DeepPick: A Deep Learning Approach to Unveil Outstanding Users Ranking with Public Attainable Features

Wanda Li, Zhiwei Xu, *Student Member, IEEE,* Yi Sun, Qingyuan Gong, Yang Chen, *Senior Member, IEEE,* Aaron Yi Ding, *Member, IEEE,* Xin Wang, *Member, IEEE,* and Pan Hui, *Fellow, IEEE*

Abstract—*Outstanding users* (OUs) denote the influential, “core” or “bridge” users in the online community. How to accurately detect and rank them is an important problem for third-party online service providers and researchers. Conventional efforts, ranging from early graph-based algorithms to recent machine learning-based approaches, typically rely on an entire network’s information. However, for privacy-conscious users or newly-registered users, such information is not easily accessible. To address this issue, we present DeepPick, a novel framework that considers both the generalization and specialization in the detection task of OUs. For generalization, we introduce deep neural networks to capture nonlinear features. For specialization, we leverage the traditional well-defined features to make use of particular information about users. Extensive experiments based on real-world datasets demonstrate that our approach achieves a high efficacy in terms of detection performance against the state-of-the-art.

Index Terms—Deep Neural Networks, Online Social Networks, Outstanding User Detection.

1 Introduction

THE rapid growth of online social networks (OSNs) brings a surge in user-generated contents (UGC). On the one hand, the massive base of UGC is of great help for people to make decisions about their daily lives; on the other hand, however, it also troubles people when they are trying to decide what to follow, especially in popular platforms like Yelp [1], [2], [3] and Foursquare [4], [5], [6]. Generally, a user tends to be influenced by the *outstanding users* (OUs) in the network, since they always play a critical role in online communities [7], [8]. Based on the popular independent cascade (IC) model [9], OUs may have better capabilities of information dissemination. Representative examples of OUs are structural hole spanners [10], big egocentric (e.g., degree centrality or ego-betweenness centrality [11]) users, and elite users in the network.

OUs have long been shown to be one of the fundamental building blocks of many business problems and social applications, e.g., recommender systems [12], [13], viral marketing [14], and information diffusion [15], [16]. Thus, there are various methods of OU detection in recent literature [17], [18], [19], [12], [20]. However, existing methods suffer from one or more of the three following drawbacks in the application: 1) The network structure sometimes is fragmentary due to users’ privacy configurations. Users may choose to hide their friend lists or follow someone privately. Thus, obtaining the entire social graph is very difficult, if not impossible, for third-party service providers or researchers. 2) The UGC and other types of user features are usually abundant in real-world scenarios and can provide idiographic information to depict a user. Ignoring these features can cause inaccuracy in the task of distinguishing particular types of users. 3) Modern OSNs usually contain millions of users or more, and the transductive algorithms may introduce complexity to the identification process.

To resolve these issues, we develop DeepPick, a novel framework that detects OUs without referring to the social connectivity information of the entire network. DeepPick only uses the information a user is willing to provide to extract the social graph-related characteristics. Such information could be the reviews, visited POIs, and profiles. The features DeepPick leverages include five types: sentiment, temporal, linguistic, spatial, and demographic. Sentiment features indicate the inner characteristics of a user. To get those implicit features, we propose the TextCNN Long-short Term Memory (TC-LSTM) structure, which uses the user’s reviews to detect OUs. Apart from the sentiment view, reviews could provide recognizable features from other particular perspectives, i.e., temporal and linguistic. We extract those features by analyzing the user lifecycle and the LIWC [21] of reviews separately. Spatial features are the user’s visited locations’ attributes, revealing users’ social status by visiting real-world locations. Demographic features are conventional material for user analysis and have been shown helpful. Most times, they can be obtained from the user’s profile.

We have made three key contributions:

- We formulate the concept of OU to represent the nodes with high diffusion capability in the information cascade (IC) model, then design and implement a frame-

- *Wanda Li, Zhiwei Xu, Yi Sun, Yang Chen, Qingyuan Gong and Xin Wang are with the School of Computer Science, Fudan University, China.*
- *Aaron Yi Ding is with the Department of Engineering Systems and Services, Delft University of Technology, Netherlands.*
- *Pan Hui is with the Department of Computer Science, University of Helsinki, Finland and the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong.*

work, DeepPick, to distinguish those OUs with the information of just a few users for training.
- We propose a mixed feature selection strategy of combining deep neural networks and traditional feature selection methods, which can be conveniently applied in detecting OUs in OSNs.
- Our framework can validate whether a user is outstanding by public attainable information instead of the whole network's structure. Exhaustive evaluations of the performance based on real-world datasets demonstrate the advantage of our approach against the state-of-the-art.

The rest of this paper is organized as follows. Section 2 introduces the background, presents the dataset, and shows the metrics of location. In Section 3, we expound on the idea of DeepPick, the OU detection framework. Section 4 describes the architecture of the sentiment features extractor, TC-LSTM, the deep learning module of our framework. Section 5 provides a thorough explanation of implementing details and evaluation. Section 6 outlines previous studies related to our work. Section 7 discusses more about related methods, definition of active users and the time complexity. In the last section, all results are summarized and concluded.

2 Background and Dataset

In this section, we illustrate the background of our study. We first formally define the outstanding users (Section 2.1) and then overview the two datasets we used in this work (Section 2.2). After that, we introduce how to depict the role of locations for outstanding user detection (Section 2.3).

2.1 Definition of Outstanding Users

As Gladwell states in "The Tipping Point" [22], outstanding users are the 20% participants who occupy 80% of social capital. Although there are diverse styles of argument [23], [24], [25] of social capital, in short it, is the metaphor about social advantages. The advantages include information, influential power, and trust. Several studies have validated that a set of users are especially related to positive indicators of social capital, including social impact [26] and information diffusion [27], [28]. According to [29], social benefits can be attained by spanning some unique positions in the network. In this paper, we refer to this group of important users as outstanding users. Formally, we have the following definition,

Here we first introduce one of the most widely used models of information diffusion, the independent cascade (IC) model [9], [30], [31]. In the case of the IC model, an activated node a in set A_i will activate its neighbours W with a probability of diffusion $p_{a,W}$ defined by their social links. This procedure is iterated in discrete steps (i.e., $i = 0, 1, \dots$). The initial set, A_0, represents the first activated node-set responsible for starting the information diffusion process. High information diffusion speed preferred by the marketing services can be reached by choosing nodes with higher social impact as A_0. Now we formally define outstanding users based on the IC model:

Definition 1. **Outstanding Users.** Given a social network $G = (V, E)$, where V is the set of all users, $E \subseteq V \times V$ is the set of all social relationships between users, outstanding users (OUs) are a group of users who have higher diffusion capability than others. OUs can widen and speed up information dissemination if they are considered as the first activated node set A_0 of the IC model.

These nodes can be selected by the criteria of information dissemination capabilities, including the speed of spread within or between communities or specific nodes' information influence. Typical examples of OU are structural hole spanners, influential users, and elite users. They are defined as follows:

Definition 2. **Structural Hole Spanners.** The structural hole theory [29] shows that people will benefit from acting as the "bridge" of different people or communities that are otherwise disconnected. Known as structural hole spanners (SHS), those people become non-trivial because they have more control over the information transmitted among communities [32], [33], [34].

SHS is a good choice for A_0 as they are the bridges of information flow between communities. They can be selected by four metrics, i.e., effective size, efficiency, constraint, and hierarchy [10]. Another structure-based OU example is the influential users:

Definition 3. **Influential users.** Influential users are a group of users with the highest influence on others. The users' influence in a network can be measured by the user centrality values [35].

Centrality metrics include degree centrality, closeness centrality, and betweenness centrality. They can measure the importance of individuals [36], [37], [12]. Selecting the influential users as A_0 will maximize the information diffusion speed inside a community.

As literature [38] shows, the above-mentioned widespread norms are correlated with cascades' properties. Most existing OUs discovery methods rely on global social connectivity information [39], [40], [20], but getting such information could be challenging due to users' privacy configuration. To deal with possible data famine, we use a subset of nodes' ego network structures to draw statistically significant conclusions about the whole population. Those metrics of ego network are shown powerful in works like [11], [10], [41], [42].

Despite the structure-based methods, a user can be identified as an OU based on both social network connections and additional metrics. In this paper, such users are named elite users.

Definition 4. **Elite users.** Elite users are labeled by online sites. The standard of elite users varies from one platform to another. For example, on Yelp, users who have well-written reviews, high-quality photos, a detailed personal profile, and a history of playing well with others are more likely to be recognized as elite users.

Using elite users as A_0 considers the condition that information spreads implicitly via UGC instead of social connections.

These specific types of OUs show characteristics related, but not limited to, network structure information, demographic features, and temporal patterns. In the following sections, we measure OUs' characteristics in these aspects and leverage them to design a model for OU detection.

2.2 Dataset Description

We employ the data of Yelp as our primary dataset to discover OUs and include data from Foursquare to ensure our results are generalizable. Both of them have attracted tens of millions of users all around the world, and are widely referred to in related studies of OSNs [43], [3], [6], [4], [5].

