Characterizing User Connections in Social Media through User-Shared Images

Ming Cheung, *Student Member, IEEE,* James She, *Member, IEEE* and Ning Wang, *Member, IEEE*

Abstract—Billions of user images, which are shared on social media, can be widely accessible by others due to their sharing nature. Using machine-generated labels to annotate those images is a reliable for user connections discovery on social networks. The machine-generated labels are obtained from encoded vectors using up-to-date image processing and computer vision techniques, such as convolution neural network. By analyzing 2 million user-shared images from 8 online social networks, a phenomenon is observed that the distribution of user similarity based on their shared images follows exponential functions. Users who share visually similar images are likely having follower/followee relationships, regardless of the origins and the content sharing mechanisms of a social network. This phenomenon is nicely formulated for a multimedia big data recommendation engine as an alternative to social graphs for recommendation. By utilizing the formulation of the distribution, it is proven the proposed engine can be 46% better than previous approaches in $F1$ score and achieves a comparable performance of friends-of-friends approach. To the best of our knowledge, this is the first attempt in related fields to characterize such phenomenon by massive user-shared images collected from real-world SNs, and then formulate into practical analytics engine for connection discovery.

Index Terms—big data system, user-shared images, connection, discovery, recommendation, social network analysis.

1 Introduction

User connection, which can be in many varieties such as users with similar interests and in the same physical location, is a fundamental attribute for online social networks (SNs). On social media, user connections could possibly be follower/followee-relationships in Twitter, and friendships on Facebook. As a result, social graphs (SGs) can be established by those follower/followee-relationships or friendships. By the definition of a connection, two users who have a connection in between them, are more inclined to share similar content [1]. Online service providers such as Twitter and Pinterest possess comprehensive data about user connections, which are used to improve the services they offer to users. However, accessing the SGs, originally provided by users, in many SNs today, is difficult and even unavailable. For example, trending mobile social networking applications, such as WhatsApp and WeChat, keep information exclusive to their related services or technologies. Some users may hide their SGs from being accessed by the public due to their privacy concerns. With such limited access to SGs, effective discoveries of user connections for novel applications are almost impossible for researchers, merchants, third-party service providers, and individuals. The aim of user connection discovery (UCD) is to discover hidden connections for applications such as follower/followee recommendations, gender identification and origin inference by analysing user-shared images. Two users with a connection are not necessary to be follower/followee, but they share similar online profiles, such as gender and location [2]. Billions of user-shared images are generated by individuals everyday in many SNs, and this particular form of user generated content is very widely accessible by others due to its nature for online social sharing. Hence, using computer vision/image processing techniques to discover user connections is a more accessible alternative than SGs for many applications. The images are first encoded into vectors, followed by a clustering to annotate the same unique machine-generated label on the images of the same cluster. User profiles are built based on the occurrence of the labels in bag-of-features tagging (BoFT) for UCD [1], by computing the user similarities with their profiles. It has been proven that the distribution of the similarity follows an exponential function.

The UCD is not affected by the techniques used to encode an image [3], such as scale-invariant feature transform (SIFT), or GIST [4], which are hand-engineered for object recognition tasks. Recently, convolutional neural network (CNN) is proven to be effective for object recognition [5] [6]. Motivated by humans' visual systems, in which neurons of CNN are arranged to respond to small regions, the structure of CNN comprises of several layers of non-linear feature extractors, which is handcrafted with learnable weights and biases from data. Different from traditional hand-engineered representations of images, CNN seeks to learn rich features automatically through a feature learning process and build a hierarchical representation from low level to high-level features. This work applies a pre-trained CNN [6] to encode images, by using the layer before the softmax layer as the encoded vector. It is interesting to investigate if the same conclusion can be found in the distribution of user similarity when another technique is used [7], and the improvement using CNN. The user profiles are calculated with the same procedures as BoFT [1]. When

- *MC and JS are with Department of Electronic and Computer Engineering, the Hong Kong University of Science and Technology, Hong Kong. NW is with Mathematical Institute and Oxford Internet Institute, the University of Oxford.*
  E-mail: {cpming,eejames}@ust.hk, wangn@maths.ox.ac.uk

Manuscript received August, 2017

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TBDATA.2017.2762719, IEEE Transactions on Big Data

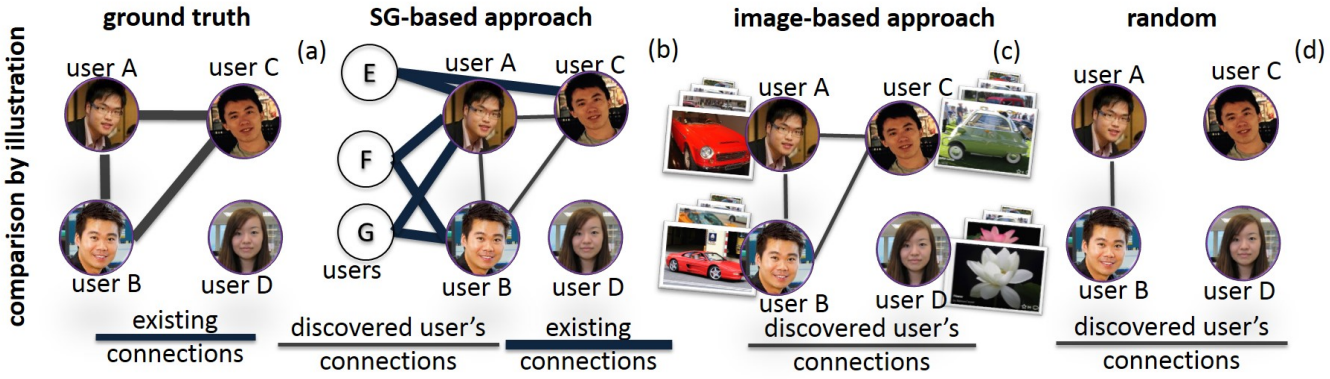

Fig. 1: Examples of discovered user connections on the SNs using different approaches with illustrations, in which existing and discovered user connections are connected with lines: (a) the ground truth, (b) the discovered user connections using their SGs, (c) the discovered user connections by a random approach, (d) the discovered user connections using shared images.

the SGs are only accessible by exclusive parties, using user-shared images to discover user connections is proved in [1] to be a more accessible and effective alternative to the traditional approach, such as using user annotated tags or SGs, in many SNs today. Fig. 1(a) to (d) show examples of discovered follower/followee relationships. Those relationships can be discovered through the similarity among their SGs and shared images.

A phenomenon of user-shared images on social media is proved and characterized that related users, who form a follower/followee relationship, share visually similar images, results in higher similarities [1]. Effective methods are therefore developed by more accessible user-shared images for UCD through the characterized phenomena. It is based on a simple computer vision approach for analysing images to calculate the user similarity from their shared images. However, there is no investigation on how the characterized phenomena can improve the recommendation, and it is not clear if other SNs with different origins and content sharing mechanisms are applicable to the same framework. As a result, main contributions of this research work are: 1) applying machine-generated labels to detect social signal from user-shared images, and proving that the users of a follower/followee pair share visually similar images, regardless of the origin country and content share mechanism of a SN; 2) proving that the user similarity distributions from their shared images follows an exponential function; 3) formulating the distributions for a multimedia big data recommendation system, and utilizing the formulation to obtain optimized cutoff similarity for a better follower/followee recommendation; 4) verifying the formulation with datasets from 8 SNs with 2M user-shared images with different origins and content sharing mechanisms.

This paper is organized as follows: section 2 discusses the related work, while section 3 introduces and summarizes the development of machine-generated labels. Section 4 conducts measurements on the user behaviours based on machine-generated labels, as well as introducing the model function and formulating the distributions. Section 5 proposes the UCD engine, with section 6 presents the experimental results. Section 7 concludes the papers.

2 Related Works

User connections are always an important information for a SN operator to improve their service relevance, such as follower/followee recommendations [8]. Recommendations are possible as those connections are not formed randomly, but following the power law distribution [9]. Similar users, such as users with the similar location [10], trajectory [11], mutual friends [12] and interest [13], tend to form follower/followee. Fig. 1 (a) and (b) show examples of ground truth and SG-based approach for follower/followee recommendations, on a network with 4 users. It is observed that many follower/followee relationships can be found. However, such observation is not shown when the relationships are recommended randomly. As a result, even without the access to SGs, UCD is also achievable using user common interests inferred from user inputs [14], [15] or user generated content [16], [17] and other personal information [13], [18], [19], [20], [21]. In particular, using user generated content is a common alternative as they refer user's interests, especially images, that are widely accessible by others and shared by many people. A common way to UCD is to utilize user annotated tags that are applied to their shared images [22], in which those tags are simple wordings created by users as a meta-data to describe their shared images [23]. These textual tags can reflect a certain level of user interests, and hence, user connections are possibly discovered or recommended by matching users with high similarities in their annotated tags. Due to the inconsistency of different wording and languages, or even levels of details, user annotated tags are therefore unreliable [24], [25] and cause a poor UCD or recommendation accuracy.

