions, which leads to performance degradation.

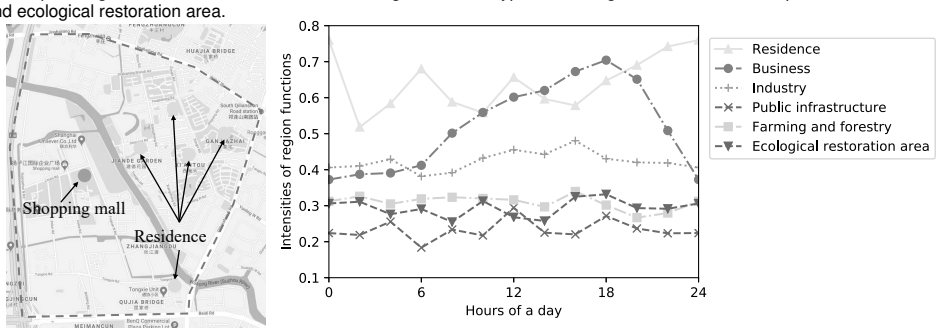
Furthermore, in order understand the clustering differences in depth, we select the models of POI, HMM, and Graph Auto-encoder and visualize the clustering results in Fig 9, where color denotes regions in the same cluster. We notice that using POI distributions can identify the central business area (red) and residence area (yellow). An important reason is that the POI categories of residence, restaurant, shopping mall, and corporation & business are popular and have sufficient records. On the other hand, since the other POI categories are less common, the land-use types like public infrastructure and industry can not be easily identified. Although we use TF-IDF to mitigate this uneven distribution of POI data, the model still does not perform well compared with other baselines. Alternatively, through leveraging mobile app usage data, HMM can differentiate the public infrastructure areas (purple), e.g., the airport. This supports the findings of previous studies [4], [31] that mobile users at airports have unique app usage patterns. Moreover, as HMM also leverages human mobility flow patterns, HMM has the ability to recognize the farming and forestry area (light green) and ecological restoration area (dark green) to some extent. In terms of Fig. 9(d), we observe that our proposed model, Graph Auto-encoder, can accurately identify all six land-use types. The main reason is that apart from intra-region features, Graph Auto-encoder also builds a relational location graph to leverage various inter-relations among regions.

In summary, through the land use identification task we demonstrate the effectiveness of learned embeddings in representing region properties. More importantly, we obtain six anchor embeddings, i.e., centroids of the six clusters, representing the six types of region functions. Specifically, we use $z_R$, $z_B$, $z_I$, $z_P$, $z_F$, and $z_E$ to denote the anchor




(a) The map of Region A. (b) The dynamic region functions of Region A.

Fig. 10. The map of Region A and the intensities of the six region function types, including residence, business, public infrastructure, farming and forestry, and ecological restoration area.




(a) The map of Region B. (b) The dynamic region functions of Region B.

Fig. 11. The map of Region B and the intensities of the six region function types, including residence, business, public infrastructure, farming and forestry, and ecological restoration area.

embeddings of residence, business, industry, public infrastructure, farming and forestry, and ecological restoration area, respectively. Given these anchor embeddings, with distinct semantic meaning, we can further reveal how the region functions change over the course of a day.

### 4.3 Revealing Dynamic Region Functions

In this section, we aim to investigate the changes in region functions throughout the day. In particular, for a region $i$ at time slot $t$, we measure its region functions by computing the cosine similarity between its embedding $z_{l_i^t}$ and the anchor embeddings. For example, a regions intensity of residence function $z^R_{l_i^t}$ is computed as,

