JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015

Mobile Contextual Recommender System for Online Social Media

Chao Wu, Student Member, IEEE, Yaoxue Zhang, Jia Jia, Member, IEEE, and Wenwu Zhu, Fellow, IEEE,

Abstract—Exponential growth of media consumption in online social networks demands effective recommendation to improve the quality of experience especially for on-the-go mobile users. By means of large-scale trace-driven measurements over mobile Twitter traces from users, we reveal the significance of affective features in shaping users' social media behaviors. Existing recommender systems however, rarely support such psychological effect in real-life. To capture such effect, in this paper we propose Kaleido, a real mobile system that achieves an online social media recommendation solution by taking affective context into account. Specifically, we design a machine learning mechanism to infer the affective pulse of online social media. Furthermore, a cluster-based latent bias model (LBM) is provided for jointly training the affective pulse as well as user's behavior, location and social contexts. Our comprehensive trace-driven experiments on Android prototype expose a superior prediction accuracy of 87%, which has 25% accuracy superior to existing mobile recommender systems. Moreover, by enabling users to offload their machine learning procedures to the deployed edge-cloud testbed, our system achieves speed-up of a factor of 1,000 against the local data training execution on smartphones.

Index Terms—Mobile recommender system, affective computing, social networks.

1 INTRODUCTION

The mass-adoption of mobile social networking services and the wide integration of sharing media content on popular mobile applications have paved the way for quantitative research efforts tackling the relations between content and virality [2]. Very recently, a novel kind of user experience is working its way through the online space, i.e., designed to take the affective pulses appeared in online social network (OSN) usage, such interface supports people to explicitly bridge their emotional states with OSN information subscription [3].

To capture users' attention, many tweets in OSNs nowadays are usually published with affective media content [4]. In real-life usage, due to the *affective pulse* [5], i.e., the first feeling when accessing an object, triggers the mind 3,000x faster than rational thoughts, many users sensibly make a quick decision whether to subscribe a media tweet by such affective pulse (e.g., happiness, sadness, or disgust) [6]. This actually opens up a new venue to integrate the affective feature for future media recommender system design. Indeed, by means of large-scale data-driven measurements in our early stage work, we reveal that 60% user clicks are motivated by media contents, among which, more than 76% are triggered with explicitly affective pulses (§2.2). Moreover, with further affect-aware measurement in our established system, users are benefited with 82% accuracy to subscribe the right tweets (§5.3). Therefore, it is highly promising to rethink the social media recommender mechanism by jointly considering users' affective pulses as well as traditional features.



1

Fig. 1. Conceptual Kaleido diagram.

Existing recommender systems, however, provide recommendations largely based on users' content preference, using content- [7], demographic- [8], knowledge- [9], geographical- [10], or utility- [11] based methods. To boost the prediction accuracy, recent studies [12] and [13] proposed methods which input the user's OSN usage pattern into a linear regression (LR) for prediction. To further improve social media propagation or streaming delivery, some researchers try to use both user preference and network context to determine the appropriate presentation method [14]. However, we have yet to see an approach that shifts mobile social media recommendation from the above approaches to a scheme that jointly tackles user's feeling, which plays a critical role in media content consumption in OSNs, behavior pattern (i.e., user preference, content attributes, media formats, time, and network), location preference (i.e., the likelihood of consuming media in different locations), and social closeness (i.e., social interaction strength).

To fill this void, in this paper we propose a real system

Preliminary results of this paper has been presented in [1]

C. Wu, J. Jia, and W. Zhu, are with the Department of Computer Science and Technology, Tsinghua University, Beijing, China. E-mail: {cwu12, jjia, ybw12]@{mails, mail, mail}.tsinghua.edu.cn

Y. Zhang is with the School of Information Science and Engineering, Central South University, Changsha, China. E-mail: zyx@csu.edu.cn

Kaleido, which jointly utilizes the unique visual features (to capture user's affective pulse context), mobile behavior context, location context, and social context, in OSNs for mobile media recommendation. In particular, let us consider the case of Fig. 1 where, by integrating Kaleido into a third-party Twitter app, i.e., Twidere [15], the user's social media experience is benefited from her habits along both affective (i.e., happiness) and social closeness (i.e., keep eyes on the sender) directions. Targeting at this goal, we propose a learning-based mechanism to infer affective pulses from media files. On this basis, we further combine the affective feature with user behavior pattern, location preference, and social friendship to predict users' potential interests in online social media usage. Moreover, we employ such inference and network environment (e.g., WiFi available or not), to execute the whole prediction.

Specifically, to better understand affective pulses in social media, we employ a learning-based affective computing mechanism and a Flickr image dataset with well-known ground-truths, which has been manual tagged with affective tags by prior work [16], to infer the probability distribution of the affective pulse. In particular, as visually presented in Fig. 7, we input the affective pulse as a 6-dimension feature to the recommender algorithm. On the other hand, through early stage measurements, we observe that both location and social friendship (i.e., the social interaction strength among users in OSN) has a critical impact on the user's tweet click behavior. Then we conduct the location clustering to identify its significance in mobile media recommendation as well as the social friendship clustering to classify a user's social friends into different groups with different levels of importance. On this basis, we next design a cluster-based Latent Bias Model (LBM) to predict user's likelihood of media click with considering all the above contexts as well as different network and time contexts. Last, by integrating Kaleido into Twidere, an Android Twitter app which has 500,000 downloads on Google Play, we collect user traces from a demographical composition of 16,952 people who consented to report usage data to us. This also enables us to conduct a data-driven measurement and design and realistic experiments to evaluate the performance of Kaleido's mobility support (§5.3). In addition, by collecting system logs from our edge-cloud servers, we reveal the effect of Kaleido testbed (§5.2).

We summarize the major contributions and vantage points of this paper as:

- We collect a large set of real-life mobile traces from 16,952 participants, and reveal the significance and importance of affective feature in social media usage. To our best knowledge, we are the first to employ the affective context in such recommender system.
- We propose a learning-based model to infer affective pulses in media files with 75% accuracy. Furthermore, we design a cluster-based LBM for jointly training affect, behavior, location and social contextual features. Through data-driven experiments, we illustrate that it achieves a prediction accuracy of 87%, which outperforms baselines using the same training features.
- · We deploy a worldwide edge-cloud testbed for real-



Fig. 2. Framework of Kaleido system.

life Twidere users by enabling them to offload machine learning procedures with a speed-up of 1,000x factor over the local execution. Our Android prototype consumes user with a low cost of cellular data and energy, which is a significant improvement against the benchmark approaches.

We shall emphasize the feasibility of using Twidere and Android smartphone as a study case. Nevertheless, the proposed methedologies are applicable to most mobile apps and OS platforms.

The rest of this paper is organised as follows: §2 outlines system framework of Kaleido, describes how the system design and work through Twidere app. §3 proposes the affective computing approach to identify the affective pulse in media content. §4 designs the machine learning mechanism that joint each features to train user's likelihood click. §5 conducts the trace-driven emulation evaluation on smartphones for evaluating Kaleido's performance. Then in §6, we discuss the impact factors in our system. §7 reviews the homogeneous studies, and §8 concludes this paper as well as future work.

2 KALEIDO: THE FRAMEWORK AND SYSTEM DESIGN

Kaleido provides a real system which consists of an edgecloud testbed as well as a mobile framework. We start with a brief review of Kaleido system (§2.1), and close the section with describing the key measurements that relate to inspire Kaleido's design details (§2.2).

To keep the illustration concrete, we assume for now that all servers use a learning-based algorithm to identify affective pulses in media; we relax that assumption in the next section. We defer a discussion of the overall algorithm within Kaleido until §4.