An emerging image-based approach [25] [26] applies computer vision or signal-processing techniques to understand user interests by producing machine annotated labels from recognized objects and in the context of user-shared images. Fig. 1 (c) shows an example of ground truth and image-based approach for follower/followee recommendations, on a network with 4 users. It is concluded that image-based approach can also tell follower/followee relationships, through their shared images. Given a set of images, the images are first encoded into a vector with R

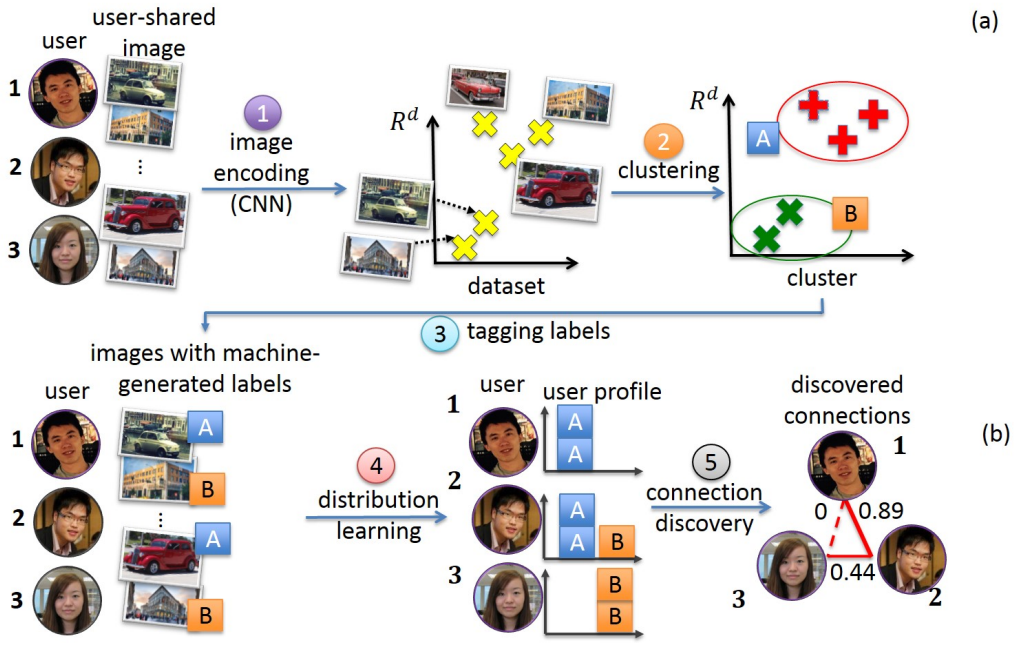

Fig. 2: Flow Chart of machine-generated labels for user connection discovery.

dimensions by some image processing or computer vision techniques. Feature-based [27] [28] and colour-based techniques [29] are proven to be useful for encoding images [3]. If a colour-based technique is used, 2 images with the same machine-generated labels are with similar colour. One of the possible approaches is convolutional neural networks for image encoding [7]. As a result, 2 images with the same label are likely to be similar objects. User profiles are then built using the occurrence of those labels on their shared images, and their image similarity can be calculated accordingly. It is proven that 2 users with a higher value of image similarity are more likely to be follower/followee [1] on Skyrock and 163, where the distribution of the image similarity follow an exponential function. However, there is no investigation how the characterized phenomena can improve the recommendation, and it is not clear if other encoding techniques and SNs are applicable to the same framework. Based on this observation, the major differences of this paper to the previous one [1] are: 1) scraping and studying 6 more SNs: in which there are 2M user-shared images studied, and conclude an exponential function can well model the similarity distributions of the 8 SNs; 2) enhancing the mathematical formulation, and hence proposing a new method to utilize the modeled distribution for a better recommendation; 3) introducing and studying $\widehat{S_{i,j}}$, the cutoff similarity, in which the $F1$ score is 46% higher than the one in [1]; 4) proposing a method to obtain $\widehat{S_{i,j}}^*$, the optimized $\widehat{S_{i,j}}$ to maximize $F1$; 5) adopting the state-of-the-art technique, CNN, for image encoding, and proved that it has out-performed the previous works;

3 User Connection Discovery with Machine-generated Labels

This section introduces and summarizes the recent development of UCD with machine-generated labels. The first part introduces the framework for discovering user connections from user-shared images through machine-generated labels, which is followed by how the connections can be applied in follower/followee recommendation. The way to discover connections from the user profiles is also discussed.

3.1 Machine-generated Label and User Profile

Machine-generated labels are used to represent visual features in a user-shared images. Fig. 2 shows the flow. The user-shared images are first encoded. In this paper, instead of using a pre-trained classifier and the class label (e.g., 'car' and 'cat'), an unsupervised approach for clustering, is used for generating labels. Many image encoding techniques, such as SIFT [28], GIST [4], and CNN [6] are also applicable for UCD [3]. In particular, CNN is adopted in this work. In this work, the last layer of the CNN from [6] was used as the encoded vector. The encoded vectors are then clustered into K clusters using K-means, and images in the same cluster are assigned with the same unique label, as shown in step 2 and 3 of Fig. 2. The occurrence of k-th labels among the shared images of a user i, $l_{i,k}$, is counted from the clustering assignment. The connections are discovered by user profile (step 5 of Fig. 2), is hence defined as follows:

$$L_i = \{l_{i,1}, ...l_{i,k}, ...l_{i,K}\} \tag{1}$$

where K is the total number of possible labels, which is the number of clusters during the clustering process. UCD is based on the image similarity of two users (step 3 of Fig. 2), i and j, by a popular formula, cosine similarity:

$$S_{i,j} = S(L_i, L_j) = \frac{L_i \cdot L_j}{||L_i|| \cdot ||L_j||} \tag{2}$$

where L_i and L_j are the profile of users i and j, respectively.

Other similarity calculations will also work. Note that as cosine similarity is used in this work, L_i does not have to be normalized, as $||L_i||$ and $||L_j||$ are in the denominator of Eq. 2. These connections can be used to discovery online friendships and identify the user gender, and it is proven that two users with a higher similarity are more likely be a pair of follower/followee [1]. In follower/followee recommendation, the goal is to recommend a set of users with the highest probability with a user. Other approaches, such as matrix factorization [30], can be applied for recommendation. The coming section discusses how follower/followee recommendation can be conducted with the similarity.

3.2 User Profile and Recommendation

Follower/followee recommendation is one of the most popular applications on social media. The goal is to recommend a user to another user that they are likely to become follower/followee, such as, they share similar interests. A user pair, users i and j, can be considered as two classes; one is a related pair while the other one is unrelated:

$$C_{i,j} = \begin{cases} 1 & \text{if two users are related as follower/followee} \\ 0 & \text{if otherwise,} \end{cases} \tag{3}$$

where $C_{i,j} = 1$ is the class of related pairs, in which a user pair is a pair of follower/followee, and $C_{i,j} = 0$ is the class of unrelated pairs, in which a user pair does not have an online friendship. Follower/followee recommendation should be made based on the probability that users i and j is a related pair given the image similarity of user i and j is s, where s is a value range from 0 to 1, $P(C_{i,j} = 1|S_{i,j} = s)$. The recommendation should be made from the highest $P(C_{i,j} = 1|S_{i,j} = s)$ to the lowest. If $P(C_{i,j} = 1|S_{i,j} = s)$ is increasing, i.e., 2 users with a higher $S_{i,j}$ are always more likely to be friends, the problem can be formulated as [1]:

$$U^*_{i,J} = \arg\max_{U_{i,J}} \prod_{j=1}^{J} S_{i,j},\ where\ j \subset U_{i,J} \tag{4}$$

where $U_{i,J}$ is the list of users to be recommended to user i, given the image similarities of all users and the list of users that are most likely to be related pairs with user i. By selecting the top J users with the highest $S_{i,j}$ with user i, Eq. 4 can be maximized, i.e., the recommendations are most likely to be accepted. Hence, the list of recommended users are those with the higher image similarity to user i. However, it is not clear whether $P(C_{i,j} = 1|S_{i,j} = s)$ is increasing on different SNs, especially SNs from different origins and content sharing mechanisms. As well, even the recommendations have the highest probability to be accepted, some recommendation could be noisy, as they may have very little probability to be accepted. Hence, it is desirable to improve Eq. 4 such that those recommendations with a very low probability to be accepted are filtered. The next section introduces the 8 datasets and how they are collected, conduct measurements on the 8 SNs and model the distribution.

4 Measurements on User-Shared Images

The goal of this section is to prove and observe the characteristics of user-shared images. The measurement is based on 2 million images and 20 million connections of following/followee from 8 real SNs originating from the East and the West with 2 different content sharing mechanisms. The first part introduces the collected massive data, study the increasing trend of $P(C_{i,j} = 1|S_{i,j} = s)$ and the second part characterizes the images.