$$z^R_{l_i^t} = cos(z_{l_i^t}, z_R), \tag{12}$$

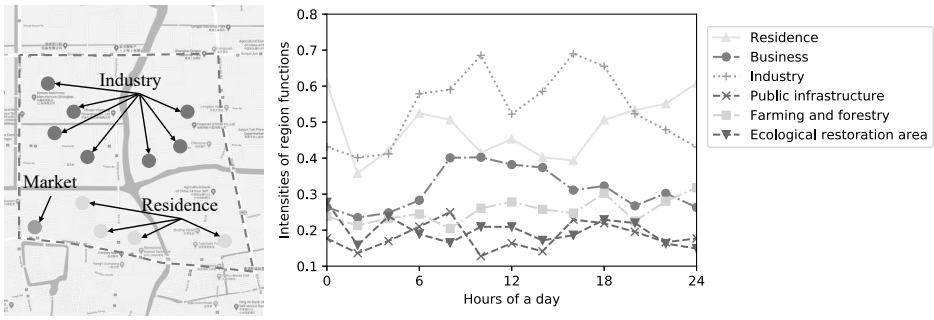
where $cos(\cdot)$ represents the cosine similarity, and $z_R$ denotes the anchor embedding of residence. Similarly, we can obtain the function intensities of business, public infrastructure, farming and forestry, and ecological restoration area and denote them as $z^B_{l_i^t}$, $z^I_{l_i^t}$, $z^P_{l_i^t}$, $z^F_{l_i^t}$, and $z^E_{l_i^t}$, respectively. In this way, we convert the time-enhanced region embedding $z_{l_i^t}$ into a vector $[z^R_{l_i^t}, z^B_{l_i^t}, z^I_{l_i^t}, z^P_{l_i^t}, z^F_{l_i^t}, z^E_{l_i^t}]$ with semantic meanings, representing the intensities of six region function types.

Given a region, we reveal its dynamic region functions by depicting how the region's intensities of the six region function types change over the course of a day. Due to space constraints, we only show our analysis of three regions.

**Region A.** First, we take Shanghai Hongqiao international airport as an example to analyze how its intensities of the six region function types change throughout the day. As shown in Fig. 10(b), Region A, i.e., the international airport, has a higher intensity of public infrastructure function compared with other function types. This corresponds to the official land use map, marking the airport as public infrastructure. Moreover, we detect that Region A has a business function during the daytime, which might be due to duty-free shops and restaurants located in the airport. As a result, by using time-enhanced location embeddings, we can reveal region functions from all perspectives. In other words, apart from the most significant function type, we can still ascertain the intensity of other types of functions.

**Region B.** The map of Region B and its dynamic region functions are depicted in Fig. 11. Specifically, the area of Region B is denoted by a red dotted polygon in Fig. 11(a). In the official land use map, Region B is classified as a residence type. We can observe that there are five major residential areas in Region B, denoted by yellow dots. Also, in terms of Fig. 11(b), Region B has a high intensity of residence function throughout the day, corresponding to the official land use map classification. Moreover, we still notice that the intensity of residence function of Region B fluctuates over the day, peaking at night with valleys at around 11.00 and 16.00. One possible reason is the working rhythm of people. When people leave home and go to work, the intensity of the residence function is weakened due to population decrease.

Meanwhile, a large shopping mall is located in Region B, indicated by a red dot, which causes Region B to exhibit a business function to some extent. As depicted in Fig. 11(b), the intensity of business function rises during the daytime




(a) The map of Region C. (b) The dynamic region functions of Region C.

Fig. 12. The map of Region C and the intensities of the six region function types, including residence, business, public infrastructure, farming and forestry, and ecological restoration area.

and reaches a peak at around 18.00. Between 14.00 and 18.00, the intensity of business function overrides the residence function, which indicates that the most significant region function changes from residential to business.

**Region C.** We depict the map of Region C and its dynamic region functions in Fig. 12. In the official land use map, Region C is marked as industrial. Nevertheless, according to Fig. 12(a), Region C is a mosaic consisting of seven industry areas (marked by brown dots), four residence areas (marked by yellow dots), and one market (marked bya red dot). From Fig. 12(b), we can observe that the industry function, as the most significant function type, has the highest intensity during the daytime, from 6.00 to 20.00. Alternatively, after 20.00, the residence function becomes the dominating function type. Again, the main reason is likely daily working rhythms. Moreover, a market is located in Region C, which gives the region a business function. However, compared with residence and industry, the business function intensity is weak.

In summary, we have successfully identified and revealed dynamic region functions based on the learned time-enhanced location embeddings. Although we only discussed three specific regions, the same analytic approach can be used to describe other remaining regions.