The Yelp dataset[1] is publicly available and spans from October 2004 to November 2018 in ten cities of North America. It comprises 1.6 million users, with their 1.9 million reviews and tips of 192 thousand businesses. The Foursquare dataset is a subset of data in [43], spans from October 2008 to February 2016 with spots all around the world. It has 2.9 million users and 630 thousand tips from 210 thousand venues.

The data entries of users vary from one platform to another. There will be user demographic information like ID, name, review account, friend list. In some cases, the platform also has the average number of their rated stars and other comprehensive assessments of the reviews and tips[2], which are a great resource for extracting the location visiting histories of users as described in [3]. On the other hand, the related locations also provide social information of users. In both datasets, locations are the real-world POIs, to which people could post check-ins and write reviews. Their IDs, locations, categories, attributes, stars, and review details are accessible to the public.

We select several types of OUs as examples in this work:

- Structural Hole Spanners: We rank them by effective size (in descending order), constraint (in ascending order), and hierarchy (in ascending order) of ego network separately [10], and label the top-k users as structural hole spanners. They are denoted as SHS(E), SHS(C), SHS(H).
- Influential Users: We employ degree centrality (noted as Degree later) [44] and ego betweenness centrality (noted as Ego-Betw later) [41] as two independent measurements.
- Elite Users: We use the users in "Elite Squad" of Yelp as the labeled elite users in the Yelp dataset.

Each group comprises 10,000 users, with an equal number of OUs and their counterparts (noted as Normal users). For each user type, we randomly choose the normal ones from the non-OUs. All the users are from the largest connected component of the social graph.

2.3 Measuring the Role of Locations

Besides conventional OSNs, geo-social networks also provide an indispensable view to distinguish between OUs and the other. Taking the reviewed POIs (named locations in this work) of users into account could lead to higher accuracy when detecting OUs. Here we first set up the geo-social network model, then describe measures of the social diversity associated with a location through its social network of visitors.

1. https://www.yelp.com/dataset
2. hereinafter called "reviews".

2.3.1 Interconnected Geo-Social Network

The model of an interconnected geo-social network carries rich information about both users and locations. Locations have the property of connecting people, and their visitors may share similar attributes [45]. An individual's social neighborhood, $N^h(b)$, is its social network links to a location b at distance h. The 1-hop social neighborhood of location b would be composed of b's direct visitors; the 2-hop social neighborhood would include all individuals in the 1-hop neighborhood and their friends. In this work, we count in second-hand redundancy brought with friendships of visitors to a location, so we set hop h to 2.

2.3.2 Homogeneity

The homogeneity of a place expresses to what extent its visitors are homogeneous in characteristics. A user is more likely to be outstanding in his/her online community if the location group he/she reviewed has a wider range of homogeneity scores. Following the definition in [45], we measure the overall social homogeneity of a location by the mean cosine distance of every pair of its visitors' place preference vectors as

$$H(b) = \frac{\sum_{u,v \in N^h(b)} \frac{\mathbf{U} \cdot \mathbf{V}}{\|\mathbf{U}\| \|\mathbf{V}\|}}{|N^h(b)|(|N^h(b)| - 1)} \tag{1}$$

where $|N^h(b)|$ is the size of the network, and $\mathbf{U}$, $\mathbf{V}$ is the preference vectors of two users u and v, separately. One's preference vector represents the percentage of each category of locations he/she had visited. The length of the preference vector is equal to the number of location categories. The homogeneity value ranges from 0 to 1, proportional to the homogeneous level of the categories reviewed by pairs of location visitors.

2.3.3 Entropy

The entropy of a place describes the extent to which it is diverse concerning visits. By far, it is the most common notion for quantifying a location's popularity [46]. As OUs tend to visit popular places [47], it helps in the detecting process. Entropy is defined by Shannon entropy value:

$$E(b) = -\sum_{u \in N^h(b)} \frac{|r(u,b)|}{|r(b)|} \, log \frac{|r(u,b)|}{|r(b)|} \tag{2}$$

where $|r(u,b)|$ is the user u's number of reviews at location b and $|r(b)|$ is the total number of reviews of location b. Entropy has been exploited in mobility studies to describe a location's popularity and its visitors' geographical diversity level [46], [48], [49]. More specifically, locations that are reviewed and given tips by highly diverse visitors will have higher entropy.

3 Design of the Outstanding User Detection Framework

In this section, we introduce our OU detection framework, DeepPick. We restrict our discussion to the setting of OU detection in online media. We demonstrate the overall structure in Section 3.1, and introduce input (i.e., reviews and descriptive data of both users and locations) in Section 3.2 and 3.3. The choice of machine learning algorithm (the decision maker) is discussed in Section 3.4.

3.1 System Overview

DeepPick has three modules, namely the review processing module, conventional feature extraction module, and the decision maker module. The framework is shown in Fig. 1.

We propose to utilize a deep neural network method to approximate the users' structural features. Deep artificial neural networks are now successful in many fields, but their palatable performance in generalizing may still require some theoretical explanation [50]. To remedy the shortcoming, we integrate a deep learning structure with handcrafted features. This heuristic method is helpful in enhancing the interpretability of our model. Before feeding the deep module, we order all review texts of each user chronologically, remove stopwords, punctuations, and conduct lemmatization. For the handcrafted features, we leverage each user's descriptive profile and the attributes of the reviewed locations to describe the user from the perspective of demographic and location. Putting the subsets of features in Table 1 together, the decision maker applies a supervised machine learning-based classifier to predict whether a user is an OU or not.

In the following subsections, we introduce the building blocks of DeepPick and discuss their contributions to the final decision.

3.2 Review Processing

The reviews provide rich information over the user's lifecycle from comprehensive perspectives, revealing the difference between outstanding and normal users from a statistical view. In DeepPick, all reviews of a user are analyzed mainly from three angles: sentiment, linguistic, and temporal. We leverage the TC-LSTM frame (see Section 4) as the deep learning module for sentimental analysis. Two other analytical views are represented as follows.

3.2.1 Linguistic Features of Reviews

User-generated contents have long been used in understanding users of OSNs, such as [51], [52]. Using the review text, we investigate how linguistic aspects affect users' outstanding status. Herein we adopt user language patterns as a proxy to look into their online actions and engagement patterns in the community. LIWC [21], or Linguistic Inquiry and Word Count, is our tool to demonstrate the characteristics of user reviews. LIWC counts words in psychologically meaningful categories. It has been widely used to detect how people's daily spoken and writing text like, reveal their social relationships, thinking styles, and individual differences [53].

Some selected review linguistic features are exhibited in Table 2. The values represent the average occurrence frequency of each specific set of words. Note that all types of OUs share the same LIWC static features compared with normal ones. The first column is the word count (WC) of each review, which correlates with psychological meanings like talkativeness and verbal fluency [53]. We can see that OUs' average review length is longer than normal users'. Also, as we show in Table 2, OUs always post more reviews than others. This means OUs are more proactive in online communities. The "Analytic" metric captures the degree to which people use words that suggest formal, logical, and hierarchical thinking patterns. OUs are constantly lower in analytical, showing their tendency to write and think in more narrative ways, focus on here-and-now and personal experiences [54]. The "focuspast" metric measures the frequency of using words and past tenses that express past tense. OUs write more about the past times, which indicates they are more practiced in flashing back. With regard to the "social" (words referring to social relationships, such as "family" and "friends") metric, OUs appear to have lower scores. The frequency of words about leisure activities like "cook", "chat", and "movie", which belong to the "Leisure" category, is presented in the last column. They are prone to appear more in OUs' reviews.

3.2.2 Review Temporal Features

Reviews, as the major part of UGC, are helpful as a proxy of users' psychological and behavioral patterns. The content shows how people think while the review time sequence indicates the life span of their generator. In this work, we define that a user's lifecycle starts when he/she posts the first review on the platform and reaches the final stage when he/she posts the last review. We divide the life span into five segments based on the time duration. Each time duration could be interpreted as a life stage of real human life [3]. Fig. 2 and Fig. 3 show the average number of reviews people write in different life stages. Regardless of categories, OUs share the same review patterns. All subsets of OUs write more reviews in all of their life stages. They also share a similar style in developing patterns: the contributing pace of most OUs shows a consistent tendency of slowing down in the formal stage but rises in the last stage. Normal users also contribute more in their final life stage, which means they are slightly engaging more in the communities.

3.3 Statistical Feature Analysis

Two kinds of features are informative in user distinguishing: their location role in the social network and demographic static. Hristova et al. [45] validated that the bridging and bonding role of places could reveal special attributes in social networks. Some measurements like homogeneity are structural, while entropy is based on probability. On the other hand, user descriptive information in their profile is a conventional feature in user distinguishing tasks [55], [3], [56]. They are easy to access and beneficial.

