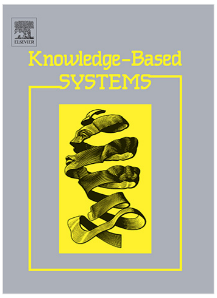
Journal Pre-proof



Usage profiling from mobile applications: A case study of online activity for Australian primary schools

Jie Yang, Jun Ma, Sarah K. Howard

PII: S0950-7051(19)30544-1
DOI: https://doi.org/10.1016/j.knosys.2019.105214
Reference: KNOSYS 105214

To appear in: *Knowledge-Based Systems*

Received date : 23 June 2019
Revised date : 6 November 2019
Accepted date : 7 November 2019

Please cite this article as: J. Yang, J. Ma and S.K. Howard, Usage profiling from mobile applications: A case study of online activity for Australian primary schools, *Knowledge-Based Systems* (2019), doi: https://doi.org/10.1016/j.knosys.2019.105214.

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2019 Published by Elsevier B.V.

Journal Pre-proof

Usage profiling from mobile applications: a case study of online activity for Australian primary schools

Jie Yang[a,*], Jun Ma[b,*], Sarah K. Howard[c]

[a]School of Computing and Information Technology, Faculty of Engineering and Information Sciences, University of Wollongong, Wollongong, NSW 2522, Australia
[b]Operations Delivery Division, Sydney Trains,
Alexandria, NSW, 2015, Australia
[c]School of Education, Faculty of Social Science,
University of Wollongong, Wollongong, NSW 2522, Australia

Abstract

Last decade has witnessed a drastically increasing development of smart devices, while related mobile applications have emerged significantly in people's daily life. As such, understanding the pattern of mobile application usage and related online behavior is of great importance for a variety of purposes, such as application engineering, resource optimization, and marketing. Existing research of online usage discovery includes surveys from end-users, application provider-related analysis, and usage log mining. These works, however, suffer from some limitations, such as lacking of user socio-economics background, insufficient coverage and sample bias, etc.

A novel and comprehensive application-usage profiling algorithm, termed as TAG, is proposed in this study to investigate online behavior. The proposed algorithm consists of three major steps: (i) T-step: representing usage data as a Term Frequency-Inverse Document Frequency based matrix; (ii) A-step: applying Alternating Least Squares factorization technique to reduce

[*] Corresponding author.
Email addresses: jiey@uow.edu.au (Jie Yang), jun.ma@transport.nsw.gov.au (Jun Ma), sahoward@uow.edu.au (Sarah K. Howard)

Journal Pre-proof

data sparseness and dimension; and last (iii) G-step: utilizing a smoothed Gaussian Mixture Model for clustering purpose.

The performance of the proposed TAG algorithm is evaluated, taking a national dataset generated from 31,280 devices and 30,155 applications over 30 months as an example. Experimental results demonstrate that the proposed algorithm outperforms existing methods via forming accurate usage groups from school-level online behavior. As such, the superior clustering outcome demonstrates the flexibility and applicability of the proposed work for understanding online pattern using complex application usage data. Resultant knowledge can in turn be used to inform decision making and improve application development.

Keywords: User online behavior, Mobile applications, Alternating Least Squares, Smooth Gaussian Mixture Model

1. Introduction

With the rapid growth of internet infrastructure and hardware development, billions of people worldwide are using smart devices (e.g. smart phones, smart watches and tablets) in their daily life. The prevalence of smart devices has accordingly promoted the popularity of mobile applications (a.k.a. apps) for different usage purposes, such as entertainment, social networking, E-commerce, and to name a few. In return, the industry stakeholders are interested in analyzing customers' app-usage data to formulate better marketing and/or development strategy. Some key questions that they want to address are the following: "who will be the potential users of apps?", "what is the usage pattern with regarding to one specific app, and any relationship with their demographic profile?", or "is there any significantly different usage pattern across various categories of apps?".

Journal Pre-proof

Journal Pre-proof

Such a study has already drawn great attention in many use cases, that drastically motivates manufacturers, marketing managers and advertisers, and application developers. A sufficient number of research efforts, therefore, have been made to discover online-usage pattern. Existing work can then be categorized into three mainstream aspects: surveys from end-users, apps provider (store)-related analysis, and online activity (log) mining. The major limitations, however, with existing work are:

- Time sensitivity: the majority of existing research is proposed within a limit time frame. The app-store related analysis from [1], for instance, is based on a five-month-duration data. As a result, their findings may not be sufficiently comprehensive to capture user behavior, and not be promisingly generative to a longer time frame and larger scale of apps and user population;
- Insufficient coverage: some research only focuses on a few specific types of apps, specific geographical regions, or survey feedback from a small group of participants. Such a limitation may introduce some biases to their findings.
- Lack of user socio-economics: very few existing work is able to match customers' usage pattern explicitly with their socio-economic information (i.e., age, residence, and social class). However, customers' socio-economic is extremely critical to investigate user preference, and understand the reason behind their online behavior patterns.

Towards this end, there is an increasing need to develop alternative frameworks for facilitating app usage analysis and to overcome the aforementioned three limitations. In this study, we are fortunate to access to an

Journal Pre-proof

Australia-wide dataset of app usage by involving an "one-to-one device program (OODP)". OODP is a non-profit program, supported by the Australian government (2012-2017), aiming to improve digital inclusion in education for young students, particularly those in remote areas. In this program, participants come from numerous primary schools (from Year 1 to Year 6) across the entire Australia. Some characteristics of harvested usage data from OODP are summarized as follows: (i) online usage from over 31,280 smart devices and 30,155 apps; (ii) more than 190 primary schools from both metropolitan and provincial areas; (iii) a total of 17,623,046 usage records collected between the period of January of 2016 to June of 2018.

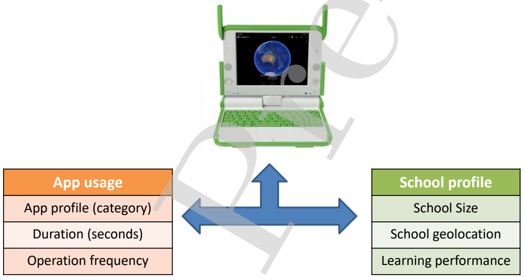


Figure 1: Dimensions for analyzing diverse behavioral patterns on app usage. The manufactured tablet machine is also illustrated.

Based on this dataset, a systematic empirical study is proposed, and the dimensions of interest (including app usage and school profile) are shown in Fig. 1. The hardware device (i.e. the tablet machine) is also illustrated. Towards this end, this article makes the following main contributions:

- We maintain one of the largest app-usage datasets with the national scale. That is the data produced by more than 31 thousands tablets since the year of 2016 to the mid 2018 (around 30-month duration). The entire data size is approximately 1 Terabyte. Harvested data re-

Journal Pre-proof

Journal Pre-proof

sources include app ID, app categories, timestamps, duration, device Id, etc. This extensive dataset will not only assist with the understanding of diverse online behavior, but also provide a valuable resource for any other big-data analytical techniques;

- To identify online pattern and usage trends, a novel profiling algorithm is proposed, which aims to cluster schools based on their usage pattern. This proposed algorithm consists of three major steps: (i) T-step: to establish a Term Frequency-Inverse Document Frequency based matrix to represent usage behavior; (ii) A-step: to apply Alternating Least Squares based factorization technique to reduce data sparseness and size (dimension); and (iii) G-step: to employ a smoothed Gaussian Mixture Model algorithm to form correct groups based on app usage;
- From resultant groups/clusters, we further systematically conduct within-cluster analysis. We identify usage dynamics cross different clusters, and discover the relationship between school online preference and offline performance. The findings will help in formulating better development strategy for online resource and/or regularizing app consumption.

The remainder of the paper is organized as follows. Section 2 provides a review of literature in apps usage analysis. Section 3 summarizes the study background, and Section 4 presents the detail of the usage-profiling framework, including data cleaning, feature generation, and clustering. Section 5 provides apps-usage analysis results, compared to state-of-the-art approaches. Section 6 offers a discussion of the within-cluster analysis, in terms of usage pattern and school profile, followed by concluding remarks

Journal Pre-proof

Journal Pre-proof

in Section 7.

2. Related work

In recent decades, there has been a dramatic growth of smart-device appliances. For instance, according to [2], approximately 20 billion connected digital devices are expected worldwide by 2020; furthermore, another recent report [3] indicates that the number of smart devices will rise at a compound annual growth rate of 11 percents between 2017 and 2023. As a result, massive volumes of user online activity are generated on the daily basis. A huge number of research work, therefore, has been proposed for analyzing user online activity, including online recommendation, web semantic analysis, etc. In this paper, we will focus on mobile app usage behavior. Existing research in understanding app usage pattern can be broadly categorized as follows: end-user-oriented survey, analysis using app-store profile, and online activity (log) mining.

2.1. End-user-oriented survey

There exists quite a number of studies on developing surveys and/or interviews to understand app usage from end users. Generally speaking, the survey process is carried out by identifying the sampling population, collecting participants' responses, and eventually analyzing responses statistically to formulate observations into meaningful knowledge.

To involve end users in mobile apps design, a self-documenting survey tool is developed in [4]. Three elicitation steps are employed to capture participant's feedback in terms of (i) contextual information (such as voice, picture and natural language text); (ii) individual needs (using text-based descriptions or audio recording); and (iii) rationale and task by describing

Journal Pre-proof

why a requirement is important to users. It is reported that this survey tool for requirements elicitation supports end-users in documenting their individual needs. Additionally, the findings also showed that requirements analysts are able to transcribe collected end-user needs into well-defined requirement descriptions without further involvement. As a proof of concept, this mobile survey tool demonstrates its applicability of informing the app developers, and delivering tailored and customized products to meet individual needs.

Another digital survey is present in [5] to explore online medical app usage. Nowadays, the medical community is also increasingly inclined to employ mobile technology to assist with clinical decision making. In this situation, a nation-wide email survey is carried out to exam the relationship between users' background and their usage preference in [5]. Four aspects of individual profile are collected via an online survey, including: specialty, level of training, use of smartphones, and favorite apps list. Later, a set of Chi-squared tests were employed to discover the relationship among these four aspects. Their results indicated that the most popular type of medical-related app is drug guides, followed by medical calculators, coding and billing apps, and pregnancy wheels. On the other hand, despite of the variety of available apps, apps with reference materials, treatment instructions and general medical knowledge are the most in need.