### 4.4 Predicting Economic Levels of Districts

Naturally, an area's urban functions are highly related to the area's economic development. In this section, we aim to predict districts' economic levels by using the dynamic functions as input features. In practice, we use GDP data as a measure of economic development for each district. From the Shanghai Economy Almanac (2017) [16], we obtain the official GDP data of the 188 administrative districts of Shanghai, ranging from 21.75 to 671.11 and with an average of 142.66[5]. Next, we discretize the GPD data into three levels, i.e., $[21.75, 42.66)$, $[42.66, 242.66)$, and $[242.66, 671.11]$.

Since one administrative district contains multiple regions, we represent its intensities of dynamic functions by averaging all regions in that district. For instance, given a district $d$, its intensity of residence function at time slot $t$ is expressed as,

$$z^R_{d,t} = \frac{1}{N_d} \sum_{i \ in \ d} (z^R_{l^t_i}), \tag{13}$$

where $i \ in \ d$ denotes the region $i$ in district $d$, $N_d$ represents the number of regions in district $d$, $z^R_{l^t_i}$ is the intensity of residence function of region $i$ at time slot $t$. With a similar method, we calculate the intensities of other types of functions. As we have six types of functions and twelve time-slots, for one district, we obtain a vector of 72 ($6 \times 12$) dimensions to indicate the intensities of the six function types over the day. We then take this vector as an input to predict its economic level.

We conduct a 5-fold cross-validation experiment using three popular classifiers, i.e., logistic regression, support vector machine, and random forest, to predict district economic levels. Table 3 presents the classification performance in terms of precision, accuracy, and F-score. We can observe that random forest achieves the best performance with an F-score of 0.8265, outperforming the linear classifiers, i.e., logistic regression and support vector machine. Also, such a high F-score and accuracy illustrates the strong correlations between dynamic functions and the economic development of administrative districts.

We further explore the importance of dynamic functions for predicting economic level by computing the mean decrease impurity (MDI) of all input features when using the random forest model. Specifically, the MDI of a feature is calculated as the total reduction of the criterion brought by that feature, which is also known as the Gini importance. The higher the MDI, the more important the feature. Fig. 13 shows the MDI score of six function types across different time slots for predicting the economic level. We can observe that the same function type has different importances at different time slots, thus validating the use of dynamic functions. Specifically, the intensities of residence and business functions have higher importance in the evening, i.e., between 18.00 and 22.00. While, the intensities of industry, farming and forestry, and ecological restoration functions are of higher importance in the morning. Such differences might be caused by human flow interactions across different functional areas throughout the day.

In summary, through the application of economic development prediction, we confirm the strong correlations between dynamic functions and the economic development of administrative districts. Also, the importance (for economic development prediction) of different function types varies over the day.

5. The unit is 100 million RMB.

TABLE 3
Performance of several classifiers using district-level dynamic function features for economic (GDP) level prediction.

| Method | Precision | Accuracy | F-score |
|---|---|---|---|
| Logistic regression | 0.6655 | 0.8157 | 0.7330 |
| Support vector machine | 0.7780 | 0.4211 | 0.5215 |
| Random forest | 0.8616 | 0.8421 | 0.8265 |

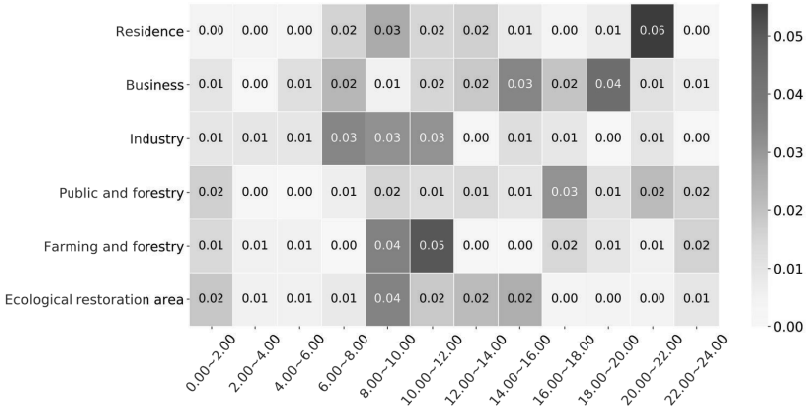



Fig. 13. The importance of different dynamic function features for economic (GDP) level prediction of districts.