3.3.1 Measurement of the Social Diversity of Locations

One of the fundamental social roles of locations is to bring people together. Locations, as people, could act as bonding hubs. Some tend to bring along friends to interact with each other, while some are more likely to gather otherwise disconnected individuals. The efficiency of the locations' social roles also depends. While bringing together strangers stimulates the formation of new social exchanges, these might not be genuinely diverse if the population of visitors is homogeneous. Pieces of literature show that visitors'

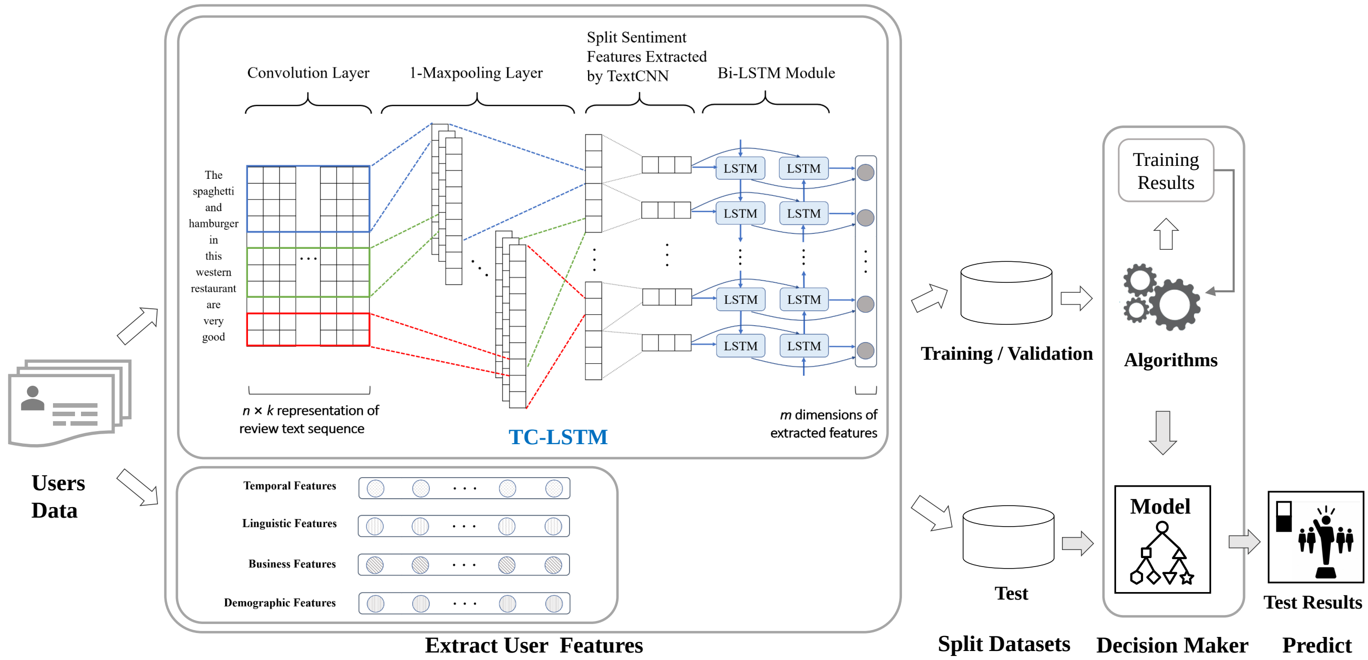
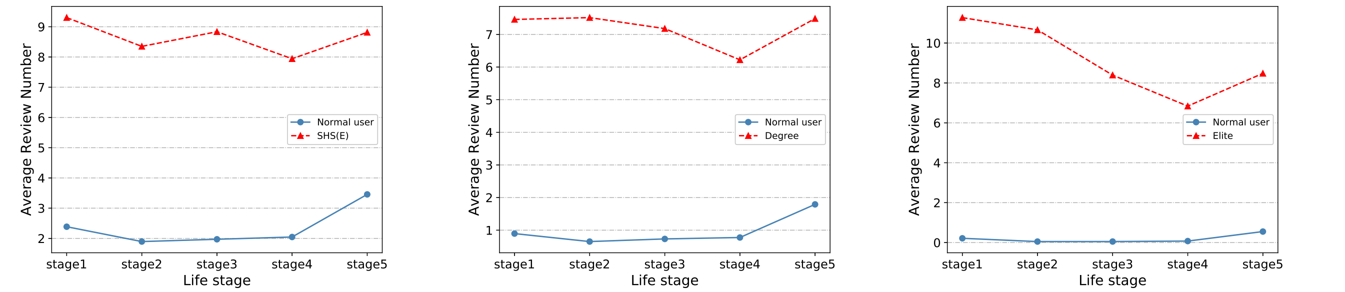


Fig. 1. DeepPick Framework. For each user's information in the input sequence, DeepPick extracts the sentiment features by TC-LSTM, the deep learning module. Then, it extracts and normalizes other handcrafted features to incorporate with the sentiment ones. After being aggregated together, these feature sets are fed into the machine learning algorithm for prediction.

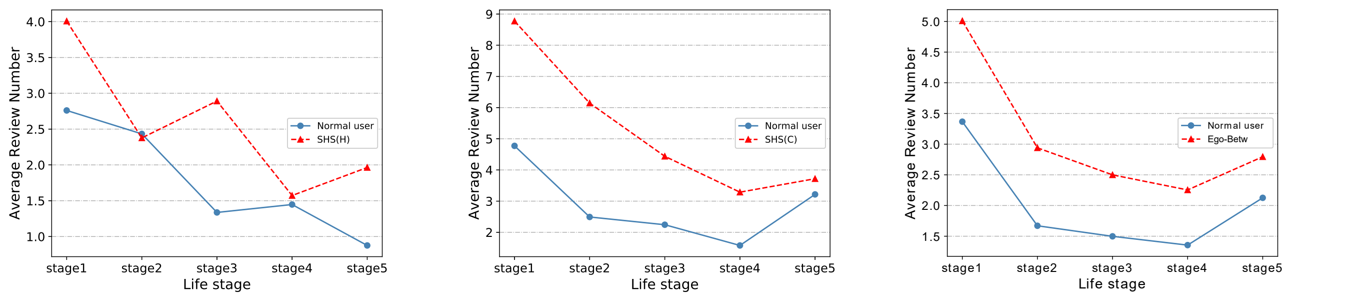
TABLE 1
Subsets of features of the classification model.

	Sentiment Features
S_i	Sentiment analysis results output by the deep learning module structure ($i \in [1, 30]$)
	Temporal Features
Review1	The number of reviews a user has posted in his/her first period of lifecycle
Review2	The number of reviews a user has posted in his/her second period of lifecycle
Review3	The number of reviews a user has posted in his/her third period of lifecycle
Review4	The number of reviews a user has posted in his/her fourth period of lifecycle
Review5	The number of reviews a user has posted in his/her last period of lifecycle
	Location Features
Entropy	The diversity of visits with respect to visitors
Homogeneity	The extent to which a location's visitors are homogeneous in their characteristics
	Linguistic Features
WC	Number of words per review
Analytic	The degree to which people use words that suggest formal, logical, and hierarchical thinking patterns
focuspast	The extent of using past focus words like "ago", "talked", and "did"
social	The extent of using social processes words like "mate", "talk", "they"
Leisure	Frequency of occurrence of words in "Leisure" category, like "cook", "chat", "movie"
	Demographic Features
Review_count	Total number of reviews the user have written
start_time	When the user posted the first review
B_a	In Foursquare, this metric means the count of the user's basic attributes, a (e.g. venue lists, contact)
C_j	In Yelp, this metric means how many compliments of type j (e.g. hot, cool, funny) are received by the user



(a) SHS(Effective size) (b) Degree (c) Elite

Fig. 2. Yelp OUs' review tendency. OUs in Yelp are much more active than normal users throughout their life span. The level of OUs engagement keeps fluctuating but still sticks to a relatively high score, which indicates that the OUs in Yelp might be detected from a very early stage of their lifecycle.



(a) SHS(Hierarchy) (b) SHS(Constraint) (c) Ego-Betweenness

Fig. 3. Foursquare OUs' review tendency. In Foursquare, the gap between OUs and normal users is smaller but still significant, especially at the beginning of the lifecycle. Most of the Foursquare users show a tendency to become inactive.

TABLE 2
Occurrence frequency of different categories of words in OUs and their counterparts' reviews. NU (Normal User) are marked with corresponding OUs' category.

	User Type	WC	Analytic	focuspast	social	leisure
Yelp	SHS(E)	103.20	60.41	4.71	7.50	2.14
	NU(S)	100.66	55.81	5.58	8.12	1.81
	Degree	138.09	62.43	4.87	7.00	2.38
	NU(D)	93.13	55.08	5.93	8.32	1.75
	Elite	148.57	63.69	5.35	6.55	2.35
	NU(E)	95.46	54.31	6.33	9.14	1.55
Foursquare	SHS(C)	48.57	82.52	1.35	5.05	3.00
	NU(C)	13.06	79.95	1.11	5.23	2.51
	SHS(H)	27.79	81.46	12.93	5.87	2.84
	NU(H)	13.11	80.14	12.15	6.25	2.61
	Betweenness	20.30	78.91	1.41	5.64	2.91
	NU(B)	12.52	76.75	1.54	6.49	1.99

diversity is in proportion to the place's characteristics. For example, the visitors' composition and social network connectivity comply with the score of homogeneity [45].