The survey work reported in [6] is to evaluate the applicability of using mobile app to support trainee doctors' workplace learning and patient care. In particular, trainees' access behavior to the electronic library was valued. Two important findings were revealed by questionnaire results: (i) online medical-related library contributed to the training process, and accordingly improved the patient care by providing accurate prescribing and treatment planning; (ii) smartphones were predominately used by males, which sug-

Journal Pre-proof

Journal Pre-proof

gests that the design of mobile technology should be sensitive to the gender factor.

Additionally, to gain a better understanding of development practices for mobile apps in general, a large-scale online survey is reported in [7]. Participants from 15 countries are invited worldwide to participate in an investigation of end-user's behavior, including app usage, individual requirements, the rationale for app adoption. Analytical results suggest numerous driven factors behind user's app adoption, such as price, app type, and cultural background, etc. More importantly, the differences in user behaviors across countries are also carefully addressed.

Overall, some typical research on discovering app usage pattern based on the survey result is summarized in Table 1.

Table 1: Existing survey research on identifying patterns for online app usage.

Ref	No. of Participants	Duration	Applied techniques
[4]	9	1.9 days	Descriptive statistics
[5]	3306	2 months	Chi-squared tests
[6]	570	13 months	Chi-squared tests
[7]	10208	2 months	Chi-squared tests, Linear regression

One major advantage of using the survey methodology is the high certainty about the participants' profile (such as their gender and age), as those information can be included in the questionnaire. However, there are also some limitations of adopting survey for analyzing online activity. One common problem comes from the selection bias while choosing the representative participants. A priori knowledge may vary among survey organizers, thereby resulting in focusing on different aspects or sampling populations. Similarly,

Journal Pre-proof

the designed questionnaire is also affected by organizers, which may lead to significantly different (or even conflicting) survey results. Meanwhile, the analytical method from existing survey work is relatively simple as the majority is more descriptive and focuses on the statistical characteristics, and correlation among multiple-variables could be always neglected. More importantly, the survey finding is also limited by a relatively smaller size of participants (usually around few-thousand people and apps) and short time window (on average 2 months). Consequently, given some characteristics of mobile app (such as the large amount and active evolution), more advanced technique is in need of discovering users' online activity.

2.2. App-store profile

Apps are available via an application distribution platform, commonly known as the app stores. These online stores provide users with services to manage their apps, such as searching, browsing, downloading, and updating. Furthermore, app stores also enable a communication channel between app developers and end users through their reviewing and rating system.

App store analysis, accordingly, studies the information obtained from app stores to discover patterns and/or trends of online activity. Recent studies have made advances in mining app store data, including opinion mining, review spam detection, and more general mining applications.

For example, to identify recurrent app topics, Valulenko et al. applied the Latent Dirichlet Allocation (LDA) algorithm on a set of 600,000 English-language app descriptions, collected from the Apple App Store [8]. This topic-modeling result is then used to structure and manage app categories. A clustering-based work is proposed in [9] by mining 120 apps from Google Play. More precisely, the resultant description-based cluster is compared

Journal Pre-proof

Journal Pre-proof

with category-based ones, while the former provided better features for app recommendation. Furthermore, Khalid et al. collected and classified 6,390 negative reviews from Apple App Store, and summarized the most frequent causes of usage complaints [10]. Their result indicated that functional errors, feature requests, and app crashes are the most frequent complaints, while users are most dissatisfied with issues of privacy invasion, as well as the hidden cost. As such, those findings have a direct and actionable impact on the app engineering, and have led to more sophisticate app development.

More recently, based on an unique dataset collected from a leading Android app-store service provider, a systematic empirical study is reported in [1]. In total, they studied usage data covering 17 million users and 0.28 million apps for a period of five month. Their contribution focuses on characterizing the diverse app-usage behavior to explore the correlation among app popularity, app management, network usage, and choice of device models. Some helpful implications are revealed, such as improving the workload and app ranking system in stores, to provide better understanding of user requirements and needs.

In addition, a systematic literature review on mining store data is reported in [11]. By surveying 34 articles, this review attempts to address three research questions: (i) available app review mining techniques; (ii) supporting software tools; and (iii) the evaluation of the maturity of existing techniques and tools. Another survey of literature that investigates app store analysis is reported in [12]. Both technical and non-technical attributes are extracted. As a result, there are a total of 7 key components investigated, including: API, Feature, Release, Review, Security, Store ecosystem, and Size and effort prediction.

Journal Pre-proof

Journal Pre-proof

2.3. Log mining

Along with survey and app-store analysis, usage log mining has also gained a great number of research attention because of the applicability and flexibility of handling huge amount of usage data. Some of our preliminary work about online user behavior can be found in [13, 14].

A joint characterization framework for analyzing the spatial and temporal dynamics of apps usage is proposed in [15]. Using data collected from tier-1 cellular operators, the K-means clustering algorithm is firstly employed to group communication cells (base stations) using their apps usage. Later a statistical analysis is conducted within each group to explore their individual pattern in terms of the spatial and temporal dynamics. Their results provide an actionable suggestion to tune network parameter settings, thereby improving the network performance.

In [16], a user-profile mining approach is proposed. More precisely, the information gain is employed to measure the importance of individual apps towards users' attributes (such as their phone price and travel status). As such, the profile mining is reformulated as a binary classification problem, and later the Support Vector Machine technique is employed as the classifier. One limitation of the work [16] is that they are only concerned with the adoption/installation of an app, without considering the actual usage duration from apps.

Another work of profiling users is present in [17]. In particular, the app-log data contains 398,764 records from 5,906 users, and the main purpose of that study is to identify user groups of scalpers in mobile healthcare services. They first propose three similarity measurements for "Activity", "Location", and "Sequence", respectively. Additionally, based on pre-defined similarity calculation, the Agglomerative Hierarchical Clustering (AHC) and Divisive

Journal Pre-proof

Journal Pre-proof

Hierarchical Clustering (DHC) algorithms are employed to group log records and users into different clusters, respectively. As such, the representative record from the center user within each cluster is formulated as the cluster profile. Abnormal users and/or records are accordingly filtered as the scalper detection outcome.

2.4. Summary

In this section, we conduct the literature review on existing work towards discovering app-usage behavior. However, there are still open research questions remaining. For instance, some of the work is conducted within a short-term duration (i.e., on average a few months). Relevant findings (from limited observation) may not be able to fully discover behavior pattern, and not be well generative to a real scenario with large scale. Additionally, few work is done to match online pattern with offline behavior. There is, certainly, a research opportunity to identify the relationship between online usage patterns and offline preference, as well as the reason behind.

3. Preliminaries

In this study, app usage data was collected with the partnership of a leading industry partner, One Education, in Australia. One Education was a non-profit company, which was associated with One Laptop Per Child International (OLPC). The aim of the OLPC program was to bring the power of digital technology to young students those most in need. This is similar to many one-to-one device programs (OODPs) happening internationally.

3.1. One Education participation

One Education provided low-cost purpose-built Android tablet devices to schools across Australia. Since 2016, a 2-in-1 device (termed XO) has

Journal Pre-proof

been developed, in that it had the functionality of a tablet and laptop (e.g. touch screen and keyboard). On one hand, those XO devices primarily ran on a custom-built object-oriented Android operating system. On the other hand, they are designed to be low-cost, durable, to withstand heat and dust, and to be used by younger children aged 4 to 12. Furthermore, One Education also included a range of educational and productivity apps on the XO devices, such as web browsers and editing tools. Additional apps could either be downloaded through the One Education online store, Google Play or Android APK sites.

Up to June of 2018, the program included a total of 258 distinct schools voluntarily participating in this program. As shown in Fig. 2, the majority of schools come from the state of New South Wales (NSW), followed by Queensland (QLD), while less come from other states and territories. Among schools, over 93% of them are government-supported while the rest is formed by private organizations. Additionally, the school size (in terms of the number of total enrolled students) ranges from 100 up to 500 approximately. This indicates a wide coverage of participants, which could provide a comprehensive insight of online behavior (school students) in reality.

3.2. Data sources and collection

Usage-related data has been collected in two categories: behavior data and metadata. Behavior data consists of the basic usage records from real devices. When an app was launched, a secondary computer agent would record the current timestamp and track how long the app was running. The resulting data included: Device Id, App Id, Starting time, Duration of use. The collected data is then locally stored into the devices, and periodically synchronized to servers later. Additionally, in terms of collected

Journal Pre-proof

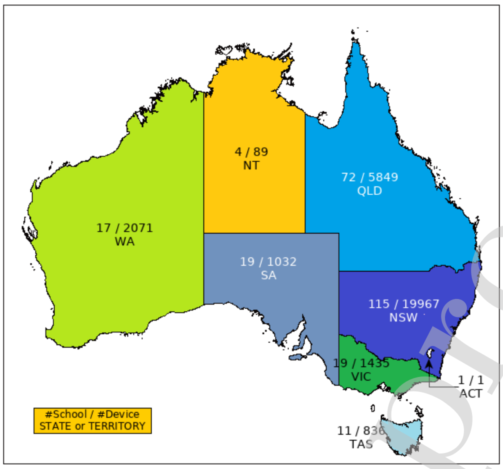


Figure 2: Schools and devices distribution among Australia states and territories.

usage data, three key steps are conducted to preserve the research ethics and user privacy. Firstly, data collection (and any subsequent analysis) is completely governed to ensure compliance with privacy policy in the Term-of-Use statements upon purchasing the device. In other words, schools were aware that anonymous app usage data could be collected. Secondly, usage data is transferred with encryption mechanism, while being assured of minimal risks of privacy leakage. At last, the anonymization is performed so that no individual user can be identified.

Eventually, this behavior dataset covers online app usage from 2016 to 2018 (approximately a 30-month period). In that time, over 17,623,046 streaming activity records have been accumulated from a total of 31,280 devices of 258 schools until June of 2018. On a daily basis, this results in an average of 12,415 activity records generated and reported back to our big-data pre-processing platform (which will be described in Section 3.3).

The second data type was metadata associated with school and apps

Journal Pre-proof

Journal Pre-proof

profiles. In terms of school profiles, we target on the school-level information, as the individual (student) profile is inaccessible. In particular, we are interested in overall school learning performance and related school profile. Towards this end, publicly-available school data (such as their Australian National Assessment Program Literacy and Numeracy (NAPLAN) score, geolocation, and total number of enrolled students) was manually collected from an online official resource [1]. As for the mobile apps, the publicly-available app information is collected from Google Play, including App Id, Developer profile, Category, Description, etc.