## 5 Discussion

### 5.1 Implications

Overall, the proposed method and results illustrate the expressive power of such graph based methods to capture important urban dynamics. In terms of the actual implications, given the ubiquity of mobile networks and the ability to collect app usage data, countries and cities could use our method to validate their existing region function classifications. Specifically, the city could first meticulously validate a sample of region functions and then use that sample to train our method models. Then they could apply these models to the entire city and check regions where there is a discordance between the model and the existing function class.

Additionally, the ability to provide dynamic region functions could help in public and private (e.g., business) decision making by allowing more nuanced decisions about, for example, funding or public support for certain regions. As another example, regions (even potentially across cities) with similar dynamic region functions likely face similar urban problems, such as those related to mobility, zoning, and development, and thus studying similar regions can be useful for cross-pollination of creative urban solutions.

### 5.2 Limitations

There are also several limitations of our study that are worth mentioning and discussing. Firstly, the datasets themselves imply some limitations just based on their nature. Specifically, for the app usage dataset, since the dataset is collected by the ISP, any apps that do not produce significant network traffic are not captured, though the number of such apps should be minimal. Additionally, our dataset only covers one major city; thus, additional studies with other cities (with different regional function landscapes) could help further validate our results. Such validation would be especially important in cases of transfer learning across varying cities. Relatedly, the app usage dataset only covers a single week, thus preventing longitudinal analysis from seeing how these dynamic functions evolve over longer timescales (like months for seasonal changes or years for infrastructure and building changes).

## 6 Related Work

### 6.1 Spatiotemporal App Usage Behavior Analysis

Many prior studies have shown that mobile app usage behavior is strongly linked to spatial context. For instance, Mehrotra *et al.* [10] collected mobile app usage data from 26 students over two weeks. After employing the analysis of variance (ANOVA) method, they determined that users are more attentive to app notifications at college, libraries, and residential areas. While, users are less receptive to app notifications at religious institutions. Do *et al.* [4] and Bohmer *et al.* [31] found that users prefer to use web and multimedia apps in the airport while waiting for trips. Moreover, Graells-Garrido *et al.* [11] analyzed a city-scale app usage dataset and found different app usage patterns on different street types, i.e., main street, secondary street, and pedestrian. For example, on main streets message apps consume more traffic, while dating apps are used more on pedestrian streets.

Alternatively, some studies leverage spatial context for better app usage prediction. For example, Parate *et al.* [32] split the app usage sequence into a variable-length Markov chain according to the prior user location, thereby achieving location-aware app usage transition modeling and prediction. Chen *et al.* [33] proposed a graph-based model call CAP to iteratively learn node embeddings from app-location, app-time, and app-category graphs. They then combined node embeddings with a user representation to predict the future app. Further, Xia *et al.* [34] designed a recurrent neural network-based model to simultaneously capture spatial

and temporal app usage patterns for prediction. Specifically, in their model, app usage history, locations, and time are jointly embedded. The next used app and visited location are then jointly predicted.

Different from previous works focusing on app-oriented tasks, in this work, we leverage mobile app usage data in a location-oriented task by exploring app-location relationships. Our study opens the door to utilizing spatiotemporal app usage data in urban analytics.

### 6.2 Graph-based Representation Learning

Graph-based representation learning aims to learn low-dimensional vectors to represent nodes in a graph by exploring graph structure information. Inspired by word embedding [28], a series of algorithms were proposed to learn node representations based on the skip-gram model [27], [35]. In general, they first applied a random walk method on the graph to generate a series of node sequences. They then treated node sequences as the equivalent of sentences and feed them into the skip-gram model to obtain node embeddings. In particular, Perozzi *et al.* [27] employed the original random walk algorithm. While Grover *et al.* [35] used a biased random walk procedure to explore different graph structures. For heterogeneous graphs, Dong *et al.* [18] proposed a meta-path-based random walk approach to explore the semantics of different node types.

Alternatively, many graph embedding methods are based on graph neural networks (GNNs). Unlike random walk-based methods, graph neural networks learn node representations by considering both graph structure and node features. Kipf *et al.* [23] proposed an influential model, graph convolutional network, which performs convolutional operations by using the graph Laplacian matrix. Furthermore, Velivckovic *et al.* [25] proposed the graph attention network that applies an attention mechanism to determine the relative importance of neighbor information for the target node. However, these models can only cope with supervised learning tasks, and thus they cannot be directly used in our scenario. Therefore, in our work, we design a graph auto-encoder model to extend GNNs to unsupervised learning cases.