We obtain the homogeneity of a place as in Equation 1. In this work, we use statistics (i.e., the mean, median, maximum, minimum, and quartiles) of all locations a user reviewed to depict his/her location preference.

3.3.2 Demographic Features

Demographic features could be extracted from each user's basic profile information. Different online platforms have different feature sets. Introducing carefully selected features will strengthen the detection framework. In our case, the demographic features include the time when the user first active on the platform (i.e., post the first review), how many reviews he/she has given and some platform-specific metrics. Specifically, Yelp describes users by a large number of accomplished items (all started by "compliment_"), which are helpful evidence to find out whether a user is outstanding.

3.4 Decision Maker

DeepPick leverages a decision maker to conduct the final judgment. The decision maker could be the classifiers using supervised machine learning algorithms such as CART decision tree [57], Random Forest [58], or recently proposed boosting systems such as XGBoost [59] and CatBoost [60].

We refer to the default log loss function to adjust the parameters, conduct a grid search on a range of possible values to optimize it. The loss is calculated by

$$L = -\frac{\sum_{i=1}^{N}(y_i log(p_i) + (1-y_i) log(1-p_i))}{N}, \quad (3)$$

where N is the total number of users, p_i is the probability that the classifier makes a correct judgement on i-th user. We could get an L with a particular set of parameters. At the end of training, the set of parameters that keeps L value lowest will be chosen to let the decision maker achieve the best performance.

For each dataset, we construct a training and validation set, which contain 80% users, and employ the rest 20% as the testing set. Both train and test datasets have the same distribution of user type as the total population. We use 5-fold cross-validation for the training and validation dataset.

4 TC-LSTM: User Review Analysis Frame

User reviews reveal what a user is thinking. It shows the personal characteristics of users, which greatly help in distinguishing OUs. A user may produce thousands of reviews, forming an interrelated text sequence with timestamps. Such sequences are hard to interpret via traditional methods and existing basic deep learning methods. To make the best use of all implicit information of the reviews, we propose TC-LSTM to extract sentiment features. It takes the review text sequence produced by one user and is trained to predict the user label. We then leverage the outputs of the LSTM layer as sentiment features to depict the user. TC-LSTM (Section 4.1) combines the advantages of TextCNN (Section 4.2) and Long Short-Term Memory (LSTM) (Section 4.3) framework, and performs better than both of them.

4.1 The Proposed Network Architecture

The middle part of Fig. 1 shows the network architecture of TC-LSTM. It consists of a TextCNN module [61] and a LSTM module [62] from bottom to top. The training label of TC-LSTM is the label of users, i.e., outstanding or not. TC-LSTM can be trained with one loss function jointly. TC-LSTM and the following machine learning algorithm share the same training/validation and test datasets.

At the bottom of TC-LSTM, the TextCNN module extracts a feature sequence from input user reviews representation. On the top of TextCNN, a BiLSTM module works to convert frames of the feature sequence into specified dimensions of sentiment features. Before being fed into TC-LSTM, the sequence of user reviews will be represented by trained word2vec vectors in a "Bag of Words" architecture [63] and padded to the same length.

For the representation of the user's review text, we extract its sentiment features via TextCNN. The TextCNN module is constructed by taking the convolutional and max-over-time pooling layers from a standard TextCNN model (fully-connected layers are removed). TextCNN needs little tuning of hyper-parameters and performs well in sentiment analysis. After the convolution operation, we perform a nonlinear transformation on the output. To maintain the nonlinear characteristics after the max-pooling operation, we use *ReLU* function to compress the feature map output by the TextCNN module into hidden variables. We then convey those features into sequential representations in order to be invariant to the length variation of sequence-like objects. The number of output features is customizable.

As the fixed kernel size of TextCNN limits it to the information of the current few words, we feed its output into a BiLSTM layer. BiLSTM views a sentence as a sequence of tokens and uses two LSTMs to represent each token of the sequence based on both past and future contexts. It has shown advantages in dealing with long-distance dependency and successes in natural language processing tasks.

4.2 Text Feature Extraction

For the TextCNN part, it first generates a word matrix for each segment of words. It represents a k-dimensional word vector as $\mathbf{x}_i \in \mathbb{R}^k$. Such a vector is corresponding to the i-th word in the sentence. After padding to n-length, a sentence could be represented as

$$\mathbf{x}_{1:n} = \mathbf{x}_1 \oplus \mathbf{x}_2 \oplus \ldots \oplus \mathbf{x}_n, \tag{4}$$

where $\oplus$ is the concatenation operator. After concatenation, the word matrix will have a shape of $n \times k$.

Second, in order to produce a new feature, TC-LSTM extracts sentiment features from word matrix by involving *filter*s for a window of *h* words in convolution operations. A *filter* is $\mathbf{w} \in \mathbb{R}^{hk}$. The model uses multiple filters (with varying window sizes) to obtain multiple features. In general, let $\mathbf{x}_{i:i+j}$ refer to the concatenation of words $\mathbf{x}_i, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_{i+j}$. A feature c_i is generated from a window of words $\mathbf{x}_{i:i+h-1}$ by

$$c_i = f(\mathbf{w} \cdot \mathbf{x}_{i:i+h-1} + b). \tag{5}$$

Here $b \in \mathbb{R}$ is a bias term and f is a nonlinear function such as the hyperbolic tangent. A *filter* will be applied to each possible window of words in the sentence $\{\mathbf{x}_{1:h}, \mathbf{x}_{2:h+1}, \ldots, \mathbf{x}_{n-h+1:n}\}$ to produce a *feature* map

$$\mathbf{c} = [c_1, c_2, \ldots, c_{n-h+1}], \tag{6}$$

with $\mathbf{c} \in \mathbb{R}^{n-h+1}$.

The next step is to apply a pooling scheme to get the corresponding features of each particular filter. In this step, a max-over-time pooling function is applied over the feature map and the maximum value $\hat{c} = max(\mathbf{c})$ is used as the target feature. In this paper, the fully connected layers in TextCNN are removed to make the model more compact and efficient.

4.3 Sequence-Based Classification

Recurrent neural network (RNN) has a strong capability of capturing contextual information within a sequence. Traditional RNN units, however, suffer from the vanishing gradient problem [64]. It limits the range of context RNN can store and adds burdens to the training process. Fortunately, the LSTM structure can handle the long-distance dependency between elements in a time sequence [62] better than standard RNN. In particular, we choose bidirectional LSTM (Bi-LSTM), which consists of forward (left to right) and backward (right to left) LSTMs. According to the study in [65], Bi-LSTMs outperform unidirectional LSTMs for classifying frames of acoustic data into phonemes.

For the LSTM part, it predicts a label distribution y_t for each frame x_t in the feature sequence $\mathbf{x} = x_1, \ldots, x_T$. The deep structure allows higher levels of abstractions than a shallow one. We apply Back-Propagation Through Time (BPTT) in TC-LSTM. The sequence of propagated differentials is concatenated into maps at the bottom of the LSTM module. Then we invert the operation of converting feature maps into feature sequences and feed them back to the TextCNN module. To connect the two main parts, we construct a process to split the output sequence of TextCNN into shorter segments.

After the review processing, the LSTM module finally outputs m features, each represented by $S_i, i \in [1 \dots m]$ in the following part. The value of m could be adjusted.

5 Implementation and Evaluation

Having explained how DeepPick is designed, we turn to study what parameters and building blocks should be chosen to ensure the best performance of the OU detection task. We present the implementation details of DeepPick in Section 5.1. In Section 5.2, we conduct thorough experiments to show the impact of equipping different components on our framework's detection performance, the contribution of different feature groups and individual features, and the most discriminative features for various types of users. Finally, we compare the results with several state-of-the-art solutions. In Section 5.3, we present a case study to figure out the difference between individuals of different types.

5.1 Implementation and Metrics

To complete the review classification task, we use PyTorch, a widely-used open-source machine learning framework implemented by Python. In the preprocessing period, we employ Continuous Bag-of-Words (CBOW) of Word2Vec [66] to embed each sentence into 100 dimensions. Concerning the parameters of TC-LSTM, for all datasets we use: filter windows (h) of 2, 3, 4 with 10 feature maps each, dropout rate (p) of 0.7. For the RNN layer, we set the hidden size as 15, and two layers of Bi-LSTM are added. The network is then trained with stochastic gradient descent (SGD). The output number of dimensions, m, is set to 30. To implement the decision maker, we use scikit-learn [67], a Python-based machine learning library.

We adopt the following classic metrics to evaluate the detection performance.

- Precision: the fraction of detected OUs who are really outstanding in their social neighborhoods.
- Recall: the percentage of OUs that have been uncovered correctly.
- F1-score: the harmonic mean value of Precision and Recall.
- AUC [68]: the probability that the classifier will rank a randomly chosen OU more powerful than a randomly chosen normal one.
- Mean Average Precision (MAP) [69]: Mean value of average precision ($AP(k)$), which is the average of the precision value obtained for the set of top-k OUs. MAP is given by

$$MAP = \frac{\sum_{k=1}^{K} AP(k)}{K} \tag{7}$$

5.2 Experiment Results

In the evaluation, we test different constructions of DeepPick by equipping it with different neural network models and decision makers to optimize the performance. We then show the contribution of separate features to the final decision and compare DeepPick with some state-of-the-art solutions.