2.1 System Overview

Conceptual Framework: To better present the logic, Fig. 2 introduces the framework of Kaleido from a high level. By learning affective pulses, social closeness, location preference and behavior pattern, Kaleido enables users with media recommendation during their operating process. More specifically, when a fresh media content arrives, Kaleido is triggered to take the relevant features of the media context (user behavior, affect, location and social friendship) as

JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015



Fig. 3. Logical workflow of Kaleido.

input to the trained cluster-based LBM, to identify the likelihood of user actions ahead of time. We will elaborate how the proposed cluster-based LBM jointly train the features in §4.4.

Mobility Support: Fig. 3 depicts how Kaleido mobile framework works in a user-centric manner (i.e., implemented on a user's mobile device), and collects social feeds when accessing new media content with Twitter application. Specifically, it offloads user's machine learning procedure to cloud servers and pushes the training patterns to smartphones asynchronously. Also note that these traces are retrieved using the Twitter REST APIs [17] in the servers in according to collected user traces. Furthermore, by pushing the training pattern to smartphone, Kaleido executes the inference locally and rapidly. Finally, it comes out with the recommended tweets for user in according to the inference results (e.g., see Fig. 1).

System Infrastructure: Similar to existing techniques, Kaleido employs an edge-cloud architecture for building a testbed to process users' machine learning offloading. Specifically, such architecture supports the time-varying bandwidth and storage allocations requested by different regions. By collecting system logs from early stage deployed servers, we further setup the hardware configuration as illustrated in Table 5. We further discuss the efficiency of such infrastructure in §5.2.

Caching Policy: Since the traces are cached temporarily, to ensure privacy, text content in tweets is not recorded and all personal fields are anonymized in advance. In addition, the cached data is uploaded to the cloud server only for further analysis when the smartphone is charging and connected to WiFi. To accelerate the process, Kaleido offloads the machine learning procedure to the testbed whose deployment information are shown in Table 5. Furthermore, the learned patterns are stored on the device to enable real-time inference by Kaleido.

Network Congestion: To relief the network congestion caused by the offloading procedure, in this paper, end-user's processing is delivered to the topological nearest server by leveraging her DNS resolution, as we elaborate on in §5.1.

2.2 Measurements in Early Stage Work

To capture the effect of affective pulse in recommending social media, we conduct data-driven measurements in early stage work.

Specifically, inspired by a very recent study [16], in this paper, we adopt 6 basic affects in describing human



Fig. 4. Kaleido users' demographic composition.

TABLE 1 Description of collected user profiles.

Contents	Collected traces		
Tweets	time, sender, receivers		
Media files	URL link, sender, receivers		
User behaviors	publish, like, retweet, mention		
App usage	launch time, close time, present time		
Feature training	start time, end time		
Coarse location	coordinates		
User request	DNS resolution, HTTP metadata		

emotional state [18], i.e., happiness, surprise, anger, disgust, fear and sadness, to investigate the forming of users' affective pulses. Furthermore, for evaluating the effectiveness of affective learning in §3, we use a well known dataset from Flickr which covers more than 1 million image traces that published by 4,725 users and has been manual tagged with the affective labels by prior work [16]. To further investigate users' media behaivor in mobile environment, during the whole early stage, i.e., from March to October 2015, as Fig. 4 illustrates, we also have collected data traces from more than 16,952 Twidere [15] users ¹ from all over the world with a diverse demographic composition. Because, although Twitter's contents are publicly available, information about when, how, and where they access these social streams are not available in particular in the mobile environment. As the aim is to enable Kaleido system by identifying the affective pulse within media tweets that the user is most interested in, a set of tweet attributes are collected as well. Twidere tracks the user social behaviors (e.g., retweet, like, or mention) of the individual tweets. The source of a tweet is also recorded by identifying whether the tweet is obtained from a direct friend or propagated through friends of others' friends. In addition, with the consent from the user, Twidere enables us to keep track of the user's activity events when reading the tweets (watching, clicking, or commenting along the timeline). Note that we also collect users' coarse location traces at the same time. The collected trace items are shown in Table 1. Moreover, in this paper, we deploy the Kaleido geographical servers by referring to user composition pro-

1536-1233 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information

^{1.} Twidere discloses the usage statistics on installation or update. Users are able to opt in or not. In the early stage work, 43% active users grant us permissions, which indicates user privacy-awareness and the effectiveness of the privacy disclosure. We keep the collected data in an anonymous and irreversible style. Also note that the social graphs and tweets are publicly available.

JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015



Fig. 5. 16,952 users' app usage across 7 months.





Fig. 6. A user's media usage across hour of day.

vided by Fig. 4.

The users clicked more than 900 million referred tweets in total. Specifically, as Fig. 5 illustrates, on average, a user refreshes 536 tweets a day but only clicked 15% of them. We further observe that more than 60% of the clicked tweets contain media files. Moreover, by referring to the affective learning in §3, we find that more than 76% media have explicitly affective pulses. Note that we infer whether a media file owns an explicitly affect, which shapes the affective pulse, by comparing with the corresponding probability with a set of trained baseline thresholds ² in §3. Fig. 6 illustrates that users are time-sensitive, e.g., on weekdays, the user tends to use the app more frequently in the nighttime especially in the midnight, while use the app sporadicly in the daytime. It motivates us to analyze user behavior pattern by taking time feature in accounts. The volume and diversity of data also reflects the real-life behavior of the participant users, which is crucial for understanding and recommending media in mobile social application network traffic, and significant in evaluating the system in a datadriven scheme.

3 LEARNING AFFECTIVE PULSE FROM MEDIA CONTENT

In this section, we introduce a learning-based affective computing mechanism by which we identify the affective pulse in a media file.

As aforementioned in §1, the forming of affective pulse, e.g., probability distribution of the 6 affects in an object, is complicated. To proceed, we employ a machine learning process. Specifically, we denote the space of affective pulse as $\mathcal{A} = \{Happiness, Surprise, Anger, Disgust, Fear, Sadness\}.$ As different people might have distinct affect even when accessing the same media ³, in our learning algorithm, we adopt the approach provided by [4], i.e., we denote every affect as a linguistic label for the matchup in the learning process. To proceed, we further denote the affect $a \in A$ expressed by a media file. Inspired by [16], we use visual variables to capture the affect expressed by the media file. We characterize each affect by training with 7 visual features, i.e., saturation (SR), saturation contrast (SRC), brightness (B), bright contrast (BRC), 5 dominant HSV colors (DC), cool

TABLE 2 Single affect leanring accuracy (%) of baselines by testing 1 million images with ground-truths.

	Kaleido	Naive Bayes	SVM (with linear kernel)
Happiness	76.18	57.05	56.12
Surprise	72.32	52.94	52.25
Anger	75.56	57.11	55.42
Disgust	76.28	56.64	55.40
Fear	74.77	55.26	54.05
Sadness	77.88	60.10	56.95

color ratio (CCR), and dull color ratio (DCR). Thus, we have feature set $\mathcal{U} = \{SR, SRC, B, BRC, DC, CCR, DCR\}$. For each media $c \in C$, we have the ground truth $a_c \in \{1, 0\}$ such that $a_c = 1$ ($a_c = 0$) means that whether the media file *c* has the specific affective feature $a \in \mathcal{A}$ (e.g., happiness) or not. Then we have:

$$f(u_c, a_c) = \frac{1}{z_{\theta}} \exp\{\theta^T u_c\}$$
(1)

where u_c are the mentioned visual features, a_c is the affect express by the media file c, θ is a vector of real valued parameters, and z_{θ} is a normalization term to avoid the potential over-fittings. On this basis, the probability distribution P of the affect a in the media files $c \in C$ can be formulated as:

$$P(A|\mathcal{C}) = \frac{1}{Z} \prod_{\mathcal{C}} f(u_c, a_c) = \frac{1}{Z} \exp\{\theta^T \beta\}$$
(2)

where $Z = z_{\theta}$ is the normalization term, β is the aggregation of factor function. In fact, this is exactly multi-variant logistic regression, and can be solved by using L2 regularization.