## 7 Conclusions

In this paper, we reveal urban dynamic functions by jointly leveraging spatiotemporal mobile app usage behavior and POI data. We propose a graph-based representation learning framework that maps time-enhanced regions into the same latent space. Thus, a region at a specific time interval is represented by an embedding vector. For one region, the region's dynamic embeddings characterize how its functions change over the course of a day. A series of experiments, including static land usage identification, dynamic region functions analysis, and economic (GDP) level prediction, demonstrate the superiority and the effectiveness of our framework. The study brings a new angle to urban analytics by leveraging mobile app usage data and lights the way for further urban-related applications, including urban planning, urban dynamic modeling, and economic analyses.

## References


[1] J. Yuan, Y. Zheng, and X. Xie, "Discovering regions of different functions in a city using human mobility and pois," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2012, pp. 186–194.

[2] P. Wang, Y. Fu, J. Zhang, X. Li, and D. Lin, "Learning urban community structures: A collective embedding perspective with periodic spatial-temporal mobility graphs," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 9, no. 6, pp. 1–28, 2018.

[3] J. Wang, J. Wu, Z. Wang, F. Gao, and Z. Xiong, "Understanding urban dynamics via context-aware tensor factorization with neighboring regularization," *IEEE Transactions on Knowledge and Data Engineering*, 2019.

[4] T. M. T. Do, J. Blom, and D. Gatica-Perez, "Smartphone usage in the wild: a large-scale analysis of applications and context," in *Proceedings of the 13th international conference on multimodal interfaces*, 2011, pp. 353–360.

[5] I. Andone, K. Błaszkiewicz, M. Eibes, B. Trendafilov, C. Montag, and A. Markowetz, "Menthal: a framework for mobile data collection and analysis," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, 2016, pp. 624–629.

[6] D. Yu, Y. Li, F. Xu, P. Zhang, and V. Kostakos, "Smartphone app usage prediction using points of interest," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 4, pp. 1–21, 2018.

[7] Y. Ouyang, B. Guo, T. Guo, L. Cao, and Z. Yu, "Modeling and forecasting the popularity evolution of mobile apps: a multivariate hawkes process approach," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 2, no. 4, pp. 1–23, 2018.

[8] S. Zhao, J. Ramos, J. Tao, Z. Jiang, S. Li, Z. Wu, G. Pan, and A. K. Dey, "Discovering different kinds of smartphone users through their application usage behaviors," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2016, pp. 498–509.

[9] T. Xia and Y. Li, "Revealing urban dynamics by learning online and offline behaviours together," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, no. 1, pp. 1–25, 2019.

[10] A. Mehrotra, S. R. Müller, G. M. Harari, S. D. Gosling, C. Mascolo, M. Musolesi, and P. J. Rentfrow, "Understanding the role of places and activities on mobile phone interaction and usage patterns," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 3, pp. 1–22, 2017.

[11] E. Graells-Garrido, D. Caro, O. Miranda, R. Schifanella, and O. F. Peredo, "The www (and an h) of mobile application usage in the city: The what, where, when, and how," in *Companion Proceedings of the The Web Conference 2018*, 2018, pp. 1221–1229.

[12] J. Lee and D.-H. Shin, "Targeting potential active users for mobile app install advertising: An exploratory study," *International Journal of Human–Computer Interaction*, vol. 32, no. 11, pp. 827–834, 2016.

[13] Y. Zheng, T. Liu, Y. Wang, Y. Zhu, Y. Liu, and E. Chang, "Diagnosing new york city's noises with ubiquitous data," in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2014, pp. 715–725.

[14] B. Map, "Baidu map api platform," http://lbsyun.baidu.com/, 2020.

[15] P. C. Sutton, C. D. Elvidge, T. Ghosh *et al.*, "Estimation of gross domestic product at sub-national scales using nighttime satellite imagery," *International Journal of Ecological Economics & Statistics*, vol. 8, no. S07, pp. 5–21, 2007.

[16] S. E. Almanac, "Development research center of shanghai municipal people's government," 2017.

[17] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, "Pathsim: Meta path-based top-k similarity search in heterogeneous information networks," *Proceedings of the VLDB Endowment*, vol. 4, no. 11, pp. 992–1003, 2011.