4.1 Dataset from 8 SNs

Those images are originally shared by users in the dataset. The dataset, collected and organized from 2013 to 2014, includes most of the public information, including user IDs, follower/followee relationships and shared images. In order to maintain the randomness of the dataset, the users are selected randomly from a larger set of users. The personal information such as names and locations are anonymized in the analysis. Table 1 shows the statistics of these datasets. The total relationship is follower/followee relationship involving a selected user and any user. Existing relationships are those follower/followee relationships among selected users only. The datasets of Flickr, Twitter, Tencent Weibo, Skyrock and 163 Weibo are collected through their official API . No authorisation is needed in these SNs for collecting the user connections and the user-shared images. Pinterest, Digu and Duitang do not provide any official API for data collection and the data of is collected by HTTP requests.

4.2 Increasing Trend of $P(C_{i,j} = 1|S_{i,j} = s)$

The connection between 2 users indicates that they share some similarity. Table 2 shows the average $S_{i,j}$ for related and all pairs on different SNs. It is observed that related pairs have a higher $S_{i,j}$ on all SNs. The table also shows the standard deviation of all pairs, σ, the z-score, z. The z is the values of a null hypothesis that related and all users have a smaller or equal similarity, and an alternative hypothesis that related user has a higher $S_{i,j}$ than unrelated users. It is observed that the z on all SNs are large. If the corresponding p of each value of z is calculated, it tends to be zeros on all SNs. This observation proves that related pairs always have higher similarities, regardless of the origin and the content sharing mechanisms, of the SN. Note that the value of average $S_{i,j}$ of unrelated and all pairs are close, as the number of unrelated pairs is much larger than related pairs. It is also interesting to investigate whether the probability that users i and j are related, $P(C = 1|S_{i,j} = s)$, is an

TABLE 2: Average $S_{i,j}$ for related, all, and unrelated pairs, and the corresponding σ, z and p

	related	unrelated	all	σ	z
Flickr	0.276	0.0793	0.0803	0.0814	90.6
Pinterest	0.230	0.162	0.162	0.110	21.4
Twitter	0.183	0.116	0.116	0.0886	27.8
Skyrock	0.106	0.0329	0.0339	0.0693	89.0
Duitang	0.230	0.145	0.150	0.122	16.6
Digu	0.150	0.112	0.113	0.0903	11.8
163 Weibo	0.112	0.0370	0.0376	0.0876	35.4
Tencent Weibo	0.0749	0.0584	0.0585	0.0863	8.23
Average	0.136	0.0707	0.0703	0.0920	37.6

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TBDATA.2017.2762719, IEEE Transactions on Big Data

Fig. 3: User interfaces of SNs from the West and the East, with different content-sharing mechanisms.

TABLE 1: The datasets from 8 real-world SNs measurement

origins	social network	user-shared images	total relationships	existing relationships	network density	users	*sharing mechanism
US	Flickr	201,006	92,056	1418	0.48%	562	image-oriented
US	Pinterest	324,369	483,798	1212	0.35%	600	image-oriented
US	Twitter	150,696	14,487,045	1364	0.65%	462	general
France	Skyrock	176,547	2,439,058	7348	1.41%	722	general
China	Duitang	596,342	21,019	654	0.43%	391	image-oriented
China	Digu	148,337	43,141	798	0.49%	404	image-oriented
China	163 Weibo	187,491	850,487	1732	0.71%	493	general
China	Tencent Weibo	490,624	1,091,492	2176	0.86%	503	general
	Total	2,275,412	19,508,096				

**Note: Users of a SN with general sharing mechanism can share images/videos/texts/etc. Image-oriented SNs are mainly with images*

increasing function. If so, it implies that a user pair is more likely to be related with a higher value of $S_{i,j}$, and Eq. 4 is applicable.

It is desirable to directly measure $P(C = 1|S_{i,j} = s)$ from the dataset. Unfortunately, $S_{i,j}$ is a continuous variable from 0 to 1, it is not possible to obtain enough user pairs with exact $S_{i,j} = s$. As a result, the value of $P(C = 1|S_{i,j} = s)$ cannot be measured directly from data measurements. Alternatively, the method of density estimation [31] is applied to estimate the probability density function (PDF) of related pairs, $f(S_{i,j} = s|C = 1)$, and all pairs, $f(S_{i,j} = s)$, from the observed discrete data with a kernel estimator. The estimator requires a parameter, window width. A larger value of window width can average out the noise, while a small window width can preserve the local features. Fig. 4 shows the estimated PDF of all pair, $f(S_{i,j} = s)$, on a real SN, Skyrock. The solid line is the estimated $f(S_{i,j} = s)$, and the circles are the observed $S_{i,j} = s$ of pairs. It is observed that the range with more observed $S_{i,j} = s$, $f(S_{i,j} = s)$ has a higher value, as shown in the left part of the figure. While $f(S_{i,j} = s)$ is lower when there are fewer observed $S_{i,j}$, as shown in the right part of the figure. It is a good estimation of the PDF.

By using a Bayesian approach, $P(C = 1|S_{i,j} = s)$ can be computed by:

$$P(C = 1|S_{i,j} = s) = \frac{P(S_{i,j} = s|C = 1)P(C = 1)}{P(S_{i,j} = s)} \quad (5)$$

$P(C_{i,j} = 1)$ is the probability that users i and j are a related pair, which is number of related pairs divided by number of all pairs, and equals to the network density.

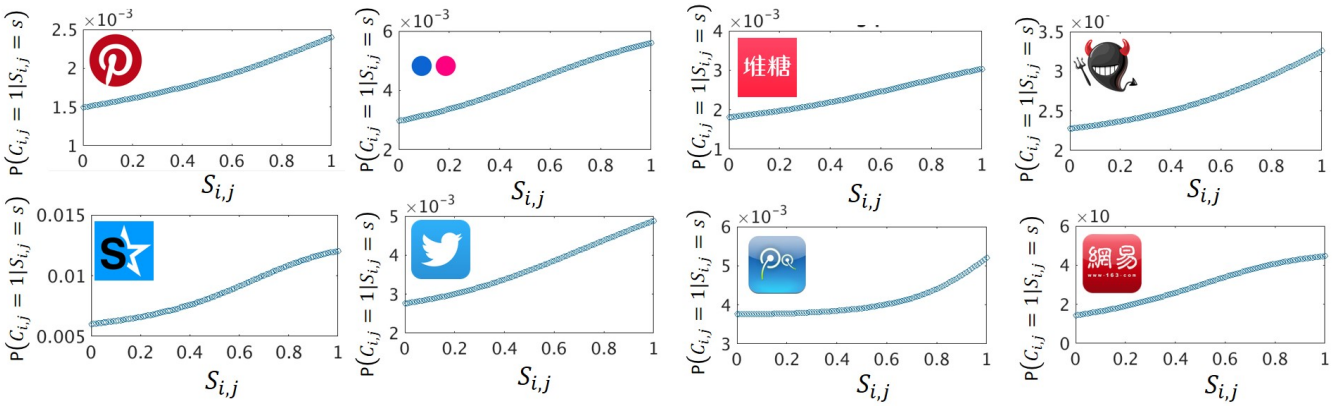

Fig. 5: $P(C = 1|S_{i,j} = s)$ on the 8 SNs, they are increasing functions that higher $S_{i,j}$ gives higher $P(C = 1|S_{i,j} = s)$

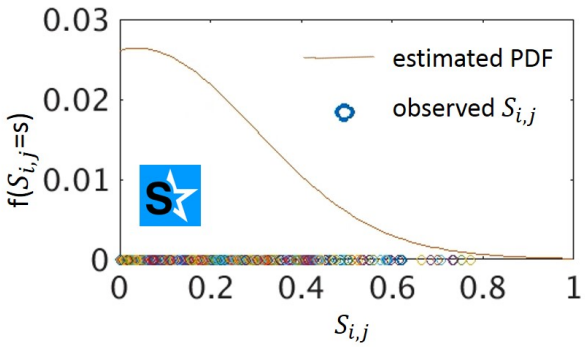

Fig. 4: Estimated $P(S_{i,j} = s)$ using density estimation

It is a constant number that can be obtained from data measurements. Hence, $P(C = 1|S_{i,j} = s)$ can be estimated by the PDFs or related and all pairs:

$$P(C = 1|S_{i,j} = s) = \frac{f(S_{i,j} = s|C = 1)P(C = 1)}{f(S_{i,j} = s)} \quad (6)$$

It is interesting to investigate if $P(C = 1|S_{i,j} = s)$ is an increasing function. Fig. 5 show the measurement of $P(C = 1|S_{i,j} = s)$ on the 8 SNs. It is observed that they are all increasing functions, and follower/followee recommendation can be made by recommending users with highest $S_{i,j}$ with user i. Besides $P(C = 1|S_{i,j} = s)$, it is also interesting to investigate how to utilize the distribution through modeling. The coming section studies and models the distributions of the 8 SNs for recommendation.