Furthermore, by taking all the affects $a \in A$ in account, the objective function can be derived as:

$$\mathcal{O} = \log P(A|\mathcal{C}) = \log \sum_{A|A^{\mathcal{U}}} \exp\{\theta^T \beta\} \quad \log Z$$
$$= \log \sum_{A|A^{\mathcal{U}}} \exp\{\theta^T \beta\} \quad \log \sum_{A} \exp\{\theta^T \beta\}$$
(3)

In addition, the gradient of θ can be represented as:

$$\frac{\partial \mathcal{O}}{\partial \theta} = \frac{\partial (\log \sum_{A|A^{\mathcal{U}}} \exp\{\theta^T \beta\} - \log \sum_A \exp\{\theta^A \beta\})}{\partial \theta}$$
(4)
$$= (\exp_{P(A|A^{\mathcal{U}})} - \exp_{P(A)})\beta$$

^{2.} In our measurements, the baseline set inclines to {0.24, 0.14, 0.63, 0.19, 0.33, 0.49}. However, it changes with different OSN cases.

^{3.} Note that since the volume of video sharing in Twitter is rare. Thus, unless otherwise specified, we mainly focus on inferring affective pulses of images.



Fig. 7. Flowchart of learning affective pulse from a media. We first extract the values of 7 visual features respectively. By training with the Flickr dataset, it learns the probabilities of 6 affects with the loopy belief propagation (LBP) [19] algorithm. Last, we regularize each P(a|C) (where $a \in A$) with comparing its probability with corresponding baseline.



Fig. 8. Statistics of a user's social interaction frequency and friendship clustering with 1 year usage.

Note that the algorithm updates the parameters by $\theta = \theta_0 + \frac{\partial O}{\partial \theta} \lambda$. Where λ is a regularization parameter to be manually tuned.

Visual Illustration: After illustrating the affective learning model, Fig. 7 visually illustrates how each visual feature u_c contributes to the proposed mechanism ⁴. We take a snapshot of the Christmas cat & dog as a study case, it results in a set of probabilities of {0.43, 0.18, 0.10, 0.18, 0.18, 0.21}. By filtering the insignificant affective features via comparing with the baselines (as illustrated in §2.2), it reports that the happiness and surprise features are significant and hence have value 1 while the rest affective features are with value 0. Finally, we obtain the affective pulse of {1, 1, 0, 0, 0, 0} as the input affective feature to the recommender model later.

In addition, through training with 80,000 affect-aware images that manual tagged by [16], we further evaluate the accuracy of our learning algorithm and baselines in Table 2. We observe that our model achieves an average prediction accuracy of 0.75, which significantly beyonds the benchmarks in identifying the probability distribution of each affect $a \in A$ in a specific media. This again demonstrates the efficiency of the proposed affective machine learning. On the other hand, we also discuss the significance and importance of adopting affective pulse in recommending media in §6.

TABLE 3 Media click probability (%) by measuring different features of 16,952 users across one year.

		close	familiar	unfamiliar
Totality	Media click	51.33	28.39	13.67
Time	Weekdays	56.38	32.63	13.06
Features	Weekends	36.94	19.30	11.75
	Mentioned by	28.57	17.72	10.81
Interaction	Liked by	43.66	27.92	14.58
Features	Retweeted by	38.10	27.69	9.60
	Replied by	51.82	28.85	13.42

4 JOINT RECOMMENDATION WITH AFFECTS, BE-HAVIOR, LOCATION AND SOCIAL CONTEXTS

In this section, we first conduct a data-driven analysis on user's mobile behavior, location traces and social interactions. Then we reveal their impact on the user's media click actions. On this basis, we last introduce how Kaleido jointly train the affective features with these three features for a location-based affect-aware social media recommendation.

4.1 Behavior Pattern Analysis

In OSNs, the generation and propagation of a media content is simple: any user who generates or re-shares it will become a new host of the media content. Users can fetch these contents from their direct friends in the social network. Intuitively, the social relationships and interactions among a user and her friends have a significant impact on the twittering behaviors. A user might treat different friends differently, and interact with some close friends frequently, while having little contact or response with some unfamiliar friends on Twitter [21].

As mentioned in §2.2, user's media click actions enjoy characteristics of high selectiveness and time sensitivity. Along this direction, in Fig. 8(a), we plot the number of one real-life user's clicked tweets with media content (i.e., media tweets) from her friends (i.e., social neighbors) on Twitter in the log-log scale. We rank the set of friends in descending order according to the number of tweets sent by them. We

^{4.} Note that the DC feature consists of a 15-dimension HSV martix. Due to space limit, we omit the details of DC feature in Fig. 7's pipeline assembling illustration, and interested readers can refer to [20].

JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015



Fig. 9. A study case of geo-distribution in according to a user's affective pulses consumption among consecutive 6 months.

observe a strong power law phenomenon, i.e., almost 70% of the tweets are from only a few friends (less than 10%), and most other friends have little contribution. This demonstrates that friendship (or social interaction strength) plays a critical role on shaping her usage behavior on Twitter.

4.2 Location Preference in Media Consumption

We then explore the impact of *location* ⁵ on user's media click action. For illustration, we first plot such information by utilizing Google Maps [22]. Intuitively, we further conduct a clustering process to investigate user's daily routine. Specifically, as Fig. 9 illustrates, one user tend to be more active around two places ⁶. Each blue or red point represents at least 1,000 nearby media content click.

Based on such preliminaries, we then cluster the historical events to estimate their most likely number of geocentrals for each user. Specifically, as Fig. 9 indicates, a user is often active around two categories of geographical centrals (geo-centrals). This drives us to make the location feature as an input for the proposed algorithm in this paper, as we describe later.

As aforementioned, users tend to be active around regular locations, e.g., *office* and *home*. This motivates us to group notifications into several clusters to identify the geographical centrals for the characterization of user's location patterns. For instance, the office-home case, as Algorithm 1 depicts, given a set of time-aware historical geolocation records (l_1, l_2, \ldots, l_M) of the media tweet, our approach starts with clustering each l_m into corresponding clusters and refers the geo-central of the cluster as the (coarse) user location.

Specifically, we first carry out the geolocation clustering process to partition user's twitter activities with several geocentrals. Similar to many location-based studies [23], [24], [25], we utilize the geolocations in notifications received from social friends as the clustering feature, and use the *Single-Linkage* clustering algorithm [26] to carry out the geolocation clustering with different geo-centrals and distance thresholds.

Moreover, we denote the referred geo-centrals as \mathcal{G} . Note that, in this case, we actually have $||\mathcal{G}|| = ||\mathcal{G}^*|| + 1$ Algorithm 1: Home-Office clustering algorithm.Input: Geolocation records $\mathcal{R} = \{(l_1, t_1), \dots, (l_n, t_n)\}.$ Output: Geographical situation of each (l_i, t_i) . $\mathcal{R}' = Geo_Clustering(\mathcal{R});$ while $(l_i, t_i) \in \mathcal{R}'$ doLocation $(l_i, t_i) =$ Others;if $t_i \in \{weekends, holidays, home time in workdays\}$ thenLocation $(l_i, t_i) =$ AtHome;if $t_i \in \{work time of workdays\}$ thenLocation $(l_i, t_i) =$ AtOffice;

6



Fig. 10. Impact on different number of geo clustering centrals in the cluster-based LBM algorithm.

clusters since sporadical and irregular geolocation samples are grouped as a cluster of "Others", which plays a noisy role in our algorithm. Intuitively, in order to train the prediction mechanism well, more fine-grained data traces with more detailed information with respect to all geolocation clusters are needed. Nevertheless, in practice it is extremely difficult to obtain such fine grained data traces since some geolocation data samples are distributed sporadically and irregularly, which results in a clustering with insufficient information. In addition, when G is too large, the number of training parameters increases significantly due to the large number of geo-centrals, which results in a heavy machine learning workload (see §5).