[18] Y. Dong, N. V. Chawla, and A. Swami, "Metapath2vec: Scalable representation learning for heterogeneous networks," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '17, 2017, p. 135–144.

[19] S. Fan, J. Zhu, X. Han, C. Shi, L. Hu, B. Ma, and Y. Li, "Metapath-guided heterogeneous graph neural network for intent recommendation," in *Proceedings of the 25th ACM SIGKDD International*

*Conference on Knowledge Discovery & Data Mining*, 2019, pp. 2478–2486.

[20] F. Vahedian, R. D. Burke, and B. Mobasher, "Weighted random walks for meta-path expansion in heterogeneous networks." in *RecSys Posters*, 2016.

[21] T. Roelleke and J. Wang, "Tf-idf uncovered: a study of theories and probabilities," in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, 2008, pp. 435–442.

[22] Z. Yao, Y. Fu, B. Liu, W. Hu, and H. Xiong, "Representing urban functions through zone embedding with human mobility patterns." in *IJCAI*, 2018, pp. 3919–3925.

[23] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[25] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *International Conference on Learning Representations*, 2018.

[26] A. Van Ooyen and B. Nienhuis, "Improving the convergence of the back-propagation algorithm," *Neural networks*, vol. 5, no. 3, pp. 465–471, 1992.

[27] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '14, 2014, p. 701–710.

[28] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems 26*, 2013, pp. 3111–3119.

[29] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, May 7-9, 2015, Conference Track Proceedings*, 2015.

[30] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical association*, vol. 66, no. 336, pp. 846–850, 1971.

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

[32] A. Parate, M. Böhmer, D. Chu, D. Ganesan, and B. M. Marlin, "Practical prediction and prefetch for faster access to applications on mobile phones," in *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, 2013, pp. 275–284.

[33] X. Chen, Y. Wang, J. He, S. Pan, Y. Li, and P. Zhang, "Cap: Context-aware app usage prediction with heterogeneous graph embedding," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, no. 1, pp. 1–25, 2019.

[34] T. Xia, Y. Li, and J. Fei, "Deepapp: Predicting personalized smartphone app usage via context-aware multi-task learning," *ACM Transactions on Intelligent Systems and Technology*, vol. 12, no. 3, p. 58, 2020.

[35] A. Grover and J. Leskovec, "Node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16, 2016, p. 855–864.

**Tong Li** received the B.S. degree and M.S. degree in communication engineering from Hunan University, China, in 2014 and 2017. At present, he is a dual Ph.D. student at the Hong Kong University of Science and Technology and the University of Helsinki. His research interests include ubiquitous computing, data science, and especially with applications to spatiotemporal data mining. He is an IEEE student member.

**Zhaoqi Yang** is an undergraduate student in department of Electronic Engineering, Tsinghua University, Beijing, China. His research interests include data mining and graph mining.

**Yong Li** (M'09-SM'16) received the B.S. degree in electronics and information engineering from Huazhong University of Science and Technology, Wuhan, China, in 2007 and the Ph.D. degree in electronic engineering from Tsinghua University, Beijing, China, in 2012. He is currently a Faculty Member of the Department of Electronic Engineering, Tsinghua University.

Dr. Li has served as General Chair, TPC Chair, SPC/TPC Member for several international workshops and conferences, and he is on the editorial board of two IEEE journals. His papers have total citations more than 6900. Among them, ten are ESI Highly Cited Papers in Computer Science, and four receive conference Best Paper (run-up) Awards. He received IEEE 2016 ComSoc Asia-Pacific Outstanding Young Researchers, Young Talent Program of China Association for Science and Technology, and the National Youth Talent Support Program.

**Benjamin Finley** received the B.S. degree in software engineering from Milwaukee School of Engineering, Milwaukee, WI, USA, and the M.S. and D.S. degrees in telecommunication engineering from Aalto University, Helsinki, Finland. He is currently a postdoctoral researcher at the Department of Computer Science, University of Helsinki. His current research interests include big telecom data analysis and user quality of experience.