4.3 Characterizing the Probability that Users Are Related by Their Shared Images

Although the estimator can show the trend of $P(C = 1|S_{i,j} = s)$, but it requires a parameter, window width. If the window is small, spurious noise may appear, while a large window width may mask the of the distribution. It is desirables to model these distributions, from the cumulative distribution function (CDF) from the observed data to compute $P(C = 1|S_{i,j} = s)$. This section first introduces a function that can apply to model the CDF. The cross in Fig. 6 shows the measurements of CDFs of related and all pairs, for a $S_{i,j}$, $F(S_{i,j}|C_{i,j} = 1)$ and $F(S_{i,j})$, from the 8 SNs, respectively. It can be concluded that the CDF of related and all pairs of all SNs follow similar trends. As observed in [1], the PDF of the distributions of related and all pairs, $f(S_{i,j} = s|C = 1)$ and $f(S_{i,j} = s)$, follow exponential functions. Hence, the CDFs can be modeled by the integration of an exponential function:

$$G(S_{i,j} = s|\lambda) = \frac{1 - e^{-\lambda s}}{1 - e^{-\lambda}}, \text{for } s \subset [0, 1] \quad (7)$$

where λ is a real number. The details of $G(S_{i,j} = 1|\lambda)$ can be found in the appendix A. Note that $G(S_{i,j} = s|\lambda)$ is an increasing function, as well, $G(S_{i,j} = 0|\lambda)$ and $G(S_{i,j} = 1|\lambda)$ equal to 0 and 1, respectively. If the value of λ is positive, $G(S_{i,j} = s|\lambda)$ grows fast at the beginning, and grow slowly when s is close to 1. Also, a larger value of λ will result in a faster growth when s is small. If the value of λ is negative, $G(S_{i,j} = s|\lambda)$ grows slowly at the beginning, and grows fast when s is close to 1. As the distribution of related and all pairs are different, there are 2λs, λ_r and λ_a, for the distributions of related and all pairs, respectively. As observed in Fig. 6, both λ_r and λ_a are all positive numbers.

In follower/followee recommendation, a list of J users, $U_{i,j}$, is recommended to user i, given the $S_{i,j} = s$ and the list of users that are most likely to be related pairs with user i. By using the definition of a CDF [32], Eq. 5 becomes:

$$\begin{aligned}
&P(C_{i,j} = 1|S_{i,j} = s) \\
&= \lim_{\Delta s \to 0} \frac{\frac{[F(S_{i,j}=s+\Delta s|C_{i,j}=1) - F(S_{i,j}=s|C_{i,j}=1)]}{\Delta s} P(C_{i,j}=1)}{\frac{[F(S_{i,j}=s+\Delta s) - F(S_{i,j}=s)]}{\Delta s}} \\
&= \lim_{\Delta s \to 0} \frac{[F(S_{i,j}=s+\Delta s|C_{i,j}=1) - F(S_{i,j}=s|C_{i,j}=1)]P(C_{i,j}=1)}{[F(S_{i,j}=s+\Delta s) - F(S_{i,j}=s)]} \\
&= \lim_{\Delta s \to 0} \frac{[G(S_{i,j} = s + \Delta s|\lambda_r) - G(S_{i,j} = s|\lambda_r)]P(C_{i,j} = 1)}{[G(S_{i,j} = s + \Delta s|\lambda_a) - G(S_{i,j} = s|\lambda_a)]}
\end{aligned} \quad (8)$$

Hence, the probability of $P(C_{i,j} = 1|S_{i,j} = s)$ can be computed by:

$$P(C_{i,j} = 1|S_{i,j} = s) = \frac{\lambda_r(1 - e^{-\lambda_a})}{\lambda_a(1 - e^{-\lambda_r})} e^{(\lambda_a - \lambda_r)s} P(C_{i,j} = 1) \quad (9)$$

The detailed steps for Eq. 9 can be found in the appendix B.

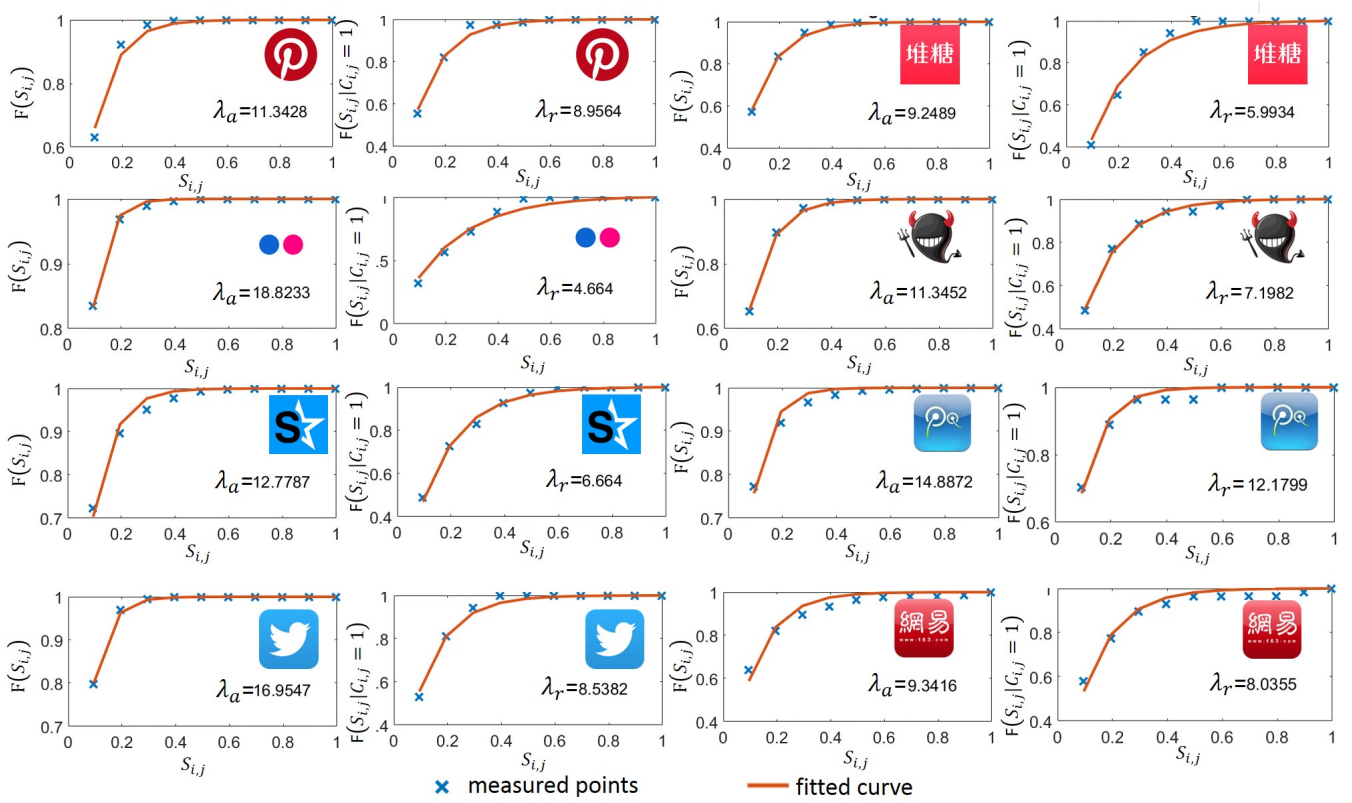

Fig. 6: Distribution of $S_{i,j}$ among related and all pairs on the 8 SNs. All the fitted curves follow the measured points well.

5 Proposed Recommendation Engine using User-Shared Images

This section introduces the flow and formulation of how recommendation can be made with discovered connections. This is a 4-stage (stages A to D) engine as shown in Fig. 7, extending the one in [1] with better formulation and methods to utilize modeled distributions for recommendation. The first part is image collection, followed by connection discovery using CNN. The third part focuses on modeling distribution. The fourth part focuses on how to recommend follower/followees based on the discovered connections and the image similarity distribution. The stages are introduced one by one below.

5.1 Image Collection from a SN

The proposed engine carries out data collection as shown in step A of Fig. 7, which shows the process to collect user generated images from social media applications, such as the 8 SNs in the datasets. The images can be provided by the operators of the social media and mobile applications or collected through the API of the SNs. The user generated images can be shared in various forms, such as posted images on social media or images shared through instant messaging applications. On SNs such as on the 8 SNs, user generated images are those images shared by users. This process is ongoing, which means that user-shared images are collected continuously.