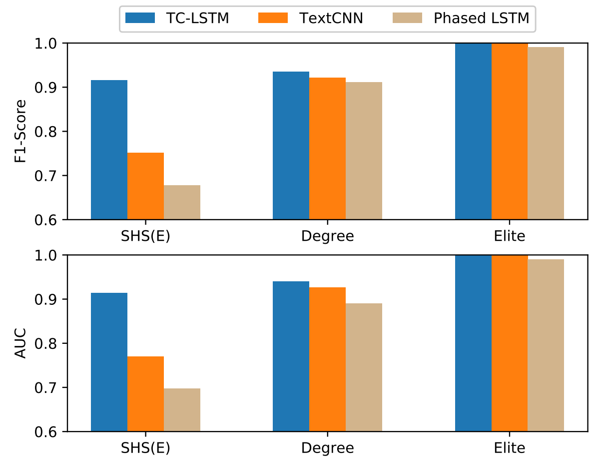


Fig. 4. Comparison of how different neural network structures work in the Yelp dataset. TC-LSTM performs the best in all cases. Specifically, the more complicated the user ranking metric, the more TC-LSTM outperforms other methods.

5.2.1 Comparison of Different Neural Network Models

Kim [61] proposed to use TextCNN to achieve good performance when processing language. Also, Phased LSTM [70], a recently emerging RNN for sparse sequences, reaches the best result in the work of Gong et al. [56]. For comparison, we measure the detection performance of these networks in our dataset using SHS(E), degree centrality, and elite users in Yelp.

We fed TextCNN and TC-LSTM with the same preprocessed reviews of users and the Phased LSTM framework with the time sequence of user reviews. Their detection performance is compared in Fig. 4. From the result, we find that TC-LSTM performs the best in all cases, especially when the users are selected based on a complicated graph structural metric.

5.2.2 Comparison of Different Decision Makers

Several algorithms could work as the decision maker in DeepPick. Classifiers, including tree boosting systems such as CatBoost and XGBoost, classic machine learning algorithms such as Random Forest (RF), and decision tree (CART), are tested for DeepPick. We show the prediction results and parameters of all algorithms in Table 3. Overall, the boosting methods perform better than others. We utilize McNemar's test [71] to examine whether there exists a difference between two classification algorithms by evaluating the statistical significance and find that XGBoost is different from the others. From the prediction performance concerning F1-score and AUC value, we see that XGBoost gets the best performance. In the following experiments, we use XGBoost as the decision maker in DeepPick.

5.2.3 Contribution of Different Feature Subsets

In this part of the experiments, we show which types of features are more discriminative through ablation experiments on different feature subsets. As a case study, we conduct experiments on different subsets of users on both platforms.

TABLE 3
Evaluation of different decision makers in the Yelp dataset.

		SHS(E)				Degree				Elite			
Models	Parameters	Precision	Recall	F1-score	AUC	Precision	Recall	F1-score	AUC	Precision	Recall	F1-score	AUC
XGBoost	learning_rate=0.01, seed=0, n_estimators=900, gamma=0.1, max_depth=3, reg_lambda=1, subsample=0.6, reg_alpha=1, min_child_weight=1, colsample_bytree=0.6	0.915	0.917	**0.916**	**0.914**	0.953	0.919	**0.935**	**0.940**	1.000	1.000	**1.000**	**1.000**
CatBoost	l2_leaf_reg=9, iterations=1000, one_hot_max_size=3, depth=4, learning_rate=0.1	0.915	0.912	0.914	0.912	0.948	0.918	0.933	0.937	0.999	1.000	1.000	0.999
CART	default	0.837	0.838	0.838	0.834	0.889	0.898	0.893	0.901	0.999	1.000	1.000	0.999
RF	max_depth=7, n_estimators=130	0.919	0.910	0.914	0.913	0.954	0.907	0.930	0.935	1.000	1.000	1.000	1.000

TABLE 4
Ablation study on different feature subsets (F1-score)

	Yelp			Foursquare		
Approach	SHS(C)	Degree	Elite	SHS(C)	SHS(H)	Ego-Betw
DeepPick	0.975	0.935	1.000	0.922	0.876	0.922
- Sentiment Features	0.898	0.926	0.991	0.872	0.716	0.755
- Temporal Features	0.972	0.857	0.991	0.912	0.837	0.867
- Location Features	0.962	0.846	0.995	0.910	0.828	0.862
- Linguistic Features	0.973	0.855	0.997	0.912	0.835	0.868
- Demographic Features	0.963	0.687	0.919	0.836	0.807	0.824
Random Guess	0.491	0.490	0.508	0.522	0.493	0.490
+ Sentiment Features	0.756	0.673	1.000	0.834	0.801	0.739
+ Temporal Features	0.837	0.811	0.932	0.666	0.582	0.562
+ Location Features	0.853	0.832	1.000	0.662	0.618	0.607
+ Linguistic Features	0.845	0.833	0.991	0.682	0.586	0.568
+ Demographic Features	0.889	0.933	1.000	0.867	0.699	0.755

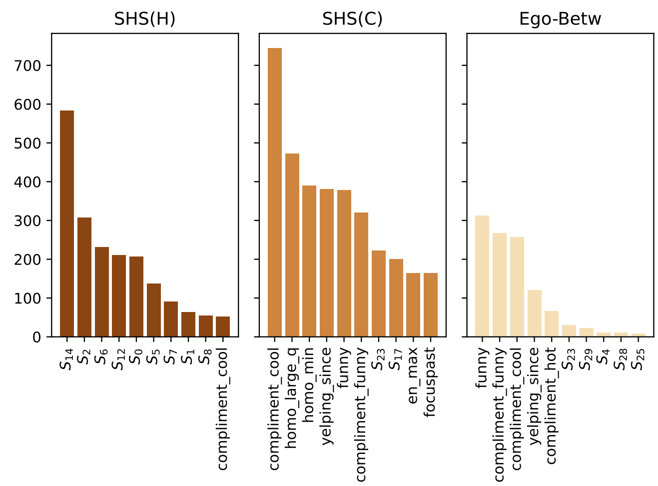


Fig. 5. Comparison of distinct features' contribution in different Yelp OU subsets. Sentiment features extracted by TC-LSTM are represented by S_i, in which i ranges from 1 to 30.

F1-score is applied to evaluate the detection performance of different approaches. First, we compare the performance of DeepPick without each type of feature. We subtract one subset at a time and validate the performance degradation accordingly. Table 4 shows that sentiment features, most of the time, are the most discriminative features in the detecting process of different types of users, as the F1-score decays fast when the sentiment features are removed. In other cases, i.e., for OUs judged by degree centrality and elite label in Yelp, demographic features are the key subsets. Second, we start from a random guess classifier and add one feature subsets to it each time. This time, F1-score increases most when demographic features are considered in most cases. Correspondingly, sentiment features enhance detection performance most when demographic features are not the best choice. The result implies that combining sentiment features and demographic features into distinguishing processes is able to increase the detection performance.

5.2.4 Contribution of Different Individual Features

We validate the importance of individual features in all feature subsets by XGBoost. It calculates the importance of the features according to their contribution to the prediction result. We plot the importance of each feature for the classification in Fig. 5 and Fig. 6. We can see that in many OU subsets, sentiment features extracted by TC-LSTM play an essential role. For structural hole spanners on Foursquare, no matter what metric they are ranked by and which dataset they come from, most top-10 features that contribute to identifying them are sentiment features. Also, demographic features from Yelp user profiles (C_j) like "compliments_cool" and visited location features like "homo_min" (which means the minimal value of visited locations' homogeneity) contribute a lot to the final decision, especially for OUs selected by less complicated graph structural metrics (i.e., degree centrality and elite-worthiness).

5.2.5 Performance on All Datasets

The performance of DeepPick (evaluated by F1-score) in all datasets is shown in Table 5. It maintains high accuracy in all subsets. In the case of elite users, the system makes no error. The reason lies in that elite users usually have some distinguishing demographic features in Yelp, such as "funny" (C_f) and "compliment_cool" (C_c). Those attributes make them especially recognizable. According to Yelp[3], their judging criteria of elites includes quality reviews and photos, truthfulness of user identity, and user age. The prediction performance shows no explicit difference between different types of users in the same dataset, which shows the scalability of DeepPick.