4.2.1 Effectiveness of " $\mathcal{G}^* = \{2 \ centrals\}$ " Location Clustering

As illustrated in Fig. 10, we denote $G = ||\mathcal{G}^*||$ and, when G = 2, almost 80% of Twitter activities are concentrated within a distance range of 140 meters. The activities would be more concentrated from each other within a range of 120, 20, and 10 meters when G = 3, 4, and 5, respectively. In addition, the GPS accuracy is around 7.8 meters and that could make influence on our analysis. Given this fact, the geographical geometrical distance in Fig. 11 is calculated from each geolocation to its corresponding cluster central. Moreover, by jointly training with other features in our proposed algorithm (see §4.4), the G = 2 case performs significantly better in the location clustering with an accuracy gap of 0.16⁷. This is also consistent with our observation in Fig. 9 that most media contents are concentrated consumed around 2 geolocations, e.g., home and office.

7. Note that the gap value caused by both different geo and social clustering numbers might variate from users. Thus, in this paper, we obtain them by averaging the heavy users (see §5.3) for illustrating the significance of our approach.

^{5.} With location, we denote a geographical situation, e.g. at home (in contrast to geolocations).

^{6.} Note that we have also explored another 100 users, the results are the general case.

JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015



Fig. 11. Geo-cluster coverage within different geographical geometrical distance to the cluster centrals, respectively.

4.3 Social Friendship Closeness

We further quantify the impact of social friendship on the user's media tweet click behaviors. To proceed, we first carry out the social friendship clustering. The intuition is that in reality a user typically has very close relationships with a small set of people (e.g., close friends), and is familiar with a group of people (e.g., colleagues). For many other people, the user would have little contact with them. With this observation, we conduct the friendship clustering using the commonly-adopted an unsupervised machine learning with K-Means algorithm [27]. As illustrated in Fig. 8(b), we utilize the number of tweets subscribed from a specific friend and the number of tweets published by the user to that friend as the features, and cluster the set of her friends into three types: close friends (i.e., cluster "close"), familiar friends (i.e., cluster "familiar"), and acquaintances with infrequent contacts (i.e., cluster "unfamiliar").

After the social friendship clustering, we then explore the impact of friendship on users' media click behavior when accessing the media tweets. Table 3 measures all users' average media click probability under different feature scenarios. In total, we observe that users click the media file with a probability of 0.42 (0.28, 0.13), when the media is sent by a close (familiar, unfamiliar) friend, respectively. This again confirms that social friendship has a significant impact on user's media click behavior. As another example, for the interaction feature, if the media tweet has been replied or mentioned by a close friend, then user would click the media file with a probability gap of 0.17.

4.3.1 Effectiveness of K = 3 Social Clustering

Fig. 12 visually confirms that K = 3 achieves a good balance and the best prediction accuracy with a performance gap of 0.13. In addition, it comforts a fact in our daily life observation that people tend to classify their friends into three types: close friends, familiar friends, and acquaintances.

4.4 Training with Affect for Recommendation

After illustrating mobile behavior pattern, location preference and social friendship closeness, we now introduce the machine learning principle in our system by jointly identifying the set of important training features to build up the learning model.

4.4.1 Training Context Features

As mentioned above, we found that four types of context features are critical: affective pulse in media, behavior pattern, , location preference during media usage, and social



7

Fig. 12. Accuracy variation of different social cluster number by referring to all users' usage.

closeness. Furthermore, we use these four features as the input to our proposed cluster-based machine learning algorithm as:

- For the behavior feature, whenever the user would click a media depends on her regular usage behavior.
- 2) For the affective feature, what she prefers to click is mainly influenced by each affect that reflect from the media content.
- 3) For the location feature, around which geo-central user tends to click the media file is counted.
- 4) For the social feature, the host (close, familiar or unfamiliar friend) of receiving media file is decisive.

In the following, we denote the number of these training features as F, and their set as \mathcal{F} .

4.4.2 Cluster-Based Latent Bias Model

We propose the learning model which is based on the Latent Bias Model (LBM) introduced in [28] that aims to utilize proper bias terms to capture the importance of different features for prediction. In fact, this inspires sophisticated collaborative filtering models such as matrix factorization. Here we extend the standard LBM to our case with social friendship clustering, and develop the cluster-based LBM approach for a data-driven learning scheme.

In particular, we define $b_{f,a,g,k}$ as the cluster-based bias term to stand for the case that a media is sent by a friend in the friendship cluster k. The parameter a expresses potential affects $a_i \in \mathcal{A}$. In addition, the clustering location central $g_i \in \mathcal{G}$ indicates where a user is nearest when clicking media content. Last, the bias term should also contain the feature $f \in \mathcal{F}$ that we introduce above. Thus, for a given media content c, we first introduce an indicator function

JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015



Fig. 13. Accuracy of three baseline algorithms.

 $I_{f,a,g,k}^c \in \{0,1\}$ such that $I_{f,a,g,k}^c = 1$ if the media c is sent by a friend in the friendship cluster $k \in \mathcal{K}$ with affect $a \in \mathcal{A}$ and contains the feature $f \in \mathcal{F}$. Then, we define the user action score for the media c as follows:

$$\gamma_c = \alpha + b_0 + \sum_{f \in \mathcal{F}} \sum_{a \in \mathcal{A}} \sum_{g \in \mathcal{G}} \sum_{k \in \mathcal{K}} b_{f,a,g,k} I^c_{f,a,g,k}$$
(5)

 α is the user's average click rate across all historical media content usage, and b_0 is the overall bias for the user. In general, a higher user action score γ_c implies a higher probability that the user will click the media content *c*.

Then, the critical task is to train the cluster-based LBM, i.e., to learn the proper bias terms in Equation 5 in order to well capture a user's media click actions. Suppose that the set of historical user data traces (historical media usage set of the user) is denoted as C. For each media $c \in C$, we have the ground truth $y_c \in \{1, 1\}$ such that $y_c = 1$ ($y_c =$

1) means that the user clicks (open) the media file c in tweet of arrival. To quantify the discrepancy between the prediction based on the media click score γ_c and the desired ground-truth output y_c , we adopt the widely-used logistic loss measure

$$\mathcal{L}(\gamma_c, y_c) = \log[1 + \exp(-\gamma_c y_c)]. \tag{6}$$

Thus, we learn proper bias terms to minimize the total loss across over the historical data trace C, i.e., $\sum_{c \in C} \mathcal{L}(\gamma_c, y_c)$. Following common practice in machine learning, in order to avoid overfitting, we also impose L_2 regulation into minimization. That is, we minimize the following objective function:

$$\mathcal{O} = \sum_{c \in \mathcal{C}} \mathcal{L}(\gamma_c, y_c) + \lambda \left(||b_0||^2 + \sum_{f \in \mathcal{F}} \sum_{a \in \mathcal{A}} \sum_{g \in \mathcal{G}} \sum_{k \in \mathcal{K}} ||b_{f,a,g,k}||^2 \right)$$
(7)

where λ is the regularization parameter to be manually tuned.

Since the function in Equation 7 is convex, we can apply the first-order condition and derive the gradients as

$$\frac{\partial \mathcal{O}}{\partial b_0} = \sum_{c \in \mathcal{C}} \left(\frac{\exp(\gamma_c y_c)}{1 + \exp(\gamma_c y_c)} \right) y_c + 2\lambda b_0, \qquad (8)$$
$$\frac{\partial \mathcal{O}}{\partial b_{f,a,g,k}} = \sum_{c \in \mathcal{C}} \left(\frac{\exp(\gamma_c y_c)}{1 + \exp(\gamma_c y_c)} \right) y_c I_{f,a,g,k}^c + 2\lambda b_{f,a,g} q_{g,k}^c$$

Similar to many machine learning studies, we can adopt the Stochastic Gradient Descent (SGD) method [29], to learn

TABLE 4 Summarization of Fig. 13.