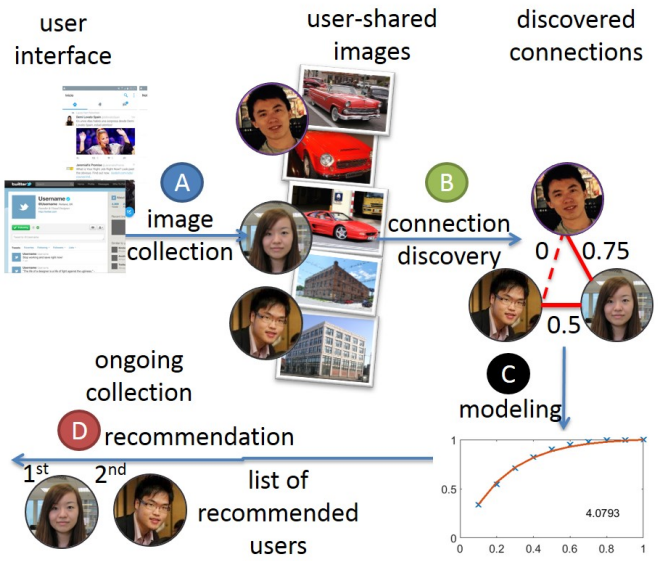

Fig. 7: Flow of the proposed recommendation engine: (a) image collection from social media; (b) connection discovery by collected images; (c) distribution modeling; (d) recommendation by discovered connections.

5.2 Connection Discovery from Images

The objective is to discover connections by annotating user generated images with machine annotated labels, as shown in step B of Fig. 7. The proposed engine applies a computer

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TBDATA.2017.2762719, IEEE Transactions on Big Data

vision approach to give a label to user generated images, which is not affected by the language, culture or other characteristics of the user who shares the image, but is based on the image's visual appearance only. The accuracy of the user generated tags is unreliable, sometimes even unavailable, and the performance of connection discovery is affected. The proposed engine applies CNN to annotate user generated images with machine annotated labels, called CNN labels. The set of user-shared images of user i is processed by the proposed engine, and a set of CNN labels, L_i, is generated to represent user i. As discussed, millions of images are generated every day, so an engine that can process big data with scalable storage design is needed for collecting and processing these user-shared images, such as a cloud-assisted engine to handle profile learning and similarity calculation [33] for a scalable engine. The feature vectors are first to split into multiple blocks in the Hadoop Distributed File System (HDFS) and distributed to virtual machines (VMs) for the k-means clustering process. Each VM is in charge of computing the distribution of different labels for several users, and the image similarity is also calculated in a distributed way.

5.3 Distribution Learning

The goal of the learning is to model the 2 distributions from measurements for follower/followee recommendation. This subsection computes the value of the distribution (probability density function) of the 8 SNs by estimating λ_a and λ_r in Eq. 7. Instead of computing the probability density function from data, the CDF is first estimated, by counting the number of related pairs and all pairs that are smaller than a given s. The parameters of the distributions can be estimated accordingly using algorithms such as Trust-Region algorithm [34], using Eq. 7 to estimate λ_a and λ_r accordingly. The parameters are then stored and used in recommendation.

5.4 Recommendation

The proposed engine can work for many types of recommendations, such as gender and origin [2]. Among those recommendations, follower/followee recommendation is one of the most popular applications on social media. The probability that two users are a related pair, or $C_{i,j} = 1$, given the image similarity of user i and j, $P(C_{i,j} = 1|S_{i,j} = s)$ can be calculated by Eq. 2 and Eq. 9 based on L_i and L_j. As shown in Fig. 5, $P(C_{i,j} = 1|S_{i,j} = s)$ is an increasing function, one can conclude that a higher $S_{i,j}$ implies a higher chance that a pair of users are friends. The recommendations to user i are the users with the highest $S_{i,j} = s$ with user i. Follower/followee recommendation should be made based on $P(C_{i,j} = 1|S_{i,j} = s)$, from the highest to the lowest. On social media, $P(C_{i,j} = 1|S_{i,j} = s)$ will become small when $S_{i,j}$ is less than some value of s, which are very unlikely to be related with user i. It would be better not to recommend those users, as they are noise to user i. Hence, the cutoff of similarity, $\widehat{S_{i,j}}$ is defined. For an engine that recommends a list of users to user i, Eq. 4 can be expressed as:

$$U_{i,J} = \arg\max_{U_{i,J}} \prod_{j=1}^{J} S_{i,j},\ where\ j \subset U_{i,J} \quad \text{subject to } S_{i,j} > \widehat{S_{i,j}} \tag{10}$$

where J and $U_{i,J}$ are the number of users and the list of users to be recommended. Note that J is not a constant, it may vary from one user to another, as the similarity of some users may be too low to be recommended. The physical meaning of Eq. 10 is to recommend users only if they have $S_{i,j}$ greater than $\widehat{S_{i,j}}$, i.e., users that are more likely to be related with user i than random. In follower/followee recommendation, there are 2 common measurements on the performance. The first one is precision (p), which is the percentage of recommendation are related with user i, and the second one is recall (r), which is the percentage of related users are covered in the recommendation. By using a higher $\widehat{S_{i,j}}$, the J recommended users are more likely to be accepted by user i (higher precision), but a fewer number of users can be recommended (lower recall). The value of p can be estimated by computing the expected value of accepted rated as follow:

$$p(\widehat{S_{i,j}}) = \frac{\sum_{j=1}^{J} P(C_{i,j} = 1|S_{i,j} = s)}{J} \quad \text{subject to } S_{i,j} > \widehat{S_{i,j}} \tag{11}$$

Similarly, r can be estimated by:

$$r(\widehat{S_{i,j}}) = \frac{\sum_{j=1}^{J} P(C_{i,j} = 1|S_{i,j} = s)}{N_{C=1}^{(p)}} \quad \text{subject to } S_{i,j} > \widehat{S_{i,j}} \tag{12}$$

where $N_{C=1}^{(p)}$ is the number of related pairs among users. The results can be evaluated by a common metric of prediction performance, $F1$ [35], for users with the highest similarities, as in step 2 of Fig. 8. $F1$ can be calculated as:

$$F1(\widehat{S_{i,j}}) = \frac{2p(\widehat{S_{i,j}})r(\widehat{S_{i,j}})}{p(\widehat{S_{i,j}}) + r(\widehat{S_{i,j}})} = \frac{2\sum_{j=1}^{J} P(C_{i,j} = 1|S_{i,j} = s)}{N_{C=1}^{(p)} + J} \quad \text{subject to } S_{i,j} > \widehat{S_{i,j}} \tag{13}$$

$F1$ gives a good balance between precision and recall. On one hand, if only confidence recommendations are made (e.g., pairs with a high $S_{i,j}$), a high precision can be achieved with a low recall. On the other hand, a high recall rate (e.g., recommend all users) leads to a low precision. It is desirable to find the optimized $\widehat{S_{i,j}}$ to obtain the best $F1$. Based on

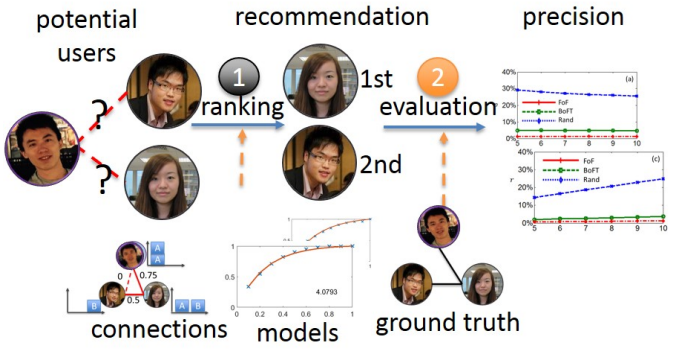

Fig. 8: Two steps in experimental setup: 1) ranking by $S_{i,j}$ of discovered connections, 2) evaluation by ground truth.

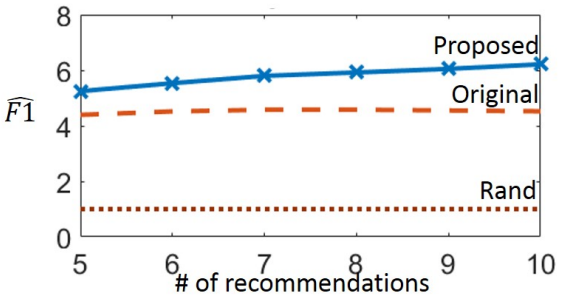

Fig. 9: $\tilde{F1}$ of follower/followee recommendations with discovered connections using different methods, for 5 to 10 recommendations.

the model, $\widehat{S_{i,j}}^*$ can be estimated by:

$$\begin{aligned}\widehat{S_{i,j}}^* &= \arg\max_{\widehat{S_{i,j}}} F1(\widehat{S_{i,j}}) \\ &= \arg\max_{\widehat{S_{i,j}}} \frac{2\sum_{j=1}^{J} P(C_{i,j}=1|S_{i,j}=s)}{N_{C=1}^{(p)}+J} \\ &= \arg\max_{\widehat{S_{i,j}}} \frac{\sum_{j=1}^{J} e^{(\lambda_a-\lambda_r)s}}{N_{C=1}^{(p)}+J} \\ &\text{subject to } S_{i,j} > \widehat{S_{i,j}}\end{aligned} \tag{14}$$

The advantage of the proposed engine is that once λ_a and λ_f are estimated, Eq. 14 can be computed directly, without the needs of the social graph. It is particularly useful when there are millions of users to recommend. The recommended users can be sent to the social media and mobile applications when a list of follower/followee recommendations is needed for a given user. Hence, Eq. 15 can be optimized as:

$$\begin{aligned}U_{i,J}^* &= \arg\max_{U_{i,J}} \prod_{j=1}^{J} S_{i,j},\ where\ j \subset U_{i,J} \\ &\text{subject to } S_{i,j} > \widehat{S_{i,j}}^*\end{aligned} \tag{15}$$

6 Experimental Results

This section introduces how the experiment is conducted, followed by the experimental results. The discussions on 2 parameters, K and $\widehat{S_{i,j}}$, conclude this section.