In summary, attributes and relationships from collected data are represented in Fig. 3. Note that herein individual's app usage data was aggregated at the school level. This approach maintains anonymity of the student but allows for differences in usage to be observed between schools.

Additionally, the overall statistics of Top-10 categories (based on total usage duration) is summarized in Table 2, which contains aggregated information related to number of apps, distinct schools (if schools ever used), and total duration (in second) over the 30-month period. Aggregated results from Table 2 indicated that the most popular category of app by schools is "Communication", followed by "Education", "Lifestyle", and "Tools". Obviously, school students are much more interested in using the devices for sharing or exchanging information with peers, as "Communication" apps are dominant among all categories. Surprisingly, the number of "Communication" related apps is the least (140), compared to others, such as "Education" (2137), "Lifestyle" (4261), and "Tools" (857). The results clearly imply the "Communication" market is dominated by only a few apps, while markets

[1] https://www.myschool.edu.au/

Journal Pre-proof

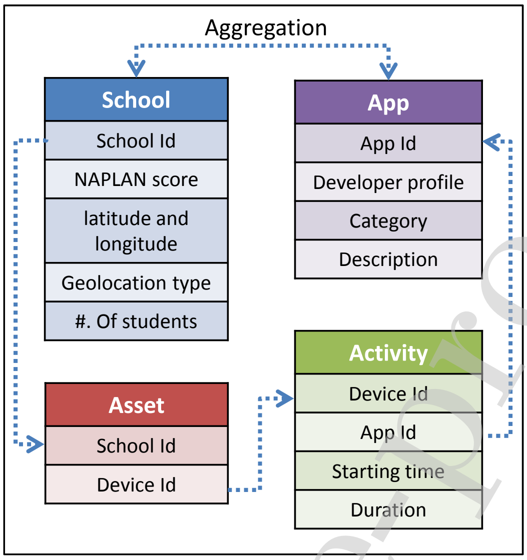


Figure 3: Collected data attributes and their relationship.

Table 2: Top-10 app categories, based on the total duration.

Category	Apps	Schools	Duration(s)
Communication	140	256	7853861417
Education	2137	251	3578378335
Lifestyle	4261	249	1786892686
Tools	857	247	1486873387
Arcade	1727	170	772491259
Action	3612	172	716283827
Productivity	291	248	680518953
Puzzle	1163	231	482293465
Games	2949	171	225694045
Education-Creativity	127	198	221631404

Journal Pre-proof

for other types of apps are still very competitive. As such, more options (apps) available to be chosen from. At last, we also notice that some app categories are not as popular as others in some schools. For instance, there are approximately 87 out of 258 schools never installing either “Arcade” or “Games” apps. That could be the result of school policy and/or unawareness. Therefore, some recommendation can be made here to utilize these online resources. We leave this recommendation study for our future work.

3.3. Big-data pre-processing platform

To cope with the huge amount of harvested data, a platform with a high storage capacity and computing power is essential. As a result, open-source big-data technologies are implemented for the scalable preprocessing purpose (mainly data storage and aggregation), and its architecture is shown in Fig 4.

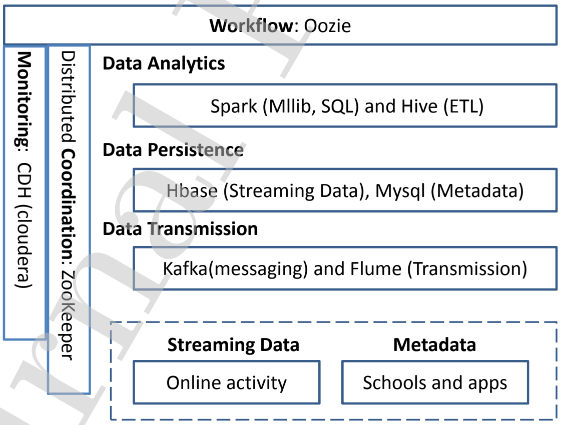

Workflow: Oozie
Monitoring: CDH (cloudera)
Distributed Coordination: ZooKeeper
Data Analytics
Spark (Mllib, SQL) and Hive (ETL)
Data Persistence
Hbase (Streaming Data), Mysql (Metadata)
Data Transmission
Kafka(messaging) and Flume (Transmission)
Streaming Data
Online activity
Metadata
Schools and apps

Figure 4: Architecture of the implemented big-data pre-processing platform for usage behavioral analysis.

As observed, the developed platform is established on some state-of-

Journal Pre-proof

Journal Pre-proof

the-art components, such as Kafka [2], Flume [3], HBase [4], and Spark [5], etc. According to the data flow, online usage data will first reach the Kafka component, that is a distributed messaging broker that guarantees fault tolerance during the data streaming. Later, Flume is employed to transfer data from Kafka to high-level components for data persistence. Due to the diversity of data format and volume, two persistence/storage mechanisms are implemented, i.e., HBase and MySQL [6]. In our study, the MySQL is used to store the metadata for both schools and apps due to the relatively smaller volume (approximately 35,000 records). By contrast, the massive activity data is hosted using HBase due to the good scalability and fault tolerance capability.

Next, we use Hive [7] and Spark to process the massive data. Hive is a Big Data extraction, transformation and load (ETL) tool, and Spark provides a scalable framework for in-memory computing to support fast data analytic. In addition, to monitor the whole platform, we also have developed a management module using ZooKeeper [8] to maintain configuration information, and to provide distributed synchronization or group services. We also consider to use Oozie [9] as the scheduling system to manage computing jobs.

Our presented big-data platform is then implemented using a virtual-cloud infrastructure with 14 computing nodes. Each of them comes with

[2] https://kafka.apache.org/

[3] https://flume.apache.org/

[4] https://hbase.apache.org/

[5] https://spark.apache.org/

[6] https://www.mysql.com/

[7] https://hive.apache.org/

[8] https://zookeeper.apache.org/

[9] https://oozie.apache.org/

Journal Pre-proof

two Intel-Xeon 1.8GHz cores and 10G memory. In addition, one computing node is set up as the master machine for both Hadoop, Spark, and Hbase, while the rest are used as the slaver nodes. Note that the big-data platform is explicitly implemented for data preprocessing, such as receiving streaming usage records, and data storage and aggregation. With this end in view, the following section describes an innovative profile-clustering algorithm that is designed to facilitate the understanding of app usage pattern.

4. Proposed app usage analysis

In this section, we explain the details of our analysis using aggregated behavioral data at school level, in order to discover online characteristic of app usage. We start by extracting the usage dynamics of aggregated data, and then propose a novel clustering algorithm to group schools with similar usage pattern, and finally introduce three evaluation metrics to verify the clustering outcome.

4.1. TF-IDF based feature generation

Note that students' usage data is anonymous, that implies the difficulty of matching or tracking behavior data with an individual student. Consequently, we aggregate students' data to the school level (again, using the aforementioned big-data platform). That is, online activity records are to be merged as long as students come from the same school. Mathematically, let $S = \{\mathrm{s}_1, \mathrm{s}_2, ..., \mathrm{s}_{|S|}\}$ be the set of all schools, and s_i represents the i-th school. Let $A = \left\{\mathrm{app}_1, \mathrm{app}_2, ..., \mathrm{app}_{|A|}\right\}$ be a set of all $|A|$ distinct apps used by $|S|$ schools. As such, one simple way of representing school-level feature is to sum up the total duration of individual app utilized by each school. That is,

Journal Pre-proof

Journal Pre-proof

let a_j represent the running duration (per use) for the j-th app, and f_i denote a feature vector for the i-th school, then we can have $f_i^j = count(\text{app}_j, \text{s}_i)$ $(\forall i \in \{1, \ldots, |S|\}, \forall j \in \{1, \ldots, |A|\})$, where $count(\text{app}_j, \text{s}_i)$ represents the total duration of the j-th app ($\text{app}_j \in A$) from all students in the school s_i.

Nevertheless, one major problem with total duration (or app consumption) based feature is that there might be some apps with less frequency of use but more appealing or attractive to students; while some apps are of less importance but with higher frequency of use, such as message notification, internet access, or other administration operation. To compensate the drawback of total consumption of app occurrence, and emphasize different contribution of an app to a school/students, a Term Frequency-Inverse Document Frequency (TF-IDF) based feature is employed, which is a popular measurement for the textual content. In text mining, TF represents the number of times that a keyword occurs in the document, while IDF is the inverse document frequency of a term. The TF-IDF value is directly proportional to the appearing times for any keyword, but is offset by the frequency of the same keyword in the total corpus. In other words, keywords appear more frequently in general will receive smaller TF-IDF value; whereas a keyword with larger TF-IDF value reflects higher importance in given documents.

As a result, by adapting the TF-IDF concept to generate a high-level feature for school records, we are able to find out more meaningful apps rather than general ones. For any app_j, its TF value is computed as follows:

$$TF(\text{app}_j, \text{s}_i) = \frac{count(\text{app}_j, \text{s}_i)}{size(\text{s}_i)}, \tag{1}$$

where $size(\text{s}_i) = \sum_{j=1}^{|A|} count(\text{app}_j, \text{s}_i)$ represents the total duration for all apps

Journal Pre-proof

in s_i. Accordingly, the IDF value for app_j is shown in the following:

$$IDF(\mathrm{app}_j) = \log \frac{|S|}{sch(\mathrm{app}_j, S) + 1}, \tag{2}$$

where $|S|$ is the number of all schools, and $sch(\mathrm{app}_j, S)$ represents how many schools ever use the j-th app (app_j) before. Towards this end, the TF-IDF feature of a particular app app_j in the s_i school can be calculated as follows:

$$TF\text{-}IDF^{\mathrm{s}_i}_{\mathrm{app}_j} = TF(\mathrm{app}_j, \mathrm{s}_i) \times IDF(\mathrm{app}_j). \tag{3}$$

Additionally, the normalized function $\|*\|$ is also introduced to smooth the TF-IDF value for each individual school:

$$\|TF\text{-}IDF^{\mathrm{s}_i}_{\mathrm{app}_j}\| = \frac{TF\text{-}IDF^{\mathrm{s}_i}_{\mathrm{app}_j}}{\max(TF\text{-}IDF^{\mathrm{s}_i}_{A})}, \forall \mathrm{app}_j \in A, \tag{4}$$

where $\max(*)$ represents the maximal value. In this context, the weighted TF-IDF value for any single app is computed using a normalized consumption against the most-frequently used app from the same school.