Algorithm	Balanced	Best	Worst
	accuracy	accuracy	accuracy
Cluster-based LBM	0.87	0.91	0.70
Linear regression	0.76	0.84	0.66
SVM with linear kernel	0.69	0.79	0.52

optimal bias terms. The key idea is to utilize the data samples to iteratively update the gradient as follows:

$$b_*^{t+1} = b_*^t \quad \epsilon_t \frac{\partial \mathcal{O}}{\partial b_*^t},\tag{10}$$

8

where b_*^t denotes a given bias term at the *t*-th iteration and $0 < \epsilon_t < 1$ is the smoothing factor for updating. As shown in [29], the SGD method converges to the optimal learning point provided a sufficiently small ϵ_t .

After learning, when a fresh tweet with media c arrives, we predict a user's click likelihood using the loss measure in (6). Specifically, we predict that a user will click the media file if $y_c = 1$ has a lower risk, i.e., $\mathcal{L}(\gamma_c, y_c = 1) < \mathcal{L}(\gamma_c, y_c = 1)$. In this case, the media is clicked by the user and otherwise in reverse.

4.4.3 Model Baselines

We last evaluate the performance of the proposed clusterbased LBM algorithm by referring to all the participants' usages. Fig. 13 depicts the cumulative probability distribution (CDF) of the prediction accuracy for all tested tweets. As mentioned above, when the friendship cluster number K = 3 and geo-central number G = 2, it achieves the best performance with an average prediction accuracy of 0.87. As baselines, we also evaluate the prediction process with linear regression (LR) in [13] and SVM. Jointed with Table 4, Fig. 13 evaluates all baselines' performance that LR approach can mostly achieve an average prediction accuracy of 0.66, which outperforms SVM approach by 16%, but be inferior to the cluster-based LBM by 11%. This demonstrates the efficiency of the cluster-based LBM. The gain of clusterbased LBM stems from the fact that the defined clusterbased bias terms can well capture the impact of affective feature on user's media click, we elaborate it in §6.

5 EXPERIMENTS

In this section, we conduct the trace-driven evaluation over the testbed and Android smartphones to investigate the performance of Kaleido.

5.1 Implementations

To evaluate the performance of Kaleido system, we deploy a worldwide network testbed to undertake users' machine learning procedure. Table 5 summarizes the network topology, geographic distribution and configuration of our testbed. Specifically, there are 3 categories of service for the deployed servers, i.e., server-load balancing (as a streaming server), caching (as a data center), and content service (as a computing center).

1536-1233 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information

JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015

.

9

TABLE 5 Details of network topology, geographic distribution and configuration of Kaleido testbed.

Region	Location	Carrier and OS	Bandwidth	Service	Configuration (CPU, Memory, Storage)
LIC	Los Angeles	Vultr (Debian 8.0)	10Mb	Server-load Balancing	2.4GHz single-core, 768MB, 20GB
05	San Mateo	Aliyun (Debian 7.5)	5MB	Caching, Content Services	2.3GHz dual-core, 2GB, 1TB
Europe	Frankfurt	Vultr (Debian 8.0)	10Mb	Server-load Balancing	2.4GHz single-core, 768MB, 15GB
Asia Tokyo Shanghai	Tokyo	Vultr (Ubuntu 14.04)	10Mb	Server-load Balancing	2.4GHz single-core, 512MB, 20GB
	Shanghai	Tencent (CentOS 7)	5MB	Caching, Content Services	2.6GHz quad-core, 12GB, 1TB
	Beijing	Campus (Debian 8.0)	1MB	Caching	2.2GHz octa-core, 128GB, 6TB

Moreover, to illustrate the effect of performance of the provided mobile framework, we also run a trace-driven evaluation for the 16,952 users that each of them keeps active for a long and consecutive period of at least 3 months with detailed trace records. The emulator runs on a Google Nexus 6, Nexus 5 and Samsung Galaxy S4 smartphone, respectively, that with access to both China Mobile TDD LTE network as well as a campus WiFi network. The emulator reads and replays the usage events collected from real-life users, including connecting to or disconnecting from WiFi networks, accessing Twitter, and reacting its media files.

5.2 Testbed Measurements

The testbed serves a diverse workload spanning massive machine learning procedure: friendship clustering, training, media content downloading, content caching, and other miscellaneous process. We use the latest 3 months' (from December 2015 to March 2016) user request logs and UNIX round robin database (RRD) logs collected from 6 vantage servers, i.e., 2 in US, 1 in Europe and 3 in Asia, to measure the performance of our deployed testbed. To this end, we leverage the Metalink standard [30], which is an XML-based download description format that provides the metadata of the content⁸. Metalink-enabled HTTP clients and proxies understand the relevant HTTP headers (e.g., to verify the authenticity and integrity of the data, discover faster mirrors, etc.), while legacy clients simply ignore them. In addition, the RRD log reports the system load, network bandwidth and stock prices with a constant disk footprint.

Table 6 summarizes that the balance and costs of Kaleido edge-cloud infrastructure. Specifically, we recorded that the most heavy system load appeared in the machine learning process server (with an average usage of 21.4%) while the most bandwidth consuming was the server-load balancing (with an average usage of 0.2MBps)⁹. In addition, the latency for all the Kaleido edge servers are less than 670ms which again comforts the factuality of our testbed.

5.3 Benchmarks

After the testbed efficiency has been discussed, we next evaluate the performance of Kaleido mobile framework

TABLE 6 Testbed performance across 3 months.

Service	Latency (ms)	Bandwidth usage (%)	System load (%)
Service	170	4.7	21.4
Caching	211	3	7.3
Server-load	670	14.4	5.7

running on the smartphones in against with existing representative recommendation algorithms. As illustrated in Fig. 13 the cluster-based LBM algorithm in Kaleido is very efficient, and achieves the average prediction accuracy of 0.87. Upon comparison, the linear regression (LR) algorithm using tweet training features can only achieve the average prediction accuracy of 0.66. In the following, we also compare different recommendation algorithms given as: In the following, we further consider the three recommendation approaches as performance benchmarks in this paper as:

- *Kaleido approach*: We implement the recommender system by running the proposed cluster-based LBM with affective context, user behavior contexts, and social contexts.
- *Content-based approach*: We implement the contentbased recommender system like [7] by using labels, key words and social topics as training features.
- *Social contextual approach*: We replace the content features with social information contexts in [14].

Specifically, we evaluate the performance of Kaleido recommender system, by considering different users, i.e., top 5% users (the most 750 active users who refresh 1,301 tweets and consume 40 media tweets per day on average), top 30% users (users that daily refresh 780 tweets and consume 22 media tweets), and top 60% users (users that daily refresh 429 tweets and consume 11 media tweets). For the evaluation of each approach, the same user trace are replayed, and media files are obtained by using the WebView component provided by the Android framework. To embrace stable data and battery consumption results, we evaluate this two items by training with each user's first 2 month trace and using at least another 2 month traces for testing.

Accuracy Growth: We first compare different approaches' accuracy growth with time. As illustrated in Fig. 14(a), our algorithm tends to be stale after one month's

^{8.} E.g., see http://releases.ubuntu.com/releases/15.10/ubuntu-15.10-desktop-i386.metalink.

^{9.} Note that this low average bandwidth consumption is due to the training procedure for all users are sporadicly triggered when each smartphone is in charge and WiFi available and after a time slot of one day from the last calling. In addition, user profiles are usually no more than 200KB.



Fig. 14. Benchmarks across accuracy growth, daily battery overhead and monthly cellular data cost.

usage. We see that Kaleido approach varies within a range of 0.65 to 0.87 and has a median of 0.81, when after 2 months' use. Upon comparison, after the same time usage, i.e., two months, content-based and social contextual approaches start with lower accuracy of 0.62 and 0.65, and perform median values of 0.74 and 0.76 respectively. This is due to the fact that affective feature plays a significant role in learning users' media click (we shall discuss it in §6). This demonstrates the efficiency of the proposed Kaleido approach in media recommendation.