5.2.6 Comparison With State-of-the-Art Solutions

We compare the performance of DeepPick with the following special user detection algorithms for online communities

3. https://www.yelp-support.com/article/What-is-Yelps-Elite-Squad?l=en_US

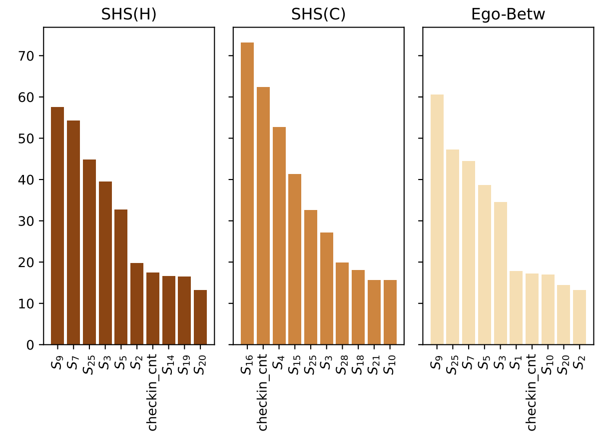


Fig. 6. Comparison of distinct features' contribution in different Foursquare OU subsets. *checkin_cnt*, one of the user's basic attributes, is the number of people's reviews. Sentiment features are $S_i, i \in [1 \dots m]$. The results show that they are among the most decisive features in all subsets.

TABLE 5
Experiment results of different datasets (F1-score)

	Subset	Precision	Recall	F1-score	AUC
Yelp	SHS(E)	0.915	0.917	0.916	0.914
	SHS(C)	0.976	0.975	0.975	0.975
	SHS(H)	0.928	0.934	0.931	0.931
	Degree	0.953	0.919	0.935	0.940
	Ego-Betw	0.973	0.953	0.963	0.963
	Elite	1.000	1.000	1.000	1.000
Foursquare	SHS(C)	0.915	0.929	0.922	0.915
	SHS(H)	0.876	0.901	0.892	0.844
	Ego-Betw	0.916	0.925	0.921	0.886

TABLE 6
Performance comparison with existing methods (F1-score)

	Yelp			Foursquare		
Models	SHS(E)	Degree	Elite	SHS(C)	SHS(H)	Ego-Betw
Ma et al.	0.751	0.613	0.811	0.795	0.620	0.693
Katsimpras et al.	0.677	0.696	0.835	0.723	0.681	0.571
MCDE	0.611	0.908	0.673	0.916	0.568	0.474
DeepPick (100OUs)	0.903	0.910	0.999	0.903	0.816	0.835
DeepPick (1,000OUs)	0.913	0.925	0.999	0.911	0.824	0.858
DeepPick	**0.916**	**0.935**	**1.000**	**0.922**	**0.876**	**0.922**

by both F1-score and a ranking-based performance metric, MAP.

- Ma et al. [72]: This work developed an RNN-based method to detect rumors in OSNs. RNN is used for learning the hidden representations that capture the variation of contextual information of users' relevant posts over time.
- Katsimpras et al. [73]: This work employed supervised random walks to rank users according to their topic-sensitive influence. Topics are extracted by LDA [74].
- MCDE [75]: MCDE mixes the value of node core, degree, and entropy (represented by the diversity of neighbors in different shells). To equate the effect of these measures, the amounts of these three parameters are considered to be equal to the original paper.
- DeepInf [76]: DeepInf first performs a random walk with a restart probability γ, and the size of the sampled sub-network is set to be w. For the embedding layer, a k-dimension network embedding is pre-trained. Then, DeepInf uses a three-layer GAT/GCN structure with r hidden units in both the first and second GAT/GCN layers, while the output layer contains 2 hidden units for binary prediction.

The result of the comparisons by F1-score is shown in Table 6. Evaluations based on ranking-based performance metrics (MAP) are also shown in Table 7. We can see that the other solutions perform well in identifying one type of OUs but fall behind in others. Even in the case they are most good at, DeepPick achieves higher performance than them.

As the available data only contains a limited part of the social graph, many related methods could not work in this case, including traditional structural hole spanners detection algorithms [32], [17], GNNs [76], [77], and structure learning methods [78], [79]. Specifically, we take DeepInf [76] as an example. In our experiments, we apply DeepInf-GCN instead of DeepInf-GAT because DeepInf-GCN performs better in our dataset. We select $\gamma = 0.8$, $w = 10$, $k = 64$, $r = 256$. To minimize the effect of missing connection information, we feed DeepInf with a subgraph of largest connectivity in the dataset, which contains 10,000 edges and 7,549 nodes. The OUs in the sampled dataset are SHS ranked by effective size. As in [76], we allow DeepInf to run at most 1,000 epochs over the training data and select the best model by early stopping using loss on the validation sets. However, DeepInf reaches AUC=0.613, Precision=0.228, Recall=0.669, and F1-score=0.340. We attribute the inferiority to the limited size of sampled networks, which cannot be very large due to the scant number of nodes in 1-hop neighborhood (i.e., ego network). Unfortunately, as some users choose to hide their friend lists from the public nowadays, it would be much more difficult to acquire neighborhoods larger than 1-hop for third-party service providers and researchers.

So far, the experiments are conducted based on datasets containing 10,000 OUs. However, some platforms might not provide so much OUs' information, i.e., the achieved datasets will likely face data imbalance problems. We conduct more experiments to validate the practicability of our proposed framework when the given outstanding user set is small, i.e., 100, 1,000. The results are shown in Table 6 and Table 7. The deduction in training OU numbers will slightly degrade the performance. However, by comparison, our method still performs better than the baselines even with less OUs.

5.3 Case Study

Since we have already illustrated that DeepPick outperforms existing approaches and shown the contribution of

TABLE 7
Performance comparison with existing methods (MAP)

	Yelp			Foursquare		
Models	SHS(E)	Degree	Elite	SHS(C)	SHS(H)	Ego-Betw
Ma et al.	0.697	0.622	0.791	0.770	0.698	0.688
Katsimpras et al.	0.643	0.610	0.815	0.769	0.684	0.625
MCDE	0.576	0.808	0.602	0.852	0.521	0.545
DeepPick (100OUs)	0.880	0.877	0.881	0.859	0.757	0.775
DeepPick (1,000OUs)	0.910	0.894	0.896	0.875	0.764	0.811
DeepPick	**0.919**	**0.988**	**0.999**	**0.884**	**0.771**	**0.819**

different features, we want to look into the detailed feature difference of different types of OUs by case studies. We randomly select 100 OUs of each type and their corresponding normal users to compare their features. We choose one feature that shows the most significant difference from each category of features in Fig. 7. As the value of different features ranges a lot, we change the value axis to a logarithmic scale. Normal users in Yelp hardly write reviews and tips, thus win almost no marks on the features of compliments like "funny". The average entropy of the places they visit is only 75% of that of OUs. The value of sentiment features (S_i) also exhibits a vast difference between normal users and OUs. In general, sentiment features have opposite trends, i.e., normal users are scored higher on sentiment features than OUs. In Foursquare, normal users prefer locations whose homogeneity metric value 6-8 times higher than OUs. They write fewer words in their reviews, have shorter lists, and post reviews in lower frequency. Foursquare's normal users show the same pattern as Yelp's when it comes to sentiment features. This result indicates that the OUs and normal users are quite distinguishable by those features. Besides, OUs valued by different strategies may share the same pattern compared with normal users. This outcome matches our common sense that users with rich information in the service are more likely to be outstanding.

6 Related Work

6.1 Social Graph-Based User Detection

Identifying the most efficient "spreaders" in a network is essential in many real-world applications, such as optimizing the use of resources and ensuring the information spreads more efficiently. The most straightforward and prevalent detecting strategy is to make use of the structure of the social graph.

For detecting SHS, Lou et al. [32] designed algorithms to identify SHS based on given communities. This solution is not able to work when community boundaries are missing or blurred, which is not unusual. Inspired by the intermingled nature of SHS and user communities, He et al. [19] proposed a harmonic modularity method to detect them simultaneously. However, their algorithms are slow and only fit small graphs due to the high space complexity (i.e., $O(n^2)$). To reduce the running time, Xu et al. [17] later adopted filtering techniques to estimate SHS from articulation points.

For detecting influential users, the criteria are generally based on network structure [35], such as PageRank, Degree, and Betweenness centrality [80]. In applications such as citation impact analysis, measurements like H-index [81] could also be used. Works using methods like Hyperlink-Induced Topic Search (HITS) [40] and PageRank [82] equipped one of the measurements mentioned above to find the most influential set of users. However, when the network is large, a collection of different nodes is likely to be ranked the same by a single metric. Thus, some works like [75] consider multiple metrics to select top-k influentials.

Many previous works [32], [19], [17] discover users of the target category by analyzing network structure features of nodes. Calculating those features, however, requires the structural information of the entire social network. The computation will become time-consuming as the network grows larger while fail to launch when the available network information is incomplete. Instead, our framework merely requires the ego network structures of less than 1% of users in the network, which is easier to acquire. Once trained, our framework could predict a user's category only through accessible information like UGC and user profiles.

6.2 Deep Learning-Based User Detection With Social Networks

Another line of research on online user detecting benefits from the emergence of deep learning methods. Generally, researchers make use of one or both the network structure and the UGC. Several related approaches have been proposed to find some specific types of OU.