As such, we can now establish an app-utilization matrix $P \in \mathbb{R}^{|S|\times|A|}$, from which the element $p_{i,j}$ from the i-th row and j-th column represents the usage of the j-th app within the i-th school, i.e., the weighted TF-IDF value:

$$p_{i,j} = \|TF\text{-}IDF^{\mathrm{s}_i}_{\mathrm{app}_j}\| \quad (\forall \mathrm{s}_i \in S, \mathrm{app}_j \in A, p_{i,j} \in P). \tag{5}$$

4.2. ALS-based data cleaning

Intuitively, we could analyze online pattern by leveraging this consumption matrix $P \in \mathbb{R}^{|S|\times|A|}$ as shown in Eq. (5). However, P is problematically sparse as majority of its elements are zero because it is unpractical for one school to use all available apps. This can be further confirmed by the statistical result shown in Table 2. The sparseness of P would impose some

Journal Pre-proof

noise, thereby leading to incorrect or misleading findings. Toward this end, an Alternating Least Squares (ALS)-based matrix factorization technique is introduced to address the data sparseness problem.

Matrix factorization techniques have proven to be highly successful in representing high-dimensional and corrupted data, with missing values and/or outliers [18–20]. In particular, as one of factorization methods, the ALS approach is inspired by the recommendation system, which aims to recommend items that may be of interest to users. By analyzing users' historical behavior, the recommendation system predicts their preference that is missing from the current observation. In this study, each app with empty consumption values is considered as an item to be evaluated in a recommendation system. Therefore, the ALS technique is employed to factorize the consumption matrix $\boldsymbol{P}$, to capture the latent relationship between schools and apps, while generating less-noisy data. Mathematically, the ALS approach can be factorized as below:

$$\boldsymbol{P} \approx \boldsymbol{U} \times \boldsymbol{F}, \tag{6}$$

where $\boldsymbol{U} \in \mathbb{R}^{|S| \times r}$ is the school-profile matrix, $\boldsymbol{F} \in \mathbb{R}^{r \times |A|}$ is the app-feature matrix after factorization, and r is a pre-defined parameter for the number of latent factors ($r << \min(|S|, |A|)$). A row vector ($\mathrm{u}_i \in \mathbb{R}^r$) from $\boldsymbol{U}$; and one column vector ($\mathrm{f}_j \in \mathbb{R}^r$) from $\boldsymbol{F}$ represents the factorized profile for the i-th school and the j-th app, respectively. Specifically, the following objective function of regularized mean-squared error (RMSE) is minimized to compute both u_i and f_j:

$$f(U, F) = \sum_{i,j} (p_{i,j} - \mathrm{u}_i \mathrm{f}_j)^2 + \lambda \left(\sum_i ||\mathrm{u}_i||^2 + \sum_j ||\mathrm{f}_j||^2 \right), \tag{7}$$

where λ is the penalty parameter. Note that the optimization problem in Eq.

Journal Pre-proof

(7) is non-convex with respect to the variable matrices of U and F. It implies that we cannot solve them together directly; instead, we need to separate the entire problem into two independent sub-problems. In this case, separated sub-optimization problems become convex with respect to only one variable matrix while fixing another one. We can iteratively calculate the variable matrix one-by-one. To begin with, we first fix the F matrix (then it becomes a constant) and update U by computing the gradient for each row:

$$\begin{aligned}\frac{\partial f(U, F_{fixed})}{\partial \mathrm{u}_i} = 0 \Rightarrow & \frac{\partial \left(\sum_{i,j}(p_{i,j} - \mathrm{u}_i \mathrm{f}_j)^2 + \lambda \left(\sum_i \|\mathrm{u}_i\|^2\right)\right)}{\partial \mathrm{u}_i} = 0, \\ & \Rightarrow \sum_{i,j} \left(\mathrm{u}_i \mathrm{f}_j \mathrm{f}_j - p_{i,j} \mathrm{f}_j + \lambda \mathrm{u}_i^T\right) = 0 \quad (\forall i \in [1, |S|]),\end{aligned} \tag{8}$$

where f_j is the j-th column vector from F. Similarly, we then can fix U and update individual columns from F. The entire process is repeat until a stopping criterion is satisfied (such as the maximal iteration, or the objection function value is less than a user-defined threshold).

Now, it is worthy noting that the matrix U is much significant (than P) for three reasons: 1) U is not as sparse as P as there are no zero values in U, which is less-noisier compared to P; 2) U (with less columns) can be regarded as a projection of P in a lower-dimensional space, and 3) we can compute the similarity of any two schools using U, as the i-th row from U now represents the profile of the i-th school [21]. As such, the matrix U is leveraged as the input for the subsequent clustering process, and the similarity of the i-th and k-th schools can also be computed as $sim(\mathrm{u}_i, \mathrm{u}_k)$, where $sim(*)$ can be a similarity measurement function, such as Pearson Correlation, Cosine Coefficient, or Mutual Information.

Journal Pre-proof

Journal Pre-proof

4.3. Smoothed GMM-based clustering

In this section, we will introduce an improved clustering technique to group schools according to their app usage. By clustering schools with similar app usage pattern, we can explore (i) what important app categories comprises from different schools, and (ii) patterns of usage arising from multiple app categories.

Towards this end, we employ Gaussian Mixture Model-based clustering (GMM) to cluster schools based on usage distributions. Assume that we aim to form K groups, and the j-th cluster ($j \in \{1, \ldots, K\}$) is generated from a Gaussian distribution with the parameter Θ_j (i.e., $\Theta_j = (\mu_j, \Sigma_j)$, where μ_j and Σ_j represents the means and variance-covariance matrix of the j-th Gaussian distribution, respectively). As such, given the i-th school behavior u_i, its posterior probability of being drawn from the j-th cluster is computed as

$$Pr(\Theta_j|\mathrm{u}_i) = \frac{Pr(\mathrm{u}_i|\Theta_j) \times Pr(\Theta_j)}{Pr(\mathrm{u}_i)} = \frac{Pr(\mathrm{u}_i|\Theta_j) \times \alpha_j}{\sum_j Pr(\mathrm{u}_i|\Theta_j) \times \alpha_j}, \tag{9}$$

where $\alpha_j = Pr(\Theta_j)$ is the prior probability of the j-th cluster, and the $Pr(\mathrm{u}_i|\Theta_j)$ can be computed by a Gaussian density function of the j-th cluster. In the standard GMM, those parameters (α_j and Θ_j, $\forall j \in \{1, \ldots, K\}$) can be updated using the Expectation-Maximization (EM) algorithm at each iteration, until a pre-defined convergence criterion is met.

To enhance the clustering performance, we further propose an optimal method to smooth the obtained results from GMM. More precisely, by computing the similarity between the target i-th school and others (i.e., $sim(\mathrm{u}_i, \mathrm{u}_k)$, $\forall k \in \{1, \ldots, |S|\}, k \neq i$), we can sort and identify a set of similar schools, which satisfied:

$$\Xi_i = \left\{ k | \arg\max_k sim(\mathrm{u}_i, \mathrm{u}_k) \right\}, \tag{10}$$

Journal Pre-proof

where u_k represents the k-th row of the U matrix. At last, the smoothed posterior probability for the i-th school can be calculated as below:

$$Pr^*(\Theta_j|\mathrm{u}_i) = \beta Pr(\Theta_j|\mathrm{u}_i) + \frac{(1-\beta)}{N}\sum_{k\in\Xi_i} Pr(\Theta_j|\mathrm{u}_k), \tag{11}$$

where $N = |\Xi_i|$ is the number of similar schools, and β is a penalty term that used to control how much the smoothed posterior probability is influenced by other schools. Obviously, if $\beta = 1$, then Eq. (11) becomes the traditional GMM process. Hence, the proposed estimation is an extension of Eq. (9).

4.4. Evaluation metrics

Hereafter, we provide the evaluation metrics for the smoothed GMM algorithm, to check whether the clustering performance is sufficiently close to the ground-truth. Let $C^* = \left\{C_1^*, C_2^*, \cdots, C_m^*\right\}$ and $C = \{C_1, C_2, \cdots, C_n\}$ be the ground-truth clustering and actual clustering result, respectively, where C_i^* and C_j are the individual clusters from C^* and C. Accordingly, the general goal is then to apply some metrics to measure the similarity between C^* and C. Three measurements based on Best Match and Interaction Probability are introduced hereafter.

Best Match: The key is to search for the best match peer in C for individual member from C^*. Let $d(*, *)$ denote a user-defined distance function, given a member of $C_i^* \in C^*$, its representative in C is defined as follows:

$$C'_j = \arg\min_{C_j\in S} d(C_j, C_i^*). \tag{12}$$

Furthermore, the distance function is calculated using the Entropy that is give by:

$$d(C_j, C_i^*) = H(C_j|C_i^*) = H(C_j, C_i^*) - H(C_i^*), \tag{13}$$

Journal Pre-proof

Journal Pre-proof

where $H(*,*)$ and $H(*)$ represent the joint and marginal entropy, respectively. Overall, the final symmetric metric under the Best Match criteria is given by:

$$d_{BM}(C, C^*) = \sum_{i=1}^{m} \sum_{j=1}^{n} d(C_j, C_i^*). \tag{14}$$

Interaction Probability: In Interaction Probability theory, we consider within-cluster and between-cluster interaction for all cluster members to measure the similarity of two clusters. For any pair of elements k, l, their interaction probability with the clustering set C (or C^*) is defined as:

$$Pr_S(k,l) = \begin{cases} \frac{1}{\|C_{same}\|}, & if \quad k, l \in C_{same}, C_{same} \in C \\ \frac{1}{\|N\|}, & otherwise \end{cases} \tag{15}$$

where $N = \sum_{i=1}^{m} |C_i^*| = \sum_{j=1}^{n} |C_j|$ is the total number of elements. Furthermore, based on the interaction probability in Eq. (15), two more metrics are introduced by adopting different distances (i.e., l_2 and Kullback-Leibler (KL)), as denoted as $d_{l_2}(C, C^*)$ and $d_{KL}(C, C^*)$:

$$d_{l_2}(C, C^*) = \sqrt{\frac{1}{N^2} \sum_{(k,l)} (Pr_C(k,l) - Pr_{C^*}(k,l))^2} \tag{16}$$

$$\begin{aligned} d_{KL}(C, C^*) = -\frac{1}{N^2} \sum_{(k,l)} & Pr_C(k,l) \log(Pr_{C^*}(k,l)) \\ & + (1 - Pr_C(k,l)) \log(1 - Pr_{C^*}(k,l)) \end{aligned} \tag{17}$$

Note that the proposed similar criteria are conceptually related to those from [22], in which the similarity of two clustering sets with overlapping members is considered (i.e., $C_k \cap C_l \neq \emptyset$ $(C_k, C_l \in C)$). Our modified criteria can be regarded as a special case of [22], as $C_k \cap C_l = \emptyset$ for our case.