6.1 Experimental Setup for Recommendation

Based on the observation that user pairs with a higher image similarity are more likely to be follower/followee, discovered connections can be evaluated as a follower/followee recommendation engine using $S_{i,j}$. Fig. 8 shows the experiment setup for the evaluation with user-shared images from the 8 SNs. As in step 1 of Fig. 8 (a), the user-shared images are analyzed using CNN, as shown in Fig. 2 (a), and users are represented by user profiles of the distribution of CNN labels that the user has, as shown in Fig. 2 (b). Connections are discovered by computing the pairwise $S_{i,j}$ using Eq. 2 by analysing the user profiles. The distribution of image similarity is then learned accordingly. The list of users to be recommended to user i, are ranked by $S_{i,j}$ in the discovered connections. According to Eq. 4, the set of J users is most likely to be follower/followee of user i if they are users with the highest similarities. As a result, the set of users with the highest similarities are recommended to user i, in which the recommended users have higher $S_{i,j}$ than $\widehat{S_{i,j}}$. The result is measured in $F1$. Three other methods are also implemented for comparative evaluation. The first method is friends-of-friends (FoF), in which user connections are discovered by its similarity with their SGs as an achievable upper bound when the difficult and limited access to SGs are fully available. The second one is the same approach in [1], in which user are recommended by ranking the similarity, without using the knowledge of distributions. The third method is a random method (Rand), in which user connections are discovered based on a randomly assigned similarity value to all possible user pairs. This serves as a baseline, or the lower bound, due to its simplicity.

6.2 Results of Follower/followee Recommendation

The number of recommendations is set to be 5 to 10, to simulate a normal recommendation engine; however, the same trend can be found even when a smaller or a bigger number of recommendations is used. When $\widehat{S_{i,j}}$ is set to be high, only confident recommendations are made (i.e., user pairs with a high $S_{i,j}$), a high p can be achieved, but r will be small as only a few recommendations can be made. $F1$ can make measurements balancing p and r. Fig. 9 shows the $F1$ from 5 to 10 recommendations on different SNs. As a SN with a higher density gives a higher $F1$, even with Rand approach. $\widetilde{F1}$ is the normalized $F1$ which is divided by the $F1$ of Rand on that SN. It makes sure that a high-density SN will not be overweighted in the result, as they have a high $F1$. It is observed that the improvement of the proposed engine, on average, there is a 28% increase in $F1$ when using the best $\widehat{S_{i,j}}$. More discussion on $\widehat{S_{i,j}}$ can be found at the end of this section.

It is also interesting to investigate the differences between eastern and western SNs, as well as image-oriented and general SNs. Fig. 10 (a) and (b) show the comparison of $\widetilde{F1}$ on between eastern and western, as well as image-oriented and general SNs, respectively. From Fig. 10 (a), FoF gives the best performance. One of the reasons is that FoF makes use of the relationships besides those among users, and hence it gives a better performance, and serves

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TBDATA.2017.2762719, IEEE Transactions on Big Data

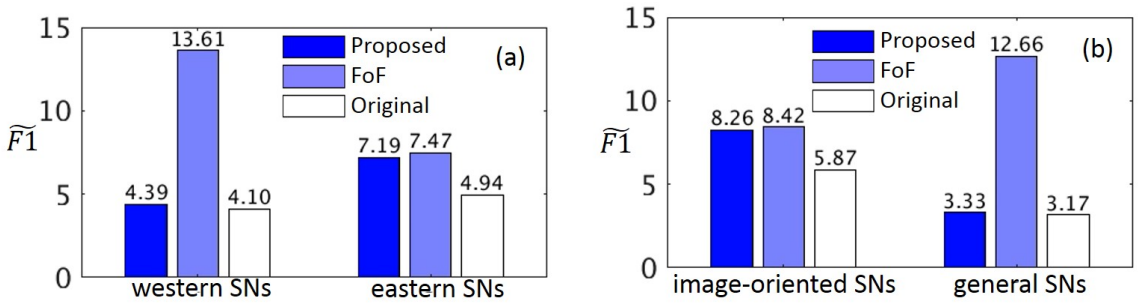

Fig. 10: The precision of follower/followee recommendations with connection discovery, for 5 to 10 recommendations, between: (a) eastern and western SNs; (b) image-oriented and general SNs.

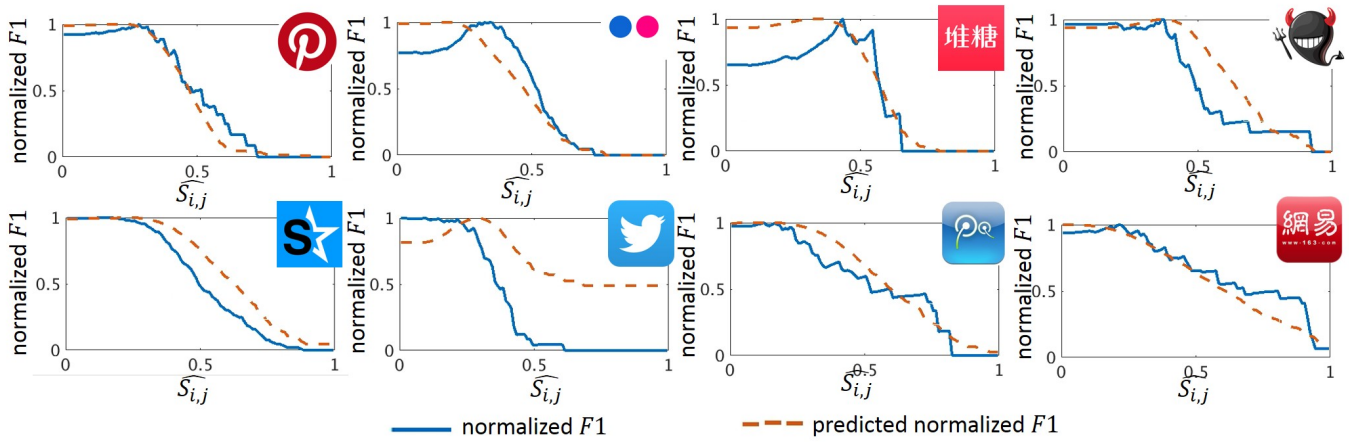

Fig. 11: Comparison of follower/followee recommendations with different values of $\widehat{S_{i,j}}$, for 5 to 10 recommendations

as the upper bound. It is observed that the proposed engine performs better in eastern SNs, in which it achieves 96.1% of FoF performance. It is also 46% better than original approach. On the other hand, the proposed engine is 6.94% better than the original approach on western SNs. A similar observation can be found in Fig. 10 (b). The proposed engine performs better in image-oriented SNs, in which it achieves 98.1% of FoF performance. It is concluded that the proposed engine preforms better on Eastern and image-oriented SNs, as the value is much closer to the upper bound, FoF. The reason for a better performance in eastern and image-oriented SNs is that the proposed engine on Daitung and Digu has a comparable performance as FoF. Daitung and Digu have the smallest number of total relationships, as shown in Table 1. The users on those SNs tends to have fewer friends, as a result the SGs are less useful. The proposed engine can achieve a better performance, when there are many missing connections, or the SGs are not accessible.

6.3 Fitting Quality of Distributions

The subsection shows the quality of the fitting of distribution, and the result are shown in Fig. 6. It is observed that they are good fits of the original functions. The fitting can be evaluated by the root-mean-square error (RMSE):

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{B}(y_i - \hat{y}_i)^2)}{B}} \tag{16}$$

TABLE 3: RMSE of the fitting of related and all pairs

SN	all pairs	related pairs
Flickr	0.00191	0.0326
Pinterest	0.00942	0.00778
Twitter	0.00118	0.0110
Skyrock	0.00918	0.00963
Duitang	0.00505	0.0218
Digu	0.00208	0.00860
163 Weibo	0.0249	0.0231
Tencent Weibo	0.00886	0.0325
Average	**0.00782**	**0.0184**

where B is the number of points takes, and y_i and $\hat{y}_i$ are the measured values from data and predicted values from the model. Table 3 shows the RMSE of the fitting on different SNs. It is observed that they are relatively small in all SNs. The reason is that the number of all pairs is much greater the number of related pairs (1% of all pairs), and the errors reduced when more samples (related pairs) are available.