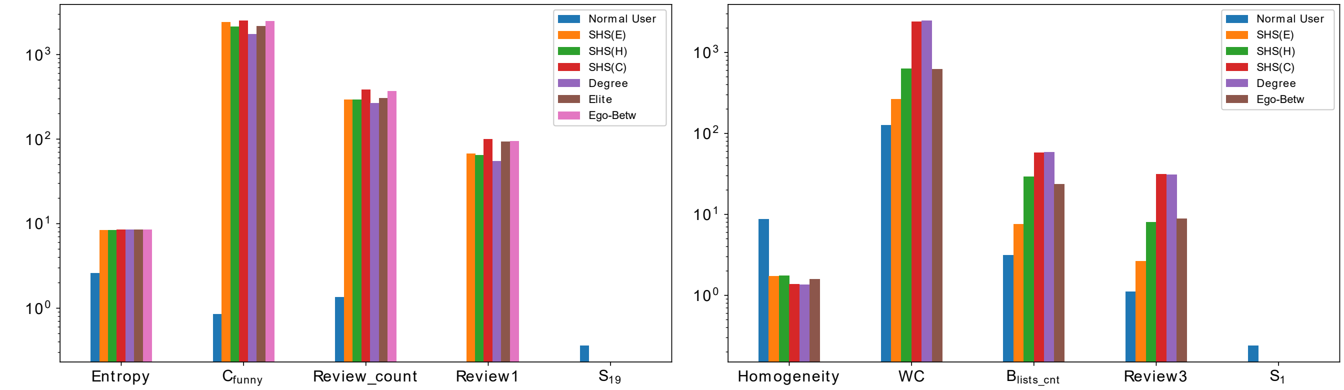
On the one hand, the graph structure is widely utilized. For example, Wei et al. [83] used network representation learning to find overlapping communities in the network, then combined the community information and node topology to rank influential nodes. Keikha et al. [84] proposed DeepIM algorithm to detect the most influential nodes inside and between networks by their local and global structural properties. These works learn to represent the structure of the whole graph by feature vectors. If the input graph is fragmentary, their neural network can either not learn network representations or give representations with large deviations. Different from those works, DeepPick utilizes neural networks to deal with existing UGC and only needs some users' local network information in the training process, which enhances its feasibility.

On the other hand, user actions on online platforms also provide rich information for distinguishing OUs. Such information includes various respects of a user's online lifestyle, e.g. information cascades [76], [18], [85], user interest [86], and user interactions [77], [87]. Deep learning methods are effective in digging latent key information, e.g., text and time series. To make the best use of all dimensions of data, we proposed to incorporate accessible node-level features with UGC, which enhances the scalability of our method.

6.3 Graph Neural Networks

Recently, there has been a surge of interest in graph neural network (GNN) approaches. For each node, GNN recursively updates its representation by aggregating the representation of its neighborhood. After K iterations, the K-hop neighborhood's structural and representation information will be aggregated into the current node's representation. GNNs have been successfully applied to social network mining, including social influence prediction [76], [88], node popularity detection [89], and diffusion prediction [90]. These works all take the embedding vector to solve the link prediction tasks, which is related to our work.

User interaction information is significant to GNN-based works. For instance, [90] predicts diffusion by modeling the joint effect of temporal embeddings and social embeddings, which are learned from users' social connections. Differently, [89] does not rely on temporal information, but still emphasizes the role of the interacting network, i.e.,



(a) Yelp (b) Foursquare

Fig. 7. Case study of each type of OUs compared with normal users. The y-axis is in log scale. This graph shows the mean value of randomly selected 100 users of each category. In Yelp, the value of essential features is similar for different OUs, while normal users show low scores. In Foursquare, the gap between OUs and normal users is also apparent.

the cascading effect along with users' interaction, in popularity prediction problems. [89] aggregates the expected influence a user receives from his/her neighborhood as the evidence in popularity prediction. These works achieve good performance in predicting influence, but do not launch well in the incomplete graphs. We will further discuss the practicability of applying GNN in Section 7.1.

7 Discussion

In this section, we discuss some design issues of our proposed framework. In Section 7.1, we present the probability of adopting GNNs in this task. In Section 7.2, we compare the computational efficiency of our method and several transductive algorithms. We show the extent to which OUs is overlapping with active users in the platforms in Section 7.3.

7.1 Practicability of Graph Neural Network

In our model, the ego network is introduced as a variant of GNN. The learning process of DeepPick is inspired by GNNs but relies much less on the knowledge of network structure to perform well. As GNN does, we also aggregate the features of nodes' neighborhood, but adopt multimodal data of the nodes instead of leveraging the structural information of nodes.

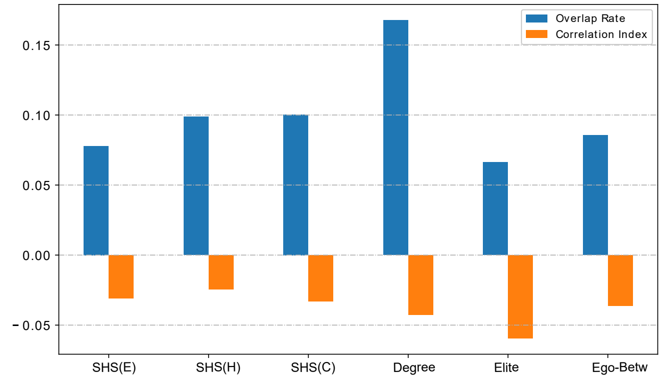
Two obstacles are preventing the direct application of GNNs in this task. First, GNNs represent each node by aggregating its neighborhood's representation via methods like random walks [76], matrix factorization [91], or GCN [92]. The aggregating expands along with the underlying network edges. If there is a partial absence of the network structure, the prediction performance will be degraded [89]. Although some approaches show applicability with the dropout of the network under the premise of network connectivity (e.g., CoupledGNN [89]), they are not applicable when only isolated ego networks are provided, which is a common case due to users' privacy concerns. Second, all the embeddings of nodes are required to be well-trained in GNN. For third-party service providers who desire a quick user distinguishing process, GNN's computation complexity is still high and unacceptable in large-scale online social network data. Thus, a tradeoff between detection performance and efficiency is urgently needed.

7.2 Time Complexity Analysis

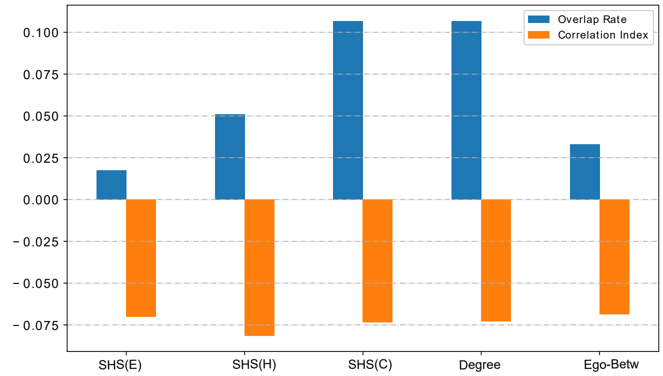
As stated above, we analyze the computational complexity of our approach in the style of GNNs. Let $|V|$ be the number of nodes in network G, K be the dimension of embedding vectors, L be the number of layers, D be the average degree, and s be the number of sampled nodes per layer. For simplicity, assume s remains equivalent across all layers of GCN. For the transductive algorithms like PageRank, all nodes participate in computing. The time complexity of convergence could be $O(t(\epsilon)|V|^2)$, where $t(\epsilon)$ is the number of iterations with convergence threshold ϵ. For the GCN-based algorithms, the training process is about importance sampling. Take node-wise sampling techniques with minibatch training as an example. The sampling algorithm iteratively samples nodes of each layer to form minibatches, then propagate forward and backward among the sampled GCN. In the forward process, for each batch of b nodes, we update $O(s^{L-1})$ activations for each node. As each new activation requires aggregate s embeddings in previous layers, the computation cost for neighborhood propagation in one batch is $O(bKs^{L-1})$. The time complexity of the representative sampling process, e.g., alias table, is $O(D)$. Thus, the overall forward time complexity is $O(bDKs^{L-1})$. In the backward propagation, GCN also needs $O(s^{L-1})$ to update parameters. For our proposed DeepPick framework, only the current central node is sampled to conduct the gradient descent process ($L = 1$), which largely reduces the number of nodes taking part in the training. In that sense, the backward propagation time is reduced to $O(1)$.

7.3 Overlapping with Active Users

Here we want to illustrate the difference between OUs and other highly-mentioned types of users in OSNs, the active users. We first clarify this question via definition, then



(a) Yelp



(b) Foursquare

Fig. 8. Overlap rate and correlation index of top OUs and active users in both datasets. Although a small part of OUs is active users simultaneously, the two groups' number of UGC shows a slightly negative correlation.

quantify the overlap rate and correlation index of the two types of users.

Active users are often those who post more UGC in OSNs [93]. On the other hand, the outstanding users are the first activated node group in the IC model, denoted as A_0 in our paper. When acting as the information source, OUs can spread information in social networks faster than others. Nevertheless, active users will not be able to spread information widely if their UGC are not popular. We analyze the overlap rate and correlation index of different subsets of OUs and active users in each dataset. We rank the users by their numbers of reviews and select the top 10,000 active users, which is of the same number as sampled OUs and Normal users. The percentage of users who are active and outstanding simultaneously is shown in Fig. 8. Yelp's active users are more likely to become OUs, as Yelp offers many awards to encourage posting reviews. We notice that OU with high degree centrality overlaps the most with active users in both datasets. Furthermore, the correlation index between the review number sequence of high degree centrality OUs and active users is small and negative. Thus, we could find that being an active user and being an OU are not highly related. Overall, some weak relationships exist between being an active user and being an OU, but having many UGC is not a guarantee to become an OU.