4.5. Summary

Overall, the proposed analytical work, termed TAG, is summarized in Algorithm 1. Firstly, TAG represents the raw usage data as the format of a TF-IDF-like matrix P, in which each element from P is calculated using aggregated usage duration at the school level. Secondly, the ALS factorization technique is employed to reduce the data sparseness and dimension, and capture the latent relationship between schools and apps. Thirdly, a smoothed GMM method is further introduced to optimize the clustering results.

Additionally, there are four key parameters, including number of latent feature r, penalty terms λ for avoiding the overfitting of the data, size of neighbor schools N, and penalty term β for smoothing the traditional GMM result. We will investigate their influence on the clustering performance in the following section.

5. Experimental results

In this section, we evaluate the performance of our proposed TAG clustering algorithm based on the collected dataset (mentioned in Section 3). The effect of key parameters is evaluated in Section 5.2. The comparison results between the proposed algorithm and other methods are then presented in Section 5.3.

5.1. Setup and configuration

For the analysis a subset data was selected, as there was considerable fluctuation in participation as a result of school attrition and enrollment from year to year. As such, in the following, our experimental analysis is based on a total of 191 schools that actively reported to our platform between the year period of 2016 to 2018, while the rest of 67 schools were removed due to less participation. This resulted in a total of 11,173,828 usage records streamed from 30,155 distinct apps. This subset forms a huge

Journal Pre-proof

```
Input  : Raw activity records from $|A|$ apps and $|S|$ schools, number
         of latent features $r$, number of clusters $K$, size of neighbor
         schools $N$, penalty terms $\lambda$ and $\beta$.
Output: $K$ groups of schools.
T-step : initialize the TF-IDF matrix $P \in \mathbb{R}^{|S|\times|A|}$, where
         $p_{i,j} = \|TF\text{-}IDF^{\mathrm{s}_i}_{\mathrm{a}_j}\|$ as defined in Eq. (4);
A-step : given $r$, apply ALS-based matrix factorization technique to
         get a small-sized $U$, so that: $ALS(P, \lambda) = U \times F$, and
         $U \in \mathbb{R}^{|S|\times r}$; for the $i$-th school, compute its $N$ neighbors by
         $\Xi_i = \{k| \arg_k \max sim(\mathrm{u}_i, \mathrm{u}_k)\}, \forall k \in \{1, \ldots, |S|\}, k \neq i$, and
         $|\Xi_i| = N$
G-step : given $\beta$, employ the improved GMM algorithm to computer
         the probability:
         $k = \arg\max_k \left(\beta Pr(\Theta_k|\mathrm{u}_i) + \frac{(1-\beta)}{N} \sum_{j\in\Xi_i} Pr(\Theta_k|\mathrm{u}_j)\right)$.
```

Algorithm 1: Proposed TAG algorithm for discovering app-usage pattern.

sparse matrix with size of 191×30,155, which is utilized as the input for the proposed TAG algorithm (see Algorithm 1).

Additionally, we also collected NAPLAN scores for selected schools during the study period, including scores on five areas: “Numeracy”, “Reading”, “Grammar”, “Spelling”, and “Writing”. These learning scores, together with other school profile data (such as school size and geographical information), will be then used to evaluate the performance of the proposed clustering algorithm.

Journal Pre-proof

5.2. Parameter validation

In this section, we analyze the robustness of the TAG algorithm to different input parameters, including number of latent features r when factorizing the TF-IDF matrix, size of neighbor schools N, and the penalty terms λ and β.

Number of latent features (r) and penalty terms λ. In ALS, we aim to factorize the consumption matrix P into $P \approx U \times F$ (as shown in Eq. (6)), while the parameter r controls the size of U (and F). Obviously, a relatively large r is more prone to converging to a minimum solution than a smaller r during the factorization. However, the larger r will also be computationally expensive. On the other hand, the penalty terms λ is used as the regularization term to avoid the overfitting the data (see Eq. (7)). As a result, we evaluate the TAG performance based on various r and λ values.

The value of λ is set within the range of $\lambda \in \{0.005, 0.01, 0.05, 0.1\}$, while r is evaluated among the range of $r \in$ {10, 50, 100, 150, 200, 250, 300, 500, 700, 1000}. For parameter tuning, we include all available data and aim to minimize the objection function of RMSE in Eq. (7), while fitting the actual app-usage data. For each selected pair of (λ, r), we run the ALS factorization for a maximal iteration of 50, and the results are demonstrated in Fig. 5.

One straightforward discovery is that ALS always converges to a local minima, that seems to be less impacted by the combination setting of (λ, r). In practice, ALS converges after about 15 iterations for all scenarios, and the average RMSE error reduces to a minima of 12.35. However, as shown in Fig. 5, various λ values lead to different final RMSE, and we normally need to try a small λ value to generated a good RMSE score. For instance, with the same $r = 300$, the error is 13.54 for $\lambda = 0.1$, compared to 11.98 of $\lambda = 0.005$. Meanwhile, fixed λ value, the RMSE decreases with the increasing

Journal Pre-proof

Journal Pre-proof

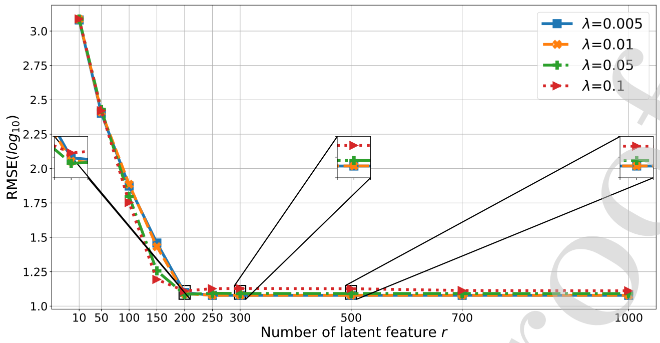


Figure 5: Training RMSE obtained from the ALS factorization as a function of (λ, r).

number of latent features r. Clearly, the ALS accuracy reaches a plateau at around $r = 300$ and less improvement is reached afterwards. Therefore, we can tell that the RMSE monotonically decreases with larger r, even though the improvement diminishes gradually.

Given aforementioned observations, we accordingly adopt $r = 300$ and $\lambda = 0.005$ to the proposed algorithm in the following experiments for the following reasons: the proposed algorithm achieved a satisfactory fitting result of the input app-usage data (low RMSE error), and an affordable computational time (smaller value of r) is expected.

Neighborhood (N) and β. In the proposed TAG algorithm, parameters N and β are used to adjust the clustering probability based on school similarity. Smaller value of N and bigger β indicates that less influence from neighbors is taken into account for adjusting the clustering outcome. In particular, when $\beta = 1$, the proposed algorithm becomes the traditional GMM. It simply implies that no adjustment is considered from similar schools. By contrast, if $\beta = 0$ schools are determined by their N neighbors for clustering, similar to a process of the N-nearest neighbors. As such, we would like to evaluate the robustness of the proposed algorithm using various combinations of (N,β).

Journal Pre-proof

Accordingly, the neighborhood size is measured by $N \in [3, 22]$; on the other hand, the penalty term is set as $\beta \in [0, 1]$.

To decide the best parameter, the Bayesian information criterion (BIC) measurement is employed. In general, BIC is a statistical measurement to balance between model complexity and performance, while a clustering outcome associated with a lower BIC value is usually preferred. In this study, the BIC value is estimated using the python library of scikit-learn ([23]). Herein we set the cluster size as $K = 5$, $r = 300$, and $\lambda = 0.005$ respectively, and the resultant clustering performance is illustrated in Fig. 6.

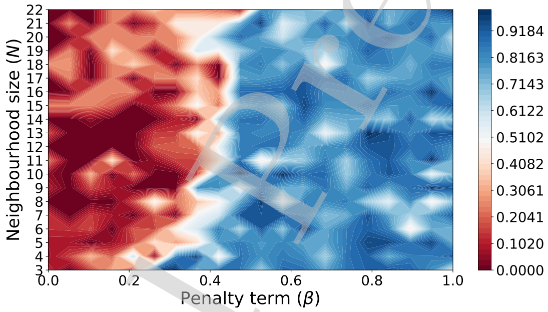


Figure 6: Clustering performance (measured by BIC) as the function of neighborhood size (N) and penalty term (β).

Generally speaking, from the comparison result, a bigger value of penalty term (β) leads to a poor clustering. For instance, a bigger BIC value is observed when $\beta \geqslant 0.4$ compared to that of $\beta < 0.4$. Note that in Eq. (11), the penalty term controls how much a clustering probability is influenced by its neighborhoods. In particular, when $\beta = 1$, the proposed clustering became to the standard GMM algorithm. However, the experimental results clearly indicated that a smaller value of β is preferred to improve the clustering

Journal Pre-proof

performance. Consequently, we confirm that the smoothed clustering outperforms the traditional GMM algorithm, as an improved performance is achieved while neighborhood-based adjustment is considered. On the other hand, with the same setting of penalty term (i.e., $0.1 \leqslant \beta < 0.4$), a range of size (i.e., $N \in [10, 14]$)is generally favored as a smaller BIC result is observed.

Note that some optimization algorithms (such as Bayesian optimization technique) can be further employed to automatically search for the optimal combination of parameters (i.e., λ, r, β, and N). However, this work is beyond the scope of this paper. We leave the parameter optimization in our future study. Overall, we consider setting key parameters as $r = 300$ and $\lambda = 0.005$ for the ALS factorization, and $\beta = 0.213$ and $N = 12$ for smoothing GMM results.

5.3. Comparison with other works

As verified by previous studies [24, 25], students' online behavior consistently aligns with their learning performance (i.e., their NAPLAN score). In other words, app-consumption based clustering should lead to grouping those schools with similar NAPLAN performance. The more similar clustering outcome compared to the NAPLAN score, the better clustering method is. Therefore, in this study, the NAPLAN score is employed as the ground-truth information for clustering. However, note that these NAPLAN results are only provided to schools four months after the test is completed, and it is only administered to students every two years (school years 3, 5, 7 and 9). That is, the possible effect of students' daily performance and online practice is often unclear. The proposed profiling method can be used to bridge the gap between annual NAPLAN scores and students' daily activity.