**Sasu Tarkoma** (SM'12) received the MSc and PhD degrees in computer science from the Department of Computer Science, University of Helsinki. He is a Professor of Computer Science at the University of Helsinki, and Head of the Department of Computer Science. He has authored 4 textbooks and has published over 200 scientific articles. His research interests are Internet technology, distributed systems, data analytics, and mobile and ubiquitous computing. He is Fellow of IET and EAI. He has nine granted US Patents. His research has received several Best Paper awards and mentions, for example at IEEE PerCom, IEEE ICDCS, ACM CCR, and ACM OSR.

**Pan Hui** (SM'14-F'18) received his Ph.D. degree from the Computer Laboratory at University of Cambridge, and both his Bachelor and MPhil degrees from the University of Hong Kong. He is the Nokia Chair Professor in Data Science and Professor of Computer Science at the University of Helsinki. He is also the director of the HKUST-DT Systems and Media Lab at the Hong Kong University of Science and Technology. He was a senior research scientist and then a Distinguished Scientist for Telekom Innovation Laboratories (T-labs) Germany and an adjunct Professor of social computing and networking at Aalto University. His industrial profile also includes his research at Intel Research Cambridge and Thomson Research Paris. He has published more than 300 research papers and with over 17,500 citations. He has 30 granted and filed European and US patents in the areas of augmented reality, data science, and mobile computing. He has been serving on the organising and technical program committee of numerous top international conferences including ACM SIGCOMM, MobiSys, IEEE Infocom, ICNP, SECON, IJCAI, AAAI, ICWSM and WWW. He is an associate editor for the leading journals IEEE Transactions on Mobile Computing. He is an IEEE Fellow, an ACM Distinguished Scientist, and a member of the Academia Europaea.

TIETOJENKÄSITTELYTIETEEN OSASTO
PL 68 (Pietari Kalmin katu 5)
00014 Helsingin yliopisto

DEPARTMENT OF COMPUTER SCIENCE
P.O. Box 68 (Pietari Kalmin katu 5)
FI-00014 University of Helsinki, FINLAND

JULKAISUSARJA **A**

SERIES OF PUBLICATIONS **A**

Reports are available on the e-thesis site of the University of Helsinki.

A-2018-1 M. Nelimarkka: Performative Hybrid Interaction: Understanding Planned Events across Collocated and Mediated Interaction Spheres. 64+82 pp. (Ph.D. Thesis)

A-2018-2 E. Peltonen: Crowdsensed Mobile Data Analytics. 100+91 pp. (Ph.D. Thesis)

A-2018-3 O. Barral: Implicit Interaction with Textual Information using Physiological Signals. 72+145 pp. (Ph.D. Thesis)

A-2018-4 I. Kosunen: Exploring the Dynamics of the Biocybernetic Loop in Physiological Computing. 91+161 pp. (Ph.D. Thesis)

A-2018-5 J. Berg: Solving Optimization Problems via Maximum Satisfiability: Encodings and Re-Encodings. 86+102 pp. (Ph.D. Thesis)

A-2018-6 J. Pyykkö: Online Personalization in Exploratory Search. 101+63 pp. (Ph.D. Thesis)

A-2018-7 L. Pivovarova: Classification and Clustering in Media Monitoring: from Knowledge Engineering to Deep Learning. 78+56 pp. (Ph.D. Thesis)

A-2019-1 K. Salo: Modular Audio Platform for Youth Engagement in a Museum Context. 97+78 pp. (Ph.D. Thesis)

A-2019-2 A. Koski: On the Provisioning of Mission Critical Information Systems based on Public Tenders. 96+79 pp. (Ph.D. Thesis)

A-2019-3 A. Kantosalo: Human-Computer Co-Creativity - Designing, Evaluating and Modelling Computational Collaborators for Poetry Writing. 74+86 pp. (Ph.D. Thesis)

A-2019-4 O. Karkulahti: Understanding Social Media through Large Volume Measurements. 116 pp. (Ph.D. Thesis)

A-2019-5 S. Yaman: Initiating the Transition towards Continuous Experimentation: Empirical Studies with Software Development Teams and Practitioners. 81+90 pp. (Ph.D. Thesis)

A-2019-6 N. Mohan: Edge Computing Platforms and Protocols. 87+69 pp. (Ph.D. Thesis)

A-2019-7 I. Järvinen: Congestion Control and Active Queue Management During Flow Startup. 87+48 pp. (Ph.D. Thesis)