Data and Battery Overheads: We next compare different recommendation approaches in terms of cellular data consumption of Twidere app (which includes the cellular traffic overhead by both Kaleido and on-demand contexts fetching by the user), and battery consumption of the app per day (i.e., the percentage of the fully-charged battery capacity). The results are shown in Fig. 14(c) and Fig. 14(b) respectively. We observe, take the top 5% users for instance, that Kaleido uses 6.8MB cellular traffic per month and about 1.4% battery usage per day on average. On the other hand, the content-based (social contextual) approach costs users with 7.5MB (5.8MB) cellular traffic per month and 1.1% (1.2%) battery usage per day on average.

Moreover, evaluations in Fig. 14 also illustrate Kaleido outperforms existing approaches in all kinds of usage cases.

6 DISCUSSIONS

6.1 Does Affective Pulse Really Matter?

Kaleido, as mentioned above, is the first step towards an affective media recommender system. Performance of the proposed algorithm in this paper demonstrates that Kaleido is promising when integrating the affective feature with user behavior and social friendship contexts. However, we have yet known how critical the affective feature plays, i.e., only using content and context features. To understand it, in Fig. 15, we compare the accuracies of all the mentioned algorithms in §4.4 (with the same user group) by removing the affective feature, i.e., we only put the behavior and social contexts into the training model. Similar to the prediction evaluation above, we adopt the K = 3 social clustering as the study case. We observe that the LBM enjoys a positive impact with affective feature and achieves an average prediction accuracy of 0.71, which has 11% performance degradation with respect to the standard algorithm. Upon comparison, the LR (SVM) algorithm achieves an accuracy of 0.68 (0.66), with a slightly small performance degradation



Fig. 15. Evaluation of no affective feature in cluster-based LBM (with 11% degradation), LR (with 3% degradation), and SVM (with 2% degradation).

of 0.03 (0.02). This again demonstrates the uniqueness and significance of Kaleido approach in exploiting the affective feature for efficient media recommendation.

6.2 Why Training on Cloud?

One key component of Kaleido is to implement the clusterbased LBM algorithm for the data training (i.e., learning the optimal bias terms from the user data traces). Intuitively, there are two data processing approaches: 1) processing on local device, i.e., we conduct the data training procedure on user's smartphone locally. 2) processing on cloud, i.e., we offload the data training to the cloud server is to leverage the strong parallel computing power to speed up the data training. To investigate the characteristics, in Fig. 16, we emulate the average time overhead of these two data processing approaches with the top 60% active users, by setting different size of user trace in the learning algorithm, with considering the network connection latency we measured in Table 6. We find that the cloud approach can significantly decrease the daily time overhead for data processing, by a factor of 1,000. We further compare the monthly data consumption for training on cloud for different users respectively. We compare that this procedure consumes user largely 2.5MB¹⁰ data per month, which confirms that the effectiveness of training on cloud. Note that since the user's behavior tends to be stable, to further save both cellular data and energy

10. Note that the Kaleido consumes additional 4.5MB/month since additional data/information download is needed for the testing.



(a) Daily training time consumption

(b) Daily battery overhead for training

(c) Monthly data cost for training

11

Fig. 16. Overhead with different training schemes for the machine learning procedure in Kaleido.



Fig. 17. Daily costs with different testing schemes for Kaleido.

usage, we can aggregate the training data for a longer while (e.g., one week) but not everyday any more, and carry out data training weekly by offloading the training to the cloud only when the user is on WiFi while charging.

6.3 Why Testing on Local?

Furthermore, we also evaluate the testing schemes based on above approaches. The results are shown in Fig. 17. We observe that the cloud approach consumes more energy (0.5% of total battery usage per day), leading to additional cellular data usage (0.9MB per month). In addition, since the cloud approach requires network connection, it brings a latency of 670ms each time, daily time overhead for testing all the tweets on local ties to the cloud approach. Thus, the local testing approach is more preferable. The reason is that, different from the training process that is computation-intensive, the testing procedure is data-centric and offloading to the cloud would incur higher delay and energy overhead for data exchange between the cloud and the device.

6.4 How does Location Feature Impact?

Kaleido is a first step towards recommending the social media for mobile users. Our algorithm's performance demonstrates that Kaleido is promising when integrating the geolocation feature with other three key features for machine learning processing.

However, it may happen that in practice geolocation is disabled. On mobile OSs, e.g., Android and iOS, a subsystem called *Location Services* provides access to the user's current coarse geolocation [31]. iOS allows users to limit an app's usage of location services on a per-app basis. In particular, when an app first attempts to access location services, the user is asked to grant or deny access. Android provides a similar subsystem, where users can toggle access to location on a global level for all apps on the phone. Due



Fig. 18. Evaluation of no location feature in cluster-based LBM (with 17% degradation).

to these features, Kaleido on Android tends to less likely explore location than iOS.

Motivated by this observation, in Fig. 18, we further investigate the accuracy of the proposed cluster-based LBM algorithm without access to location. It depicts the CDF of the prediction accuracy for all twitter notifications without location support. Similar to the accuracy evaluation above, we adopt the 3 friendship cluster case. It achieves an average prediction accuracy of 0.69, with a performance degradation of 0.18 with respect to the case with location feature. This shows the importance of location in Kaleido. As benchmark, we also implement the prediction process with linear regression (LR) without location, which has an average accuracy of 0.61. Through extensive trace-driven evaluation in the following section, we confirm that Kaleido without access to location can achieve better performance over other benchmark approaches. This is mainly due to the significance of the social friendship feature. Thus, if a user cares about her location privacy and disables geolocation, Kaleido is still a superior solution among the alternatives without access to location.

JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015



Fig. 19. Energy usage performance with/without location feature on Google Nexus 6 smartphone.

6.5 Limitations of Using Location Feature

6.5.1 Privacy Concerns

In practice, to protect sensitive privacy, it can happen that the geolocation access is disabled by user. Specifically, in modern mobile OS, e.g., Android or iOS, a subsystem called Location Services provides access to the user's current coarse geolocation. iOS allows users to limit an app's usage of location services on a per-app basis. In particular, when an app first attempts to access location services, the user is asked to grant or deny access. Android provides a similar subsystem, where users can toggle access to location on a global level for all apps on the phone. Thus, the location feature some times can be a potential factor of privacy leakage. Due to above reasons, Kaleido tends to less likely call location information (context). Thus, it raises a high demand that our system should also has better performance than original mechanism when there is no location feature support.

6.5.2 Energy Usage increase

We also explore the energy usage performance caused by invoking location feature. As Fig. 19 shows, through extensive trace-driven evaluation on the basis of above measurements, we confirm that Kaleido without access to location can achieve better performance than using location service. Specifically, in worst (common) case, using location service with Kaleido can cost 0.3% more battery cost ever day. Note that this is very few increase because, the mobile social application can refresh the location information. Thus, we can read the data from cache directly. Morever, if a user cares about his/her location privacy and disables geolocation, as discussed before (§6.4), Kaleido is still a superior solution.

7 RELATED WORK

In this section, we review three directions of prior research directly related to our work. Specifically, we highlight the key differences of Kaleido against them respectively.

7.1 Media-based Affective Computing

Although many recent studies, e.g., [32], keep on emphasizing the significance of affective media computing, but their approaches are still highly subjective and difficult to embrace quantitative measurements [33]. A recent study [16] focus on training data and models for identifying the emotional influence, based on the ground-truth affects that are manually labeled in order to guarantee the prediction accuracy. However, it also brings a challenge to efficiently process massive images in OSNs [16]. Motivated by this issue, [34] uses image tags and comments from user behaviors to predict potential affect expressed by the media content, i.e., image, in social networks. Along a different line, motivated by the insight that user behavior, social friendship and media affect play critical roles on user's emotion-triggered action in OSNs, in this paper, we propose a novel learning-based mechanism to intelligently deal with media recommender system which support the affectiveaware recommendation.