On the other hand, in our proposed work, we focus on (i) introducing

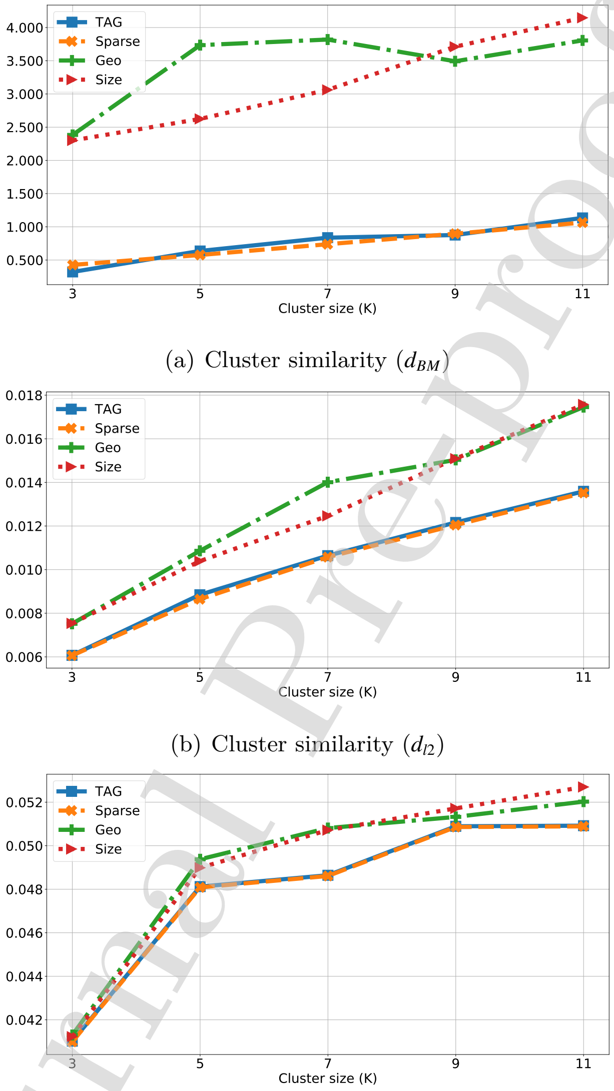
Journal Pre-proof

an ALS-based factorization technique for cleaning sparse data and reducing data dimension; (ii) applying the smoothed GMM algorithm for clustering. Therefore, our experiments are designed to evaluate the applicability of both the cleaned data (from ALS) and the improved GMM method.

To begin with, we examine the validity of reduced data obtained using the ALS-factorization technique. It will be compared with the original sparse data (before ALS), school profile (such as school size, in terms of the number of enrolled students, and geolocation, in terms of latitude and longitude). The proposed GMM-based method is applied for the clustering purpose, while settings of key parameters are referred to Section 5.2. For simplicity, the result obtained from ALS-based data, sparse data, school size, and geolocation is denoted as “TAG”, “Sparse”, “Size” and “Geo”, respectively. Additionally, three evaluation metrics proposed in Section 4.4 are employed to measure the similarity between the NAPLAN-based clustering and other approaches. As such, the results are shown in Figure 7. In this figure, the X-axis plots the number of cluster size (K varies from 3 to 11) and the Y-axis plots the similarity.

As observed, the results from both TAG and Sparse are superior than those of Size and Geo, which indicates the app-usage based data is capable of effectively capturing students’ characteristics. Those online profile indeed alines with their learning performance in reality. On the other hand, school profile (such as number of students or location) is insufficient to represent or connect with students’ daily behavior, which leads to a poorer clustering outcome. Additionally, it is also found that the ALS-based data achieves comparable clustering result as Sparse. Note that the Sparse data represents the entire usage dataset without any dimension reduction. The similar performance implies the reduced dataset (from ALS) is sufficient for

Journal Pre-proof



(a) Cluster similarity (d_{BM})

(b) Cluster similarity (d_{l2})

(c) Cluster similarity (d_{kl})

Figure 7: Comparisons among different inputs for clustering, including “TAG” (ALS-based data), “Sparse” (original data before ALS), “Size” (school size) , and “Geo” (geolocation) respectively.

Journal Pre-proof

the clustering purpose without losing much information. Another advantage of ALS-based data comes from its less-dimension, that is beneficial to the subsequent clustering process via reducing the computational cost.

Next, the performance of the proposed GMM algorithm is compared with some state-of-the-art clustering methods, while the same input information is given (i.e., the reduced data from ALS as inputs). The aim of the following experiment is to verify the robustness of our method, against different numbers of clusters. The following five clustering algorithms are included in this paper for comparison:

1. Traditional Spectral Clustering (TSC) method starts from constructing a similarity graph [26]. Vertexes and edges from this graph represent the data sample and similarity among samples, respectively. Then TSC aims to partition this graph into groups based on weights of edges;
2. Sparse subspace clustering (SSC) is a subspace-based clustering technique [27]. In SSC, each data sample is represented using a liner combination of other samples, while regularizing the coefficients with the l_1 norm;
3. Low Rank clustering (LRC) is another type of subspace-based clustering method [28]. The major difference between SSC and LRC is that LRC considers the nuclear-norm constraint for coefficients, instead of the l_1 norm;
4. Elastic Net Clustering (ENC) introduces a so-called "elastic net regularizer" (a mixture of the l_1 and l_2 norms) and applies it to derive a scalable representation for original data samples [29];
5. Sparse K-Means Clustering (SKMC) groups samples according to only

Journal Pre-proof

Journal Pre-proof

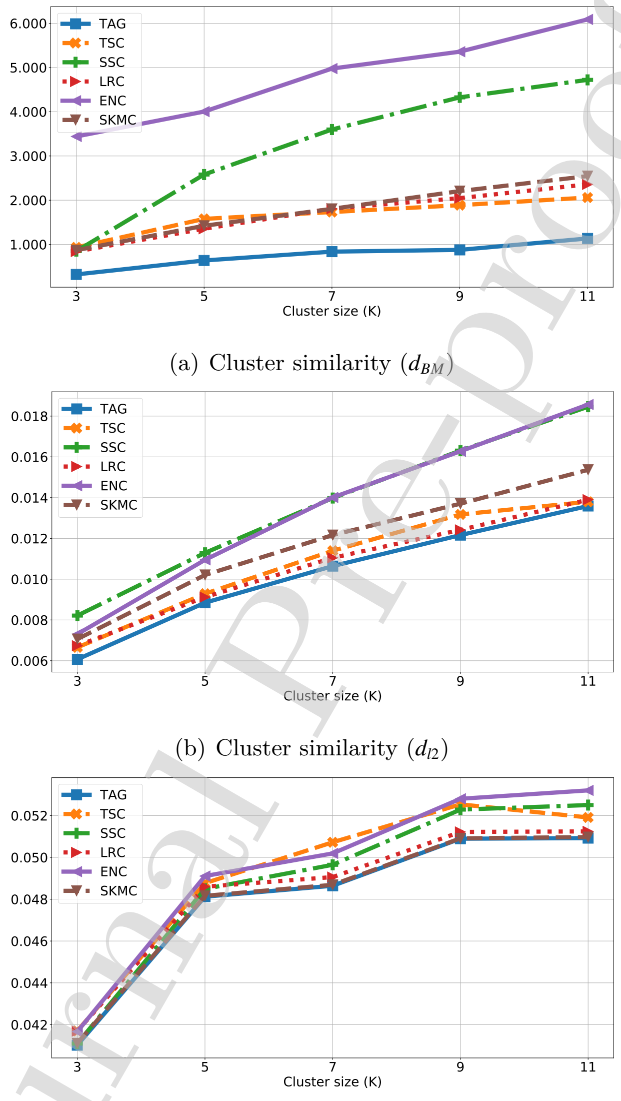
an adaptively-chosen subset of original features [30]. In SKMC, only a small portion of features is selected by maximizing the within-clusters similarity and between-clusters dissimilarity.

Figure. 8 presents the averaged results from different approaches over 10 runs. Clearly, the proposed algorithm yields the best clustering ability in comparison to other methods, while TSC, LRC and SKMC methods come second, and SSC and ENC methods perform worst. Meanwhile, with the increasing number of clusters, clustering performance from all six methods drops; however, our proposed TAG algorithm achieves relatively stable results compared to others, which demonstrates its robustness towards different numbers of clusters.

Table 3 reports the averaged computational time (in second) from various clustering algorithms over 10 runs. As observed, the proposed TAG algorithm requires the affordable processing time, compared to most of other methods. In total, approximately 1 second is spent on the completion of all five clustering tasks. On the other hand, SKMC takes the longest time as it needs more than 30 seconds on average for only one clustering task. The computational cost for the LRC algorithm is also expensive as its bottleneck is the operation of singular value decomposition. We also notice that the SSC method spends less time than TAG, especially while the number of clusters is more than 7. Nevertheless, the performance from TAG is much better than that of SSC in terms of the clustering accuracy, which compensates for its computational cost.

From the aforementioned comparison, it can also be empirically confirmed that the proposed TAG algorithm can appropriately group schools based on aggregated online activity, which is predictive of their offline pat-

Journal Pre-proof



(a) Cluster similarity (d_{BM})

(b) Cluster similarity (d_{l2})

(c) Cluster similarity (d_{kl})

Figure 8: Clustering outcome from different approaches measured by similarity metrics, i.e., d_{BM}, d_{l2}, and d_{kl}.

Journal Pre-proof

Table 3: Summary of the computational time (second) from the proposed TAG algorithm against existing algorithms.

Algorithms	#.Cluster = 3	#.Cluster = 5	#.Cluster = 7	#.Cluster = 9	#.Cluster = 11
TAG	0.099 ± 0.001	0.145 ± 0.001	0.191 ± 0.001	0.239 ± 0.001	0.287 ± 0.001
TSC	0.28 ± 0.002	0.363 ± 0.002	0.386 ± 0.002	0.444 ± 0.003	0.48 ± 0.002
SSC	0.15 ± 0.003	0.158 ± 0.001	0.171 ± 0.009	0.177 ± 0.003	0.178 ± 0.002
LRC	19.085 ± 0.06	19.097 ± 0.052	19.112 ± 0.043	19.092 ± 0.056	19.118 ± 0.074
ENC	9.88 ± 0.081	9.869 ± 0.03	9.928 ± 0.066	9.901 ± 0.1	9.888 ± 0.048
SKMC	32.129 ± 0.112	32.171 ± 0.083	32.225 ± 0.107	32.178 ± 0.08	32.282 ± 0.122

tern (i.e., performance on standardized tests). That is, schools achieving similar NAPLAN scores end up with similar app-usage pattern.