6.4 Results of $\widehat{S_{i,j}}$

In Tencent, the performance of recommendation of $\widehat{S_{i,j}} = 0.325$ is the best, while $\widehat{S_{i,j}} = 0.455$ on Digu. It is interesting to study how to study the effect of $\widehat{S_{i,j}} = 0.325$ on different SNs, and how good the proposed engine can estimate $\widehat{S_{i,j}}^*$. Fig. 11 shows the results of different values of $\widehat{S_{i,j}}$ and the corresponding actual (from measurement) as well as

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TBDATA.2017.2762719, IEEE Transactions on Big Data

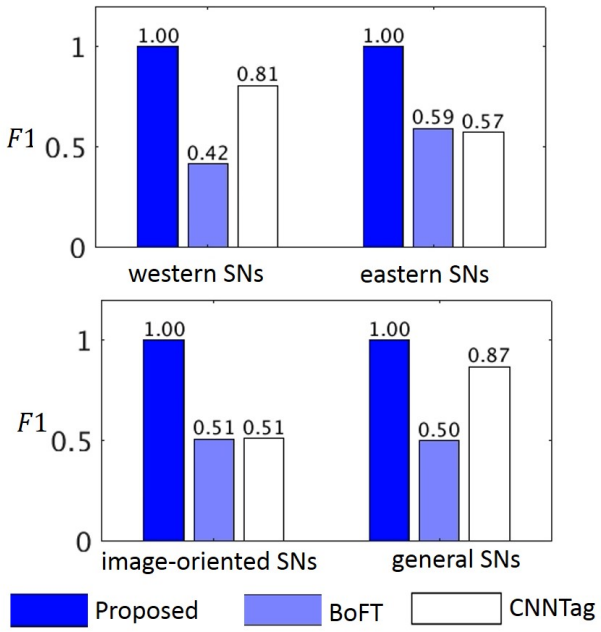

Fig. 12: Comparison of follower/followee recommendations with different approaches

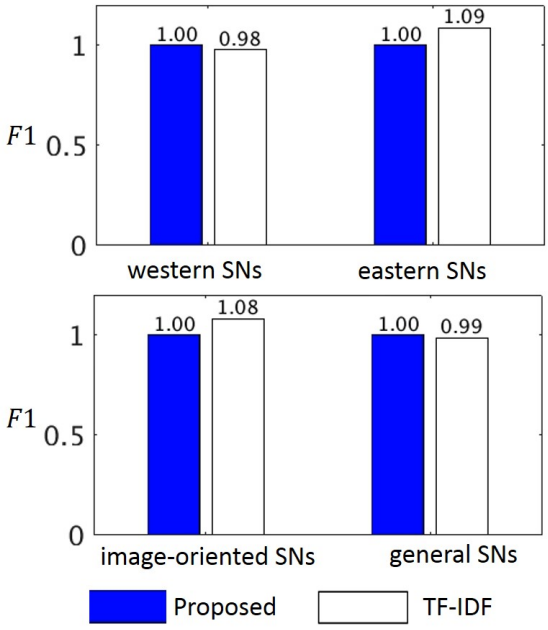

Fig. 13: Comparison of follower/followee recommendations with TF-IDF

predicted (from Eq. 13) normalized $F1$ on different SNs. The actual $F1$ is computed by making recommendations with different $\widehat{S_{i,j}}$, while the predicted $F1$ is the one predicted using Eq. 14 with different $\widehat{S_{i,j}}$. Note that the two values are normalized by dividing the maximum value in each set of data. As observed, the predicted $\widehat{S_{i,j}}$ is a good indicator of the trend of the actual $\widehat{S_{i,j}}$. The value of $\widehat{S_{i,j}}^*$ of different SNs is closed to the actual value of the maximum $F1$.

6.5 Showcase: Comparing CNN and BoFT

It is desirable to investigate how much improvement can be achieved by using a CNN-based approach. An experiment that make 1-10 recommendation per user has been conducted, and the corresponding $F1$ is recorded. Note that the experiment does not consider the distribution and Psi, as the distribution from other approaches may not follow the same distribution. Two approaches are implemented for the evaluation. The first one is Bag-of-features Tagging (BoFT) [1], which has similar procedures as the proposed engine, but uses a SIFT-based technique to encode the vector. It is the baseline to evaluate the improvement by applying CNN for image encoding. The second uses is using the tag generated by CNN [6] (CNNTag), which it is used to classify images by the 1000 objects. The user profiles are built and the connections are discovered using the same procedures as the proposed engine.

The results are shown in Fig. 12. Note that they are then divided by the $F1$ of the proposed engine, and averaged according to their origins and mechanisms. It is clear that proposed engine is at least 138% better in western SNs, and 69% better in eastern SNs than BoFT. it is proven that the proposed engine is 23% and 75% better than CNNTag in western and eastern SNs. A similar conclusion can be found while comparing their mechanisms. It is proven that the proposed engine is better.

6.6 Showcase: Comparing TF-IDF

One of the most common techniques in using user annotated tags is through term frequency-inverse document frequency (TF-IDF) [36]. It is a statistic method to reflect the importance of a tag in any document in similarity calculation. For those tags that are commonly used across many users have a lower weight. In the experiment, the same procedures as the proposed engine are used, but using TF-IDF when building the user profile:

$$l_{i,k}^{(T)} = (1 + log(l_{i,k})) \cdot log(1 + \frac{N^{(u)}}{df(k)}) \quad (17)$$

where $l_{i,k}$ is the number of labels of user i on the k-th machine-generated label, $N^{(u)}$ is the number of users and $df(k)$ is the user frequency of k-th machine-generated label. For example, if k-th machine-generated label is annotated on images of 3 users only, then $df(k) = 3$. A label presents in many users has a lower weight. Fig. 13 shows the result of the comparison. It is observed that the approach using TF-IDF can only slightly improve the performance. One of the reasons is the limited number of unique labels.

7 Conclusion

This work has demonstrated that user connections can be discovered using the 2M user-shared images from 8 real world SNs originating from the East and the West with different content sharing mechanisms. It is demonstrated that related users tend to share similar images, and this phenomenon is independent of the origins and sharing mechanisms of the SNs, using a deep learning technique, convolutional neural network. By utilizing this observed

phenomenon, an analytics engine is proposed to discover connections for follower/followee recommendations. To the best of our knowledge, this is the first attempt in related fields to characterize such phenomenon by massive user-shared images collected from real-world SNs, and then formulate and develop the results into practical methods to discover user connections. This work can significantly enhance the understanding of user-shared images on social media, and create a long-term impact to the related fields.

APPENDIX A
DETAILS OF $G(S_{i,j} = s|\lambda)$

Consider the exponentiate function from [1]:

$$g(S_{i,j} = s|\lambda, \alpha) = \alpha e^{-\lambda s} \tag{18}$$

As $S_{i,j}$ is ranged for 0 to 1:

$$\begin{aligned} \int_0^1 g(S_{i,j} = s|\lambda, \alpha) ds &= 1 \\ \int_0^1 \alpha e^{-\lambda s} ds &= 1 \\ \alpha &= \frac{1}{\int_0^1 e^{-\lambda s} ds} \\ \alpha &= \frac{-\lambda}{e^{-\lambda s}|_0^1} \\ \alpha &= \frac{\lambda}{1 - e^{-\lambda}} \end{aligned} \tag{19}$$

Hence, Eq. 18 becomes:

$$g(S_{i,j} = s|\lambda) = \frac{\lambda e^{-\lambda s}}{1 - e^{-\lambda}} \tag{20}$$

By taking integration from 0 to t, where $t \subset [0, 1]$:

$$\begin{aligned} G(S_{i,j} = t|\lambda) &= \int_0^t \frac{\lambda e^{-\lambda s}}{1 - e^{-\lambda}} ds \\ &= \frac{\lambda e^{-\lambda s}|_0^t}{-\lambda(1 - e^{-\lambda})} \\ &= \frac{1 - e^{-\lambda t}}{1 - e^{-\lambda}} \end{aligned} \tag{21}$$

which is Eq. 7.