12

7.2 Recommendation Techniques

There are two prevalent schemes for building recommender systems, i.e., content-based (CB) [13] and collaborative filtering (CF) [35]. The CB method is on a basis of recommending items, e.g., images or videos, that are similar to those in which users are interested in according to the historical feeds. The CF approach, on the other hand, recommends items to the user based on other individuals with similar preferences or tastes. Many recent studies, such as [14], [36], [37], are built on both CB and CF systems, usually by rating a set of items. To avoid this extra burden on the user, leveraging implicit interest indicators [38], such as the purchase history, views, clicks, or queries, has recently become more popular in recommender systems. Along a different line, motivated by the insight that time, social, and network context play critical roles on users' media click behavior, in this paper we propose a novel recommendation system based on the generalized cluster-based bias model.

7.3 Mobile OSN Studies

To analyze social behaviors of mobile Twitter users, [39] identifies people using microblogging to talk about their daily activities and to seek or share information as well as analyzing the user intentions associated at a community level, showing how users with similar intentions connect with each other. In addition, a number of recent studies, such as [40], [41], address the problem of computing influence in Twitter-like networks and finding leaders whose tweets are influential. Our work does not aim at finding users who are influential directly. Instead, we exploit that different social friends have different impact on a user's activation on media usage or propogation. [42] proposes a tree-based algorithm to mine user-friend graphs to discover strong friends of a user. In contrast to our work, [42] does not consider how to utilize the social friendship structure to facilitate the information and content sharing among users, in particular, under a rich communication environment.

7.4 Geolocation-based Application

Recent geo-related research addresses that mobile location is very important for improving the quality of user experience (QoE). Primarily, [23] draws an ideological framework of location-based services on smartphones. In addition, [24] reveals some of the complexities involved in designing underlying technologies of collaborative location-based services. Furthermore, a number of recent studies emphasizes

JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015

the issue in geolocation-based networks to find high-impact leader objects. [25] leverages GPS location data on smartphones to estimate the likelihood using the place and then infers a more accurate user location. On the other hand, [43] aims to mine users' activity histories (GPS logs) and recommends activities suited for specific locations. This work mainly focuses on how to identify the current location of a user. While in our work, we predict user's action when located in a set of specific geolocations and time, which tackles another branch of location-based learning issues.

7.5 News Feed Streams and Recommendations

Recommender systems researchers have explored ways to use news feed algorithms to help users connect with the content they are most interested in on social media sites like Facebook. Several research teams have proposed different ways to use information about network ties, topic preferences, and characteristics of posts in the system to make recommendations. For example, Sharma et al. [44] proposed using information about content preferences from the Facebook profiles of users and their Facebook Friends to help generate recommendations for movies, television shows, and books. Items recommended by the algorithm variant that used Facebook Friends profile in formation received the highest number of views. In a followup study [45], they found that such social recommendations were more persuasive when they came from people who were close friends whose interests are known to the user. Rader et al. [46] explored how individuals make sense of the influence of algorithms, and how awareness of algorithmic curation may impact their interaction with these systems. The goal of Kaleido is orthogonal to them as it utilizes social relationship and emotion features toward a better recommender service.

8 CONCLUSION

We have presented Kaleido, a system for smart OSN media recommendation on smartphones. Building upon our cluster-based machine learning mechanism, Kaleido automatically learns relationships among various content and context impacts. Experiments with real Twitter traces from 16,952 people and an Android prototype show that Kaleido can achieve superior performance of a significant media recommendation accuracy while with minor additional data or energy consumption. Moreover, our design enables offloading of machine learning procedures to a cloud server, and achieves a speed-up of up to about 1,000 over local execution on smartphones. For future work, we will consider extending our system with a comprehensive implementation to support more media formats, e.g., video.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for valuable comments and all the Twidere data providers for their contributions in this paper.

This work is supported by Tsinghua University Initiative Scientific Research Program.

REFERENCES

- C. Wu, J. Jia, W. Zhu, X. Chen, B. Yang, and Y. Zhang, "Affective contextual mobile recommender system," in *Proceedings of the 24st* ACM international conference on Multimedia. ACM, 2016.
- [2] A. Geven, M. Tscheligi, and L. Noldus, "Measuring mobile emotions: Measuring the impossible?" in *MobileHCI*. ACM, 2009, p. 109.
- [3] M. Guerini and J. Staiano, "Deep feelings: A massive cross-lingual study on the relation between emotions and virality," in WWW. ACM, 2015, pp. 299–305.
- [4] J. Tang, Y. Zhang, J. Sun, J. Rao, W. Yu, Y. Chen, and A. M. Fong, "Quantitative study of individual emotional states in social networks," *IEEE Transactions on Affective Computing*, vol. 3, no. 2, pp. 132–144, 2012.
- [5] R. W. Picard and R. Picard, Affective computing. MIT press Cambridge, 1997, vol. 252.
- [6] L. Sayegh, W. P. Anthony, and P. L. Perrewe, "Managerial decisionmaking under crisis: The role of emotion in an intuitive decision process," *Human Resource Management Review*, vol. 14, no. 2, pp. 179–199, 2004.
- [7] N. Georgis, F. Li-De Lin, P. Hwang, L.-J. Chang, M. Kuroko, T. Miller, D. Nguyen, J. R. Zheng, M. A. Hanson, N. J. Colsey *et al.*, "System and method of selective media content access through a recommednation engine," Aug. 17 2007, uS Patent App. 11/840,814.
- [8] E. A. Vandewater, V. J. Rideout, E. A. Wartella, X. Huang, J. H. Lee, and M.-s. Shim, "Digital childhood: electronic media and technology use among infants, toddlers, and preschoolers," *Pediatrics*, vol. 119, no. 5, pp. e1006–e1015, 2007.
- [9] B. Sigurbjörnsson and R. Van Zwol, "Flickr tag recommendation based on collective knowledge," in *Proceedings of the 17th international conference on World Wide Web*. ACM, 2008, pp. 327–336.
- [10] M. Ye, P. Yin, and W.-C. Lee, "Location recommendation for location-based social networks," in *Proceedings of the 18th SIGSPA-TIAL international conference on advances in geographic information* systems. ACM, 2010, pp. 458–461.
- [11] M. Prangl, T. Szkaliczki, and H. Hellwagner, "A framework for utility-based multimedia adaptation," *IEEE Transactions on Circuits* and Systems for Video Technology, vol. 17, no. 6, pp. 719–728, 2007.
- [12] M.-H. Park, J.-H. Hong, and S.-B. Cho, "Location-based recommendation system using bayesian user's preference model in mobile devices," in *International Conference on Ubiquitous Intelligence* and Computing. Springer, 2007, pp. 1130–1139.
- [13] M. J. Pazzani and D. Billsus, "Content-based recommendation systems," in *The adaptive web*. Springer, 2007, pp. 325–341.
- [14] M. Jiang, P. Cui, R. Liu, Q. Yang, F. Wang, W. Zhu, and S. Yang, "Social contextual recommendation," in *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM, 2012, pp. 45–54.
- [15] "Twidere-twitter client. http://twidere.mariotaku.org."
- [16] X. Wang, J. Jia, J. Tang, B. Wu, L. Cai, and L. Xie, "Modeling emotion influence in image social networks," *IEEE Transactions on Affective Computing*, vol. 6, no. 3, pp. 286–297, 2015.
- [17] "Twitter rest apis. https://dev.twitter.com/rest/public."
- [18] P. Ekman, "An argument for basic emotions," Cognition & emotion, vol. 6, no. 3-4, pp. 169–200, 1992.
- [19] K. P. Murphy, Y. Weiss, and M. I. Jordan, "Loopy belief propagation for approximate inference: An empirical study," in *Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1999, pp. 467–475.
- [20] C. Grana, R. Vezzani, and R. Cucchiara, "Enhancing hsv histograms with achromatic points detection for video retrieval," in *Proceedings of the 6th ACM international conference on Image and video retrieval*. ACM, 2007, pp. 302–308.
- [21] M. E. Newman, "Modularity and community structure in networks," *Proceedings of the national academy of sciences*, vol. 103, no. 23, pp. 8577–8582, 2006.
- [22] "Google maps apis. https://developers.google.com/maps/."
- [23] J. Schiller and A. Voisard, *Location-based services*. Elsevier, 2004.
- [24] S. Benford, D. Rowland, M. Flintham, A. Drozd, R. Hull, J. Reid, J. Morrison, and K. Facer, "Life on the edge: supporting collaboration in location-based experiences," in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 2005, pp. 721–730.

JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015

- [25] H. Liu, Y. Zhang, Y. Zhou, D. Zhang, X. Fu, and K. Ramakrishnan, "Mining checkins from location-sharing services for clientindependent ip geolocation," in *INFOCOM*, 2014 Proceedings IEEE. IEEE, 2014, pp. 619–627.
- [26] S. C. Johnson, "Hierarchical clustering schemes," *Psychometrika*, vol. 32, no. 3, pp. 241–254, 1967.
- [27] P. S. Bradley and U. M. Fayyad, "Refining initial points for kmeans clustering." in *ICML*, vol. 98. Citeseer, 1998, pp. 91–99.
- [28] L. Hong, R. Bekkerman, J. Adler, and B. D. Davison, "Learning to rank social update streams," in *Proceedings of the 35th international* ACM SIGIR conference on Research and development in information retrieval. ACM, 2012, pp. 651–660.
- [29] L. Bottou, "Stochastic gradient descent tricks," in Neural Networks: Tricks of the Trade. Springer, 2012, pp. 421–436.
- [30] A. Bryan et al., "Metalink/http: Mirrors and hashes," Tech. Rep., 2011.
- [31] B. Rao and L. Minakakis, "Evolution of mobile location-based services," *Communications of the ACM*, vol. 46, no. 12, pp. 61–65, 2003.
- [32] B. Li, W. Xiong, W. Hu, and X. Ding, "Context-aware affective images classification based on bilayer sparse representation," in *Proceedings of the 20th ACM international conference on Multimedia*. ACM, 2012, pp. 721–724.
- [33] M. Cristani, A. Vinciarelli, C. Segalin, and A. Perina, "Unveiling the multimedia unconscious: Implicit cognitive processes and multimedia content analysis," in *Proceedings of the 21st ACM international conference on Multimedia*. ACM, 2013, pp. 213–222.
- [34] S. Siersdorfer, E. Minack, F. Deng, and J. Hare, "Analyzing and predicting sentiment of images on the social web," in *Proceedings* of the 18th ACM international conference on Multimedia. ACM, 2010, pp. 715–718.
- [35] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," *Communications* of the ACM, vol. 35, no. 12, pp. 61–70, 1992.
- [36] I. Guy, N. Zwerdling, I. Ronen, D. Carmel, and E. Uziel, "Social media recommendation based on people and tags," in *Proceedings* of the 33rd international ACM SIGIR conference on Research and development in information retrieval. ACM, 2010, pp. 194–201.
- [37] C. Wu, X. Chen, Y. Zhou, N. Li, X. Fu, and Y. Zhang, "Spice: Socially-driven learning-based mobile media prefetching," in *IN-FOCOM*, 2016 Proceedings IEEE. IEEE, 2016.
- [38] M. Claypool, P. Le, M. Wased, and D. Brown, "Implicit interest indicators," in *Proceedings of the 6th international conference on Intelligent user interfaces*. ACM, 2001, pp. 33–40.
- [39] A. Java, X. Song, T. Finin, and B. Tseng, "Why we twitter: understanding microblogging usage and communities," in *Proceedings* of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis. ACM, 2007, pp. 56–65.
- [40] M. Cha, H. Haddadi, F. Benevenuto, and P. K. Gummadi, "Measuring user influence in twitter: The million follower fallacy." *ICWSM*, vol. 10, no. 10-17, p. 30, 2010.
- [41] H. Kwak, C. Lee, H. Park, and S. Moon, "What is twitter, a social network or a news media?" in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 591–600.
- [42] J. J. Cameron, C. K.-S. Leung, and S. K. Tanbeer, "Finding strong groups of friends among friends in social networks," in *Dependable, Autonomic and Secure Computing (DASC)*, 2011 IEEE Ninth International Conference on. IEEE, 2011, pp. 824–831.
- [43] V. W. Zheng, Y. Zheng, X. Xie, and Q. Yang, "Collaborative location and activity recommendations with gps history data," in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 1029–1038.
- [44] A. Sharma and D. Cosley, "Network-centric recommendation: Personalization with and in social networks," in *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third International Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on.* IEEE, 2011, pp. 282–289.
- [45] A. Sharma, M. Gemici, and D. Cosley, "Friends, strangers, and the value of ego networks for recommendation." *ICWSM*, vol. 13, pp. 721–724, 2013.
- [46] E. Rader and R. Gray, "Understanding user beliefs about algorithmic curation in the facebook news feed," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 2015, pp. 173–182.



Chao Wu received the B.Eng. degree in software engineering from Southeast University, Nanjing, China, in 2012, and is currently pursuing the Ph.D. degree at the department of computer science and technology, Tsinghua University, Beijing, China. His research interests include mobile computing systems and networked systems with a focus on fog computing, operating system, wireless communication and applications.



Yaoxue Zhang received the B.S. degree from Northwest Institute of Telecommunication Engineering, China, and the Ph.D. degree in computer networking from Tohoku University, Japan in 1989. Then, he joined the Department of Computer Science, Tsinghua University, China. He was a visiting professor of Massachusetts Institute of Technology and University of Aizu, in 1995 and 1998, respectively. His major research interests include computer networking, operating systems, ubiquitous computing, big data driven

service optimisation, and mobile networks. He serves as an Associate Editor of Chinese Journal of Computers, Chinese Journal of Electronics, etc., the guest editor of IEEE Transactions on Computers. Currently, he is a fellow of the Chinese Academy of Engineering and a professor in computer science and technology in Tsinghua University, China. He is the president of Central South University, China.



Jia Jia is an associate professor in the Department of Computer Science and Technology, Tsinghua University, Beijing. She got her B.S. and Ph.D. degrees both in computer science and technology from Tsinghua University in 2003 and 2008 respectively. Her research interests are affective computing and computational speech perception.



Wenwu Zhu received the Ph.D. degree in electrical and computer engineering from the New York University Polytechnic School of Engineering, New York, NY, USA, in 1996. He is currently a Professor with the Computer Science Department of Tsinghua University, Beijing, China. His current research interests include multimedia cloud computing, social media computing, multimedia big data, and multimedia communications and networking. He has been serving as EiC for IEEE Transactions on Multimedia since Jan

1, 2017. He served on various editorial boards, including serving as Leading Editor of Computer Networks and Distributed Computing for the Journal of Computer Science and Technology; Guest Editor of the Proceedings of the IEEE, the IEEE Transactions on Circuits and Systems for Video Technology, and the IEEE Journal on Selected Areas in Communications; and Associate Editor for the IEEE Transactions on Mobile Computing, the IEEE Transactions on Multimedia, and the IEEE Transactions on Circuits and Systems for Video Technology. He was the recipient of the Best Paper Award at ACM Multimedia 2012, the Best Paper Award for the IEEE Transactions on Circuits and Systems for Video Technology in 2001, and four other international Best Paper Awards.

1536-1233 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.