6. Within-cluster analysis

In this section, we accordingly propose a two-step methodology to systematically conduct the within-cluster analysis. Firstly, we conduct a comprehensive analysis of usage dynamics cross each single cluster. The purpose is to find out whether consumption from few application categories dominates school online activity. Secondly, we will analyze the correlation between individual application category and school learning performance, which may help in a better planning for online learning or regularization of app consumption.

At first, we investigate usage pattern from individual clusters. The category information from an app indicates the functionality and application domains of the app. We can infer the school needs and interests according to their selected app's category. To make a reference, we find out Top-15 (out of 36 in total) app categories across the entire dataset, which is shown

Journal Pre-proof

in Fig. 9. Note that here these Top-15 categories are calculated using data from selected 191 schools, while Top-10 categories shown in Table 2 are based on the entire dataset from all 258 schools.

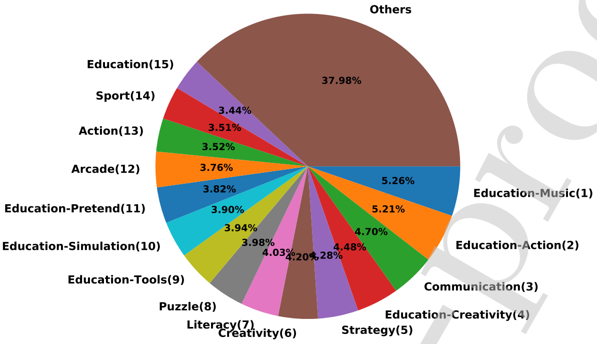


Figure 9: Top-15 popular app categories (and related consumption percentage) across the entire dataset. The digital number inside the bracket represents the category order.

As observed, accumulated consumption from Top-15 popular categories nearly dominate school online activity, as they occupied approximately 62.02% of app usage. We then utilize the app consumption from Top-15 category as a base line, to verify the difference among clusters. The comparison is illustrated in Fig. 10, and some observations (for each individual cluster) are summarized as follows:

Cluster 1: top-3 categories from this cluster are: “Strategy” (13.91%), “Puzzle” (11.92%), “Education-Tools” (10.42%), respectively, as shown in Fig. 10(a). We also notice a minor consumption from the category of “Education-Music” and “Creativity” within this cluster, which are ranked as the 1st and 6th popular category from the entire dataset;

Cluster 2: the consumption distributions are relatively even in this cluster. This implies that schools have accessed a diverse set of applications.

Journal Pre-proof

Meanwhile, the categories of popular apps from this cluster is quiet similar with the average data;

Cluster 3: schools from this cluster seem to use more apps from categories of “Education-Tools”(9.78%), “Strategy” (9.07%), and “Education-Simulation” (8.80%). On the other hand, as shown in in Fig. 10(c), a relatively smaller consumption is observed in terms of the category of “Education-Music” and “Creativity”, that is similar to the pattern from Cluster 1;

Cluster 4: “Education-Creativity” (15.08%), “Education-Tools” (13.86%), and “Education” (11.83%) are the most popular categories from this cluster. Another important difference is the absence of app consumption for the type of “Education-Music”, “Education-Action”, “Education-Simulation”;

Cluster 5: we observe that three categories (e.g. “Strategy” (29.52%), “Arcade” (16.96%), and “Action” (11.57%)) make up a predominant percentage of online consumption in this cluster. By contrast, the most popular category (i.e. “Education-Music”) identified from Fig. 9 is again observed with a zero consumption in this cluster.

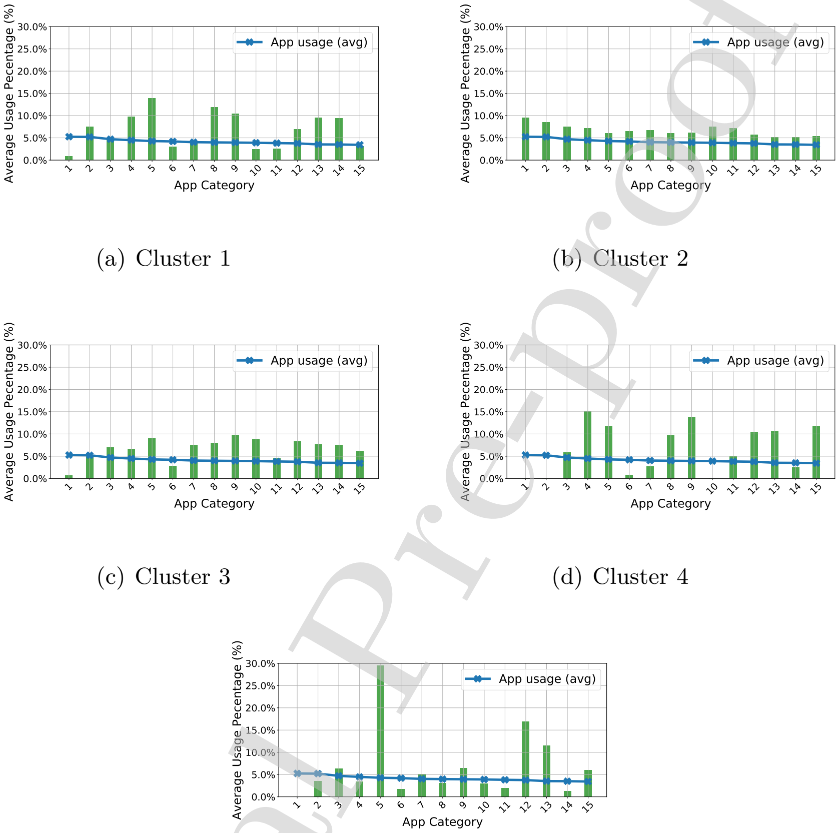
As such, we confirm that a few app categories dominate school online activity from some clusters (such as the cluster of 4 and 5), while other schools seem to utilize a diverse set of applications. Next, we use the information entropy of usage to further measure the diversity of app consumption in one cluster (say x), which is defined as below:

$$H(\mathrm{x}) = -\sum_{i}^{n} p(x_i) \log p(x_i), \tag{18}$$

where $\mathrm{x} \in \mathbb{R}^n$, $p(x_i)$ represents the access probability for each element x_i ($x_i \in \mathrm{x}, i \in \{1, \ldots, n\}$), and it can be calculated as:

$$p(x_i) = \frac{count(x_i)}{\sum_i^n count(x_i)}, \tag{19}$$

Journal Pre-proof



(a) Cluster 1

(b) Cluster 2

(c) Cluster 3

(d) Cluster 4

(e) Cluster 5

Figure 10: App usage (consumption percentage) from each individual cluster, compared to those from overall Top-15 categories (represented using the solid line). The X-axis represents overall-top categories (i.e., “Education-Music” (label 1), “Education-Action” (2), “Communication” (3), “Education-Creativity” (4), “Strategy” (5), “Creativity” (6), “Literacy” (7), “Puzzle” (8), “Education-Tools” (9), “Education-Simulation” (10), “Education-Pretend” (11), “Arcade” (12), “Action” (13), “Sport” (14), and “Education” (15)), and the Y-axis plots the actual consumption from each individual cluster.

Journal Pre-proof

where *count*(x_i) denotes the total duration/consumption of x_i.

In this paper, we would like to investigate the entropy for both of the category and individual apps within different clusters. Therefore, for the app category, we have x_i representing the i-th type of apps, where $i \in \{1, \ldots, n\}$ and $n = 36$ (total 36 categories); on the other hand, as for the individual app, we have x_i representing the i-th app, and $i \in \{1, \ldots, 30155\}$ (as there are in total 30,155 mobile applications available from the entire dataset). Additionally, we also normalize the entropy measurement for a fair comparison among different clusters at the end.

According to Eq. (19), the bigger normalized entropy value is, the more even distribution of apps or categories schools utilized. By contrast, a small entropy value indicates less variety of apps while app interest might be limited. Clusters associated with small entropy value tend to be limited on a certain online resources (i.e. few applications or categories). We finally list the entropy results for each clusters in Table 4. Note that "ALL" in Table 4 represent the average entropy using data from all five clusters.

Table 4: Comparison among 5 different clusters in terms of the activity entropy.

	Entropy	
	By categories	By applications
Cluster 1	4.51	9.00
Cluster 2	4.94	12.39
Cluster 3	4.85	10.01
Cluster 4	4.39	6.82
Cluster 5	4.06	6.86
ALL	4.95	12.48

Journal Pre-proof

As observed, the result of the category entropy is consistent with that of Fig. 10. For instance, the average category entropy is 4.95 from the entire dataset, which is similar with the outcome from Cluster 2. That further confirms that schools from Cluster 2 have a more broader interest in numerous application categories (as it also scores the higher entropy value). On the other hand, the most limited online activity is observed from Cluster 5, that is associated with the lowest entropy (4.06).

Another interesting observation comes from the entropy of applications. Again, Cluster 2 has the bigger value, that implies that related schools tend to utilize or explore more number of applications (not only the categories) compared to other clusters. However, the lowest entropy of applications is from Cluster 4 (instead of Cluster 5 like the category case). The reason can be schools from Cluster 4 play more number of applications than Cluster 5, but these apps fall in a smaller number of categories.

In the aforementioned cluster analysis, we focus on the app consumption (in terms of categories and individual applications) associated with different clusters. The following analysis will further uncover the relationship between online activity and learning performance.

Table 5 takes the number of clusters (K=5) as an illustration example. We observe differences in terms of learning performance across 5 clusters. Generally, Cluster 1 and 4 score lowest (less than 400 on average), while Cluster 5 has the highest score.

Result from Table 5 indicates an interesting relationship between app usage and learning performance. For instance, Cluster 5 is limited by the range of apps and categories, which indicates schools from this cluster have a specific-interested topics and focus. The concentration could lead to a better learning outcome (451.4 on average in their NAPLAN score).

Journal Pre-proof

Table 5: Average NAPLAN score per cluster, where the number of clusters (K) is 5.