A-2019-8 J. Leinonen: Keystroke Data in Programming Courses. 56+53 pp. (Ph.D. Thesis)

A-2019-9 T. Talvitie: Counting and Sampling Directed Acyclic Graphs for Learning Bayesian Networks. 70+54 pp. (Ph.D. Thesis)

A-2019-10 J. Toivonen: Modeling and Learning Monomeric and Dimeric Transcription Factor Binding Motifs. 61+109 pp. (Ph.D. Thesis)

A-2019-11 S. Hemminki: Advances in Motion Sensing on Mobile Devices. 113+89 pp. (Ph.D. Thesis)

A-2019-12 P. Saikko: Implicit Hitting Set Algorithms for Constraint Optimization. 70+54 pp. (Ph.D. Thesis)

A-2020-1 J. Leppä-aho: Methods for Learning Directed and Undirected Graphical Models. 50+84 pp. (Ph.D. Thesis)

A-2020-2 P. Zhou: Edge-Facilitated Mobile Computing and Communication. 137 pp. (Ph.D. Thesis)

A-2020-3 J. N. Alanko: Space-Efficient Algorithms for Strings and Prefix-Sortable Graphs. 67+ 82 pp. (Ph.D. Thesis)
A-2020-4 H. Mäenpää: Organizing and Managing Contributor Involvement in Hybrid Open Source Software Development Communities. 78+67 pp. (Ph.D. Thesis)
A-2020-5 H. Laamanen: Epistemological Approach to Dependability of Intelligent Distributed Systems. 204+112 pp. (Ph.D. Thesis)
A-2020-6 T. Pulkkinen: Supporting the WLAN Positioning Lifecycle. 113+73 pp. (Ph.D. Thesis)
A-2020-7 O. Waltari: Privacy-Aware Opportunistic Wi-Fi. 51+44 pp. (Ph.D. Thesis)
A-2020-8 A. Niskanen: Computational Approaches to Dynamics and Uncertainty in Abstract Argumentation. 100+144 pp. (Ph.D. Thesis)
A-2020-9 M. Pozza: Enabling Network Flexibility by Decomposing Network Functions. 85+75 pp. (Ph.D. Thesis)
A-2020-10 A. Zavodovski: Open Infrastructure for Edge Computing. 77+58 pp. (Ph.D. Thesis)
A-2020-11 E. Khoramshahi: Multi-Projective Camera-Calibration, Modeling, and Integration in Mobile-Mapping Systems. 85+107 pp. (Ph.D. Thesis)
A-2021-1 J. Sakaya: From Approximations to Decisions. 115+54 pp. (Ph.D. Thesis)
A-2021-2 P. Xu: Efficient Approximate String Matching with Synonyms and Taxonomies. 62+ 64 pp. (Ph.D. Thesis)
A-2021-3 M. Khan: Privacy of User Identities in Cellular Networks. 112+88 pp. (Ph.D. Thesis)
A-2021-4 K. Alnajjar: Computational Understanding, Generation and Evaluation of Creative Expressions. 56+91 pp. (Ph.D. Thesis)
A-2021-5 C. Zhang: Performance Benchmarking and Query Optimization for Multi-Model Databases. 68+90 pp. (Ph.D. Thesis)
A-2021-6 Y. Chen: Performance Tuning and Query Optimization for Big Data Management. 64+120 pp. (Ph.D. Thesis)
A-2021-7 K. Rantanen: Optimization Algorithms for Learning Graphical Model Structures. 68+76 pp. (Ph.D. Thesis)
A-2021-8 A. I. Maarala: Scalable computational methods for high-throughput sequencing data analytics in population genomics. 98+61 pp. (Ph.D. Thesis)
A-2022-1 C. He: Entity-Based Insight Discovery in Visual Data Exploration. 62+63 pp. (Ph.D. Thesis)
A-2022-2 T. Vuong: Behavioral Task Modeling for Entity Recommendation. 85+171 pp. (Ph.D. Thesis)
A-2022-3 S. Linkola: Creative Systems, Agents and Societies: Theoretical Analysis Tools and Empirical Collaboration Studies. 71+58 pp. (Ph.D. Thesis)
A-2022-4 G. Yuan: Keyword Searches and Schema Transformation for Multi-Model Databases. 76+96 pp. (Ph.D. Thesis)