APPENDIX B
PROOF OF EQ. 9

From Eq. 8 and putting Eq. 7:

$$\begin{aligned} &P(C_{i,j} = 1|S_{i,j} = s) \\ &= \lim_{\Delta s \to 0} \frac{[F(S_{i,j} = s + \Delta s|\lambda_r) - F(S_{i,j} = s|\lambda_r)]P(C_{i,j} = 1)}{(F(S_{i,j} = s + \Delta s|\lambda_a) - F(S_{i,j} = s|\lambda_a))} \\ &= \lim_{\Delta s \to 0} \frac{(\frac{1 - e^{-\lambda_r s + \Delta s}}{1 - e^{-\lambda_r}} - \frac{1 - e^{-\lambda_r s}}{1 - e^{-\lambda_r}})P(C_{i,j} = 1)}{(\frac{1 - e^{-\lambda_a s + \Delta s}}{1 - e^{-\lambda_a}} - \frac{1 - e^{-\lambda_a s}}{1 - e^{-\lambda_a}})} \\ &= \frac{1 - e^{-\lambda_a}}{1 - e^{-\lambda_r}} \frac{e^{-\lambda_r s}}{e^{-\lambda_a s}} P(C_{i,j} = 1) \lim_{\Delta s \to 0} \frac{1 - e^{-\lambda_r \Delta s}}{1 - e^{-\lambda_a \Delta s}} \end{aligned} \tag{22}$$

By using L'hôspital rule:

$$\begin{aligned} &P(C_{i,j} = 1|S_{i,j} = s) \\ &= \frac{1 - e^{-\lambda_a}}{1 - e^{-\lambda_r}} \frac{e^{-\lambda_r s}}{e^{-\lambda_a s}} P(C_{i,j} = 1) \lim_{\Delta s \to 0} \frac{\lambda_r e^{-\lambda_r \Delta s}}{\lambda_a e^{-\lambda_a \Delta s}} \\ &= \frac{\lambda_r(1 - e^{-\lambda_a})}{\lambda_a(1 - e^{-\lambda_r})} e^{(\lambda_a - \lambda_r)s} P(C_{i,j} = 1) \end{aligned} \tag{23}$$

ACKNOWLEDGMENTS

This work is supported by HKUST-NIE Social Media Lab., HKUST.

REFERENCES

[1] M. Cheung, J. She, and Z. Jie, "Connection discovery using big data of user-shared images in social media," *Multimedia, IEEE Transactions on*, vol. 17, no. 9, pp. 1417–1428, 2015.

[2] M. Cheung and J. She, "Evaluating the privacy risk of user-shared images," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 12, no. 4s, p. 58, 2016.

[3] M. Cheung, J. She, and X. Li, "Non-user generated annotation on user shared images for connection discovery," in *2015 IEEE International Conference on Data Science and Data Intensive Systems*. IEEE, 2015, pp. 204–209.

[4] M. Douze, H. Jégou, H. Sandhawalia, L. Amsaleg, and C. Schmid, "Evaluation of gist descriptors for web-scale image search," in *Proceedings of the ACM International Conference on Image and Video Retrieval*. ACM, 2009, p. 19.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[6] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *arXiv preprint arXiv:1405.3531*, 2014.

[7] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the ACM International Conference on Multimedia*. ACM, 2014, pp. 675–678.

[8] E. M. Jin, M. Girvan, and M. E. Newman, "Structure of growing social networks," *Physical review E*, vol. 64, no. 4, p. 046132, 2001.

[9] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and analysis of online social networks," in *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*. ACM, 2007, pp. 29–42.

[10] J.-D. Zhang and C.-Y. Chow, "igslr: personalized geo-social location recommendation: a kernel density estimation approach," in *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 2013, pp. 334–343.

[11] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: user movement in location-based social networks," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2011, pp. 1082–1090.

[12] L. Lü and T. Zhou, "Link prediction in complex networks: A survey," *Physica A: Statistical Mechanics and its Applications*, vol. 390, no. 6, pp. 1150–1170, 2011.

[13] S.-H. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, and H. Zha, "Like like alike: joint friendship and interest propagation in social networks," in *Proceedings of the 20th international conference on World wide web*. ACM, 2011, pp. 537–546.

[14] I. Guy, N. Zwerdling, D. Carmel, I. Ronen, E. Uziel, S. Yogev, and S. Ofek-Koifman, "Personalized recommendation of social software items based on social relations," in *Proceedings of the third ACM conference on Recommender systems*. ACM, 2009, pp. 53–60.

[15] W. H. Hsu, A. L. King, M. S. Paradesi, T. Pydimarri, and T. Weninger, "Collaborative and structural recommendation of friends using weblog-based social network analysis." in *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, 2006, pp. 55–60.

[16] X. Xie, "Potential friend recommendation in online social network," in *Green Computing and Communications (GreenCom), 2010 IEEE/ACM Int'l Conference on & Int'l Conference on Cyber, Physical and Social Computing (CPSCom)*. IEEE, 2010, pp. 831–835.

[17] W. S. Chow and L. S. Chan, "Social network, social trust and shared goals in organizational knowledge sharing," *Information & Management*, vol. 45, no. 7, pp. 458–465, 2008.
[18] J. Chen, W. Geyer, C. Dugan, M. Muller, and I. Guy, "Make new friends, but keep the old: recommending people on social networking sites," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2009, pp. 201–210.
[19] I. Kahanda and J. Neville, "Using transactional information to predict link strength in online social networks." *ICWSM*, vol. 9, pp. 74–81, 2009.
[20] E. Gilbert and K. Karahalios, "Predicting tie strength with social media," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2009, pp. 211–220.
[21] S. A. Golder and S. Yardi, "Structural predictors of tie formation in twitter: Transitivity and mutuality," in *Social Computing (SocialCom), 2010 IEEE Second International Conference on*. IEEE, 2010, pp. 88–95.
[22] X. Li, L. Guo, and Y. E. Zhao, "Tag-based social interest discovery," in *Proceedings of the 17th international conference on World Wide Web*. ACM, 2008, pp. 675–684.
[23] T. C. Zhou, H. Ma, M. R. Lyu, and I. King, "Userrec: A user recommendation framework in social tagging systems." in *AAAI*, 2010.
[24] A. Shepitsen, J. Gemmell, B. Mobasher, and R. Burke, "Personalized recommendation in social tagging systems using hierarchical clustering," in *Proceedings of the 2008 ACM conference on Recommender systems*. ACM, 2008, pp. 259–266.
[25] X. Zhang, X. Zhao, Z. Li, J. Xia, R. Jain, and W. Chao, "Social image tagging using graph-based reinforcement on multi-type interrelated objects," *Signal Processing*, vol. 93, no. 8, pp. 2178–2189, 2013.
[26] E. Moxley, J. Kleban, J. Xu, and B. Manjunath, "Not all tags are created equal: Learning flickr tag semantics for global annotation," in *Multimedia and Expo, 2009. ICME 2009. IEEE International Conference on*. IEEE, 2009, pp. 1452–1455.
[27] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International journal of computer vision*, vol. 42, no. 3, pp. 145–175, 2001.
[28] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.
[29] M. J. Swain and D. H. Ballard, "Color indexing," *International journal of computer vision*, vol. 7, no. 1, pp. 11–32, 1991.
[30] A. K. Menon and C. Elkan, "Link prediction via matrix factorization," in *Joint european conference on machine learning and knowledge discovery in databases*. Springer, 2011, pp. 437–452.
[31] B. W. Silverman, *Density estimation for statistics and data analysis*. CRC press, 1986, vol. 26.
[32] B. D. Ripley, *Pattern recognition and neural networks*. Cambridge university press, 2007.
[33] M. Cheung, X. Li, and J. She, "An efficient computation framework for connection discovery using shared images," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 2017.
[34] J. J. Moré and D. C. Sorensen, "Computing a trust region step," *SIAM Journal on Scientific and Statistical Computing*, vol. 4, no. 3, pp. 553–572, 1983.
[35] C. J. V. Rijsbergen, *Information Retrieval*, 2nd ed. Newton, MA, USA: Butterworth-Heinemann, 1979.
[36] H. Chim and X. Deng, "A new suffix tree similarity measure for document clustering," in *Proceedings of the 16th international conference on World Wide Web*. ACM, 2007, pp. 121–130.

Ming Cheung was born in Hong Kong. He received his B.Eng. and M.Phil in Electronic and Computer Engineering at Hong Kong University of Science and Technology (HKUST) in 2010 and 2012 respectively. He joined the HKUST-NIE Social Media Lab, Asia's first social media lab, in 2012 as a research assistant, and currently is a Ph.D. candidate at HKUST. His research interests include social media analytics, information diffusions and user behavior predictions.

James She is an assistant professor in the Department of Electronic and Computer Engineering at the Hong Kong University of Science and Technology (HKUST), and a visiting research fellow at the University of Cambridge. He is also the founding director of Asia's first social media lab, HKUST-NIE Social Media Lab, and spearheads multidisciplinary research and innovation in cyber-physical social media systems, viral media analytics and mobile media broadcast systems. Celebrated as a thought leader in new media and emerging cyber-physical societies, James is a member of the World Economic Forum's Global Agenda Council (Social Media) and joins other government and business leaders to develop solutions to key social media issues on the global agenda.

Ning Wang works as Senior Research Fellow in Data Science at the Oxford-NIE financial Big Data Lab, Mathematical Institute, University of Oxford. He also works as Research Associate at the Oxford Internet Institute. Prior to Oxford, he was a postdoctoral researcher at the Computer Laboratory, University of Cambridge. He is interested in exploring research opportunity in analysing a wide range of social and economic problems by exploiting big data approaches, with the hope that this work could contribute to the intersection of data science and computational systems. His research has appeared in PLOS one, Social Networks, Data Science, Entropy, etc.