Subjects	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Numeracy	386.1±38.7	419.1±29.6	416.0±31.0	369.1±25.7	423.6±15.6
Reading	386.4±49.8	428.9±29.0	422.3±30.8	384.5±22.3	451.8±13.2
Grammar	399.9±40.4	438.7±29.6	425.1±26.5	405.2±15.9	463.5±17.1
Spelling	396.1±53.2	438.4±28.0	430.2±23.3	415.7±12.7	467.7±12.1
Writing	392.2±39.7	428.4±21.8	422.2±21.0	421.2±14.2	450.2±10.5
Average	392.1±44.4	430.7±27.6	423.2±26.5	399.1±18.2	451.4±13.7

By contrast, it is not guaranteed that the high variety of app usage results in poor learning. Two typical examples can be found from cluster 2 and 3. We noted that both these two clusters have similar profile in terms of entropy of apps and categories, and have similar NAPLAN scores (around 430.7 and 423.2, respectively). The result clearly indicates schools from these two clusters tend to use a great diversity of apps (corresponding to the relatively higher entropy). Their broad interest might help in developing different skill sets, which could improve their learning outcome. Additionally, schools from cluster 1 and 4 show the lowest NAPLAN results, while the related app entropy is somehow belonging to the middle range. Comparisons from our study then imply that schools, either concentrating on certain apps or utilizing numerous online resources, perform better than their peers. However, schools need to select appropriate apps.

Another interesting observation comes from the different impact of apps usage on learning subjects. For instance, all clusters score the similar "Nu-

Journal Pre-proof

Journal Pre-proof

meracy" results, but subjects like "Grammar" and "Spelling" vary dramatically. This difference implies that existing apps might have limited contribution to improve "Numeracy", but have impact on learning aspects that are associated with their writing skill (such as "Grammar" and "Spelling").

7. Conclusion

In this paper, we maintain a national mobile application dataset using the open-source big-data platform (i.e., Hadoop and Spark, etc.). Additionally, an enhanced clustering algorithm (termed as TAG) is introduced for online-usage profiling, which consists of three major steps:

- T-step: to establish a normalized TF-IDF matrix for representing aggregated usage data;
- A-step: to apply ALS-based matrix factorization technique to reduce data sparseness and size (dimension);
- G-step: to employ a smoothed GMM algorithm for clustering schools.

Empirically, the proposed TAG algorithm is evaluated by streaming data collected from approximately 200 Australian schools, which yields a more-accurate clustering result in comparison to traditional methods. More precisely, we showed that the TAG algorithm can deal with high-dimensional usage data, while still keeps the data characteristics (based on the T/A steps). Additionally, it further smooths original cluster results via automatically grouping similar schools (in the G step). By testing with different cluster sizes and evaluating with three performance metrics, experimental results show that the TAG algorithm can appropriately cluster schools based on their online activity.

Journal Pre-proof

Journal Pre-proof

The experimental outcome further confirms that school online behavior (app consumption) strongly align with their learning performance. Given the importance of high-quality early learning experiences, it is essential that the use of apps in schools is better understood. As such, we also explore varied app usage pattern within distinct clusters separately. Findings provide depth insights into the complexity of multiple app use from schools associated with different learning performance. Implications of online patterns in relation to learning and teaching are also discussed.

Research on discovering online activity pattern is still in its active stages. In this paper, we have investigated an improved GMM algorithm for activity profiling. There are many possibilities for future research directions, two of which are mentioned below:

1. online sequential clustering: the proposed algorithm is based on an offline learning flowchart, which means the input data for clustering is stationary and fixed. It would be very interesting to extend the proposed method to online sequential clustering, where the streaming input arrive continuously;
2. other complex tasks: Another appealing extension is to utilize the proposed algorithm in other domains or more-sophisticated datasets. For instance, the application of our method in recommendation systems or leadership identification in social network might be of great interest, in particular, when a large-scale dataset is present.

References

[1] X. Liu, H. Li, X. Lu, T. Xie, Q. Mei, F. Feng, H. Mei, Understanding diverse usage patterns from large-scale appstore-service profiles, IEEE

Journal Pre-proof

Transactions on Software Engineering 44 (2018) 384–411.

[2] Gartner, Leading the IoT, 2017. URL: https://www.gartner.com/imagesrv/books/iot/iotEbook_digital.pdf.

[3] Ericsson, Ericsson mobility report, 2018. URL: https://www.ericsson.com/491e34/assets/local/mobility-report/documents/2018/ericsson-mobility-report-november-2018.pdf.

[4] N. Seyff, F. Graf, N. Maiden, Using mobile RE tools to give end-users their own voice, in: 2010 18th IEEE International Requirements Engineering Conference, 2010, pp. 37–46.

[5] O. I. Franko, T. F. Tirrell, Smartphone app use among medical providers in acgme training programs, Journal of Medical Systems 36 (2011) 3135–3139.

[6] W. Hardyman, A. Bullock, A. Brown, S. Carter-Ingram, M. Stacey, Mobile technology supporting trainee doctors' workplace learning and patient care: an evaluation, BMC Medical Education 13 (2013) 6.

[7] S. L. Lim, P. J. Bentley, N. Kanakam, F. Ishikawa, S. Honiden, Investigating country differences in mobile app user behavior and challenges for software engineering, IEEE Transactions on Software Engineering 41 (2015) 40–64.

[8] S. Vakulenko, O. Muller, J. V. Brocke, Enriching itunes app store categories via topic modeling, in: International Conference on Information Systems, 2014, p. 1–11.

[9] D. Lavid Ben Lulu, T. Kuflik, Functionality-based clustering using short textual description: Helping users to find apps installed on their

Journal Pre-proof

mobile device, in: Proceedings of the 2013 International Conference on Intelligent User Interfaces, IUI '13, 2013, pp. 297–306.

[10] H. Khalid, E. Shihab, M. Nagappan, A. E. Hassan, What do mobile app users complain about?, IEEE Software 32 (2015) 70–77.

[11] M. Tavakoli, L. Zhao, A. Heydari, G. Nenadic, Extracting useful software development information from mobile application reviews: A survey of intelligent mining techniques and tools, Expert Systems with Applications 113 (2018) 186 – 199.

[12] W. Martin, F. Sarro, Y. Jia, Y. Zhang, M. Harman, A survey of app store analysis for software engineering, IEEE Transactions on Software Engineering 43 (2017) 817–847.

[13] J. Yang, B. Yecies, Mining chinese social media ugc: a big-data framework for analyzing douban movie reviews, Journal of Big Data 3 (2016). doi:10.1186/s40537-015-0037-9.

[14] J. Yang, B. Yecies, P. Y. Zhong, Characteristics of chinese online movie reviews and opinion leadership identification, International Journal of Human–Computer Interaction 0 (2019) 1–16. doi:10.1080/10447318.2019.1625570.

[15] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang, J. Wang, Geospatial and temporal dynamics of application usage in cellular data networks, IEEE Transactions on Mobile Computing 14 (2015) 1369–1381.

[16] S. Zhao, G. Pan, Y. Zhao, J. Tao, J. Chen, S. Li, Z. Wu, Mining user attributes using large-scale app lists of smartphones, IEEE Systems Journal 11 (2017) 315–323.

Journal Pre-proof

Journal Pre-proof

[17] C. Xie, H. Cai, Y. Yang, L. Jiang, P. Yang, User profiling in elderly healthcare services in china: Scalper detection, IEEE Journal of Biomedical and Health Informatics 22 (2018) 1796–1806.

[18] X. Luo, M. Zhou, S. Li, Y. Xia, Z. You, Q. Zhu, H. Leung, Incorporation of efficient second-order solvers into latent factor models for accurate prediction of missing qos data, IEEE Transactions on Cybernetics 48 (2018) 1216–1228.

[19] X. Luo, M. Zhou, S. Li, M. Shang, An inherently nonnegative latent factor model for high-dimensional and sparse matrices from industrial applications, IEEE Transactions on Industrial Informatics 14 (2018) 2011–2022.

[20] X. Luo, H. Wu, H. Yuan, M. Zhou, Temporal pattern-aware qos prediction via biased non-negative latent factorization of tensors, IEEE Transactions on Cybernetics (2019) 1–12.

[21] Y. Zhou, D. Wilkinson, R. Schreiber, R. Pan, Large-scale parallel collaborative filtering for the netflix prize, in: Algorithmic Aspects in Information and Management, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 337–348.

[22] M. K. Goldberg, M. Hayvanovych, M. Magdon-Ismail, Measuring similarity between sets of overlapping clusters, in: 2010 IEEE Second International Conference on Social Computing, 2010, pp. 303–308.

[23] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duch-

Journal Pre-proof

esnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[24] H. C. Wei, H. Peng, C. Chou, Can more interactivity improve learning achievement in an online course? Effects of college students' perception and actual use of a course-management system on their learning achievement, Computers & Education 83 (2015) 10 – 21.

[25] W. Xing, R. Guo, E. Petakovic, S. Goggins, Participation-based student final performance prediction model through interpretable genetic programming: Integrating learning analytics, educational data mining and theory, Computers in Human Behavior 47 (2015) 168 – 181.

[26] H. Liu, T. Liu, J. Wu, D. Tao, Y. Fu, Spectral ensemble clustering, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '15, Sydney, NSW, Australia, 2015, pp. 715–724.

[27] C. You, D. P. Robinson, R. Vidal, Scalable sparse subspace clustering by orthogonal matching pursuit, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 3918–3927.

[28] J. Chen, H. Mao, Y. Sang, Z. Yi, Subspace clustering using a symmetric low-rank representation, Knowledge-Based Systems 127 (2017) 46 – 57.

[29] C. You, C. Li, D. P. Robinson, R. Vidal, Oracle based active set algorithm for scalable elastic net subspace clustering, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 3928–3937.

Journal Pre-proof

[30] Y. Kondo, M. Salibian-Barrera, R. Zamar, RSKC: An R package for a robust and sparse K-Means clustering algorithm, Journal of Statistical Software, Articles 72 (2016).

Journal Pre-proof

Journal Pre-proof

AUTHOR DECLARATION TEMPLATE

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us.

We confirm that we have given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In so doing we confirm that we have followed the regulations of our institutions concerning intellectual property.

We understand that the Corresponding Author is the sole contact for the Editorial process (including Editorial Manager and direct communications with the office). He/she is responsible for communicating with the other authors about progress, submissions of revisions and final approval of proofs.

We confirm that we have provided a current, correct email address which is accessible by the Corresponding Author and which has been configured to accept email from jiey@uow.edu.au, jma@uow.edu.au, sahoward@uow.edu.au

Signed by all authors as follows:

Name Jie Yang	Signature	Date 22/06/2019
Name Jun Ma	Signature	Date 22/06/2019
Name Sarah K. Howard	Signature	Date 22/06/2019

Journal Pre-proof