8 Conclusion and Future Work

In this paper, we defined the concept of OUs and proposed a mixed metric selection strategy to discover them. To manifest its efficiency, we designed and implemented DeepPick, a deep learning-based OUs detection framework. DeepPick only needs a small fraction of users' information (i.e., ego networks, demographic data, UGC, and visited POIs) to train. After bootstrapping, it will be able to detect OUs by their publicly visible information. Extensive experiments on representative OSNs validate the effectiveness of our strategy. Also, DeepPick is compatible with different types of OUs, which is indeed an advantage of our system. According to our evaluation on several example definitions of OUs, DeepPick outperforms the existing solutions. This study sheds light on the path to unveil OUs in OSNs when the network structure is not fully accessible, which is a common case. It is useful for different relevant entities, such as OSN operators, Internet service providers, third-party service providers, and academic researchers.

In future works, we will further study the effectiveness of combining more features in the framework. There are two types of features that can be considered. First, in the fields of diffusion prediction, social homophily and temporal influence are shown to be crucial indicators [90]. We would like to examine if they are also likely to provide more information about the outstanding users. Second, popularity evaluation [89] of outstanding users is also worth exploration. We will study how the users' popularity level correlated with their outstanding features.

Acknowledgement

This work is sponsored by National Natural Science Foundation of China (No. 62072115, No. 71731004, No. 61602122, No. 61971145), CERNET Innovation Project (NGII20190105), the project "PCL Future Greater-Bay Area Network Facilities for Large-scale Experiments and Applications (LZC0019)", the Research Grants Council of Hong Kong (No.16214817), the 5GEAR project and FIT project from the Academy of Finland. The work is also partially supported by the European Union's Horizon 2020 research and innovation programme under the grant agreement No. 101021808, and iSafe project funded by TU Delft Safety & Security Institute. Yang Chen is the corresponding author.

References

[1] A. Hicks, S. Comp, J. Horovitz, M. Hovarter, M. Miki, and J. L. Bevan, "Why People Use Yelp.com: An Exploration of Uses and Gratifications," *Computers in Human Behavior*, vol. 28, no. 6, pp. 2274–2279, 2012.

[2] J. W. Byers, M. Mitzenmacher, and G. Zervas, "The Groupon Effect on Yelp Ratings: A Root Cause Analysis," in *Proceedings of the 13th ACM Conference on Electronic Commerce*, 2012, pp. 248–265.

[3] Y. D. Kwon, D. Chatzopoulos, R. C.-W. Wong, P. Hui *et al.*, "GeoLifecycle: User Engagement of Geographical Exploration and Churn Prediction in LBSNs," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, no. 3, p. 92, 2019.

[4] Q. Gong, Y. Chen, J. Hu, Q. Cao, P. Hui, and X. Wang, "Understanding Cross-Site Linking in Online Social Networks," *ACM Transactions on the Web (TWEB)*, vol. 12, no. 4, pp. 1–29, 2018.

[5] A. Noulas, B. Shaw, R. Lambiotte, and C. Mascolo, "Topological Properties and Temporal Dynamics of Place Networks in Urban Environments," in *Proceedings of the 24th International Conference on World Wide Web*, 2015, pp. 431–441.

[6] G. Wang, S. Y. Schoenebeck, H. Zheng, and B. Y. Zhao, ""Will Check-in for Badges": Understanding Bias and Misbehavior on Location-Based Social Networks," in *Tenth International AAAI Conference on Web and Social Media*, 2016.

[7] M. Cha, H. Haddadi, F. Benevenuto, and K. P. Gummadi, "Measuring User Influence in Twitter: The Million Follower Fallacy," in *Fourth International AAAI Conference on Weblogs and Social Media*, 2010.

[8] H. Wang, Q. Meng, J. Fan, Y. Li, L. Cui, X. Zhao, C. Peng, G. Chen, and X. Du, "Social Influence Does Matter: User Action Prediction for In-Feed Advertising," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, 2020, pp. 246–253.

[9] J. Goldenberg, B. Libai, and E. Muller, "Talk of the Network: A Complex Systems Look at the Underlying Process of Word-of-Mouth," *Marketing Letters*, vol. 12, no. 3, pp. 211–223, 2001.

[10] R. S. Burt, *Structural Holes: The Social Structure of Competition*. Harvard University Press, 2009.

[11] L. C. Freeman, "Centrality in Social Networks Conceptual Clarification," *Social Networks*, vol. 1, no. 3, pp. 215–239, 1978.

[12] F. Riquelme and P. González-Cantergiani, "Measuring User Influence on Twitter: A Survey," *Information Processing & Management*, vol. 52, no. 5, pp. 949–975, 2016.

[13] M. Moricz, Y. Dosbayev, and M. Berlyant, "PYMK: Friend Recommendation at Myspace," in *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, 2010, pp. 999–1002.

[14] X. Hu, S. Liu, Y. Zhang, G. Zhao, and C. Jiang, "Identifying Top Persuaders in Mixed Trust Networks for Electronic Marketing Based on Word-of-mouth," *Knowledge-Based Systems*, vol. 182, p. 104803, 2019.

[15] Q. Li, B. Kailkhura, J. Thiagarajan, Z. Zhang, and P. Varshney, "Influential Node Detection in Implicit Social Networks Using Multi-task Gaussian Copula Models," in *NIPS 2016 Time Series Workshop*, 2017, pp. 27–37.

[16] 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*, 2010, pp. 591–600.

[17] W. Xu, M. Rezvani, W. Liang, J. X. Yu, and C. Liu, "Efficient Algorithms for the Identification of Top-k Structural Hole Spanners in Large Social Networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 5, pp. 1017–1030, 2017.

[18] A. K. Bhowmick, M. Gueuning, J.-C. Delvenne, R. Lambiotte, and B. Mitra, "Temporal Sequence of Retweets Help to Detect Influential Nodes in Social Networks," *IEEE Transactions on Computational Social Systems*, 2019.

[19] L. He, C.-T. Lu, J. Ma, J. Cao, L. Shen, and P. S. Yu, "Joint Community and Structural Hole Spanner Detection via Harmonic Modularity," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016, pp. 875–884.

[20] K. Musiał, P. Kazienko, and P. Bródka, "User Position Measures in Social Networks," in *Proceedings of the 3rd Workshop on Social Network Mining and Analysis*. ACM Paris, France, 2009, pp. 1–9.

[21] J. W. Pennebaker, M. E. Francis, and R. J. Booth, "Linguistic Inquiry and Word Count: LIWC 2001," *Mahway: Lawrence Erlbaum Associates*, vol. 71, no. 2001, p. 2001, 2001.

[22] M. Gladwell, *The Tipping Point: How Little Things Can Make a Big Difference*. Little, Brown, 2006.

[23] J. S. Coleman, *Foundations of Social Theory*. Harvard University Press, 1994.

[24] J. Nahapiet and S. Ghoshal, "Social Capital, Intellectual Capital, and the Organizational Advantage," *Academy of Management Review*, vol. 23, no. 2, pp. 242–266, 1998.

[25] R. S. Burt, "Closure as Social Capital," *Social Capital: Theory and Research*, pp. 31–55, 2001.

[26] M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley, and H. A. Makse, "Identification of Influential Spreaders in Complex Networks," *Nature Physics*, vol. 6, no. 11, p. 888, 2010.

[27] J. M. Podolny and J. N. Baron, "Resources and Relationships: Social Networks and Mobility in the Workplace," *American Sociological Review*, pp. 673–693, 1997.

[28] R. S. Burt, "Secondhand Brokerage: Evidence on the Importance of Local Structure for Managers, Bankers, and Analysts," *Academy of Management Journal*, vol. 50, no. 1, pp. 119–148, 2007.

[29] Burt, Ronald S, "Structural Holes and Good Ideas," *American journal of sociology*, vol. 110, no. 2, pp. 349–399, 2004.

[30] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the Spread of Influence through a Social Network," in *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003, pp. 137–146.

[31] Y. Li, J. Fan, Y. Wang, and K.-L. Tan, "Influence Maximization on Social Graphs: A Survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 10, pp. 1852–1872, 2018.

[32] T. Lou and J. Tang, "Mining Structural Hole Spanners Through Information Diffusion in Social Networks," in *Proceedings of the 22nd International Conference on World Wide Web*. ACM, 2013, pp. 825–836.

[33] M. P. Van Den Heuvel and O. Sporns, "Rich-Club Organization of the Human Connectome," *Journal of Neuroscience*, vol. 31, no. 44, pp. 15 775–15 786, 2011.

[34] Z. Lin, Y. Zhang, Q. Gong, Y. Chen, A. Oksanen, and A. Y. Ding, "Structural Hole Theory in Social Network Analysis: A Review," *IEEE Transactions on Computational Social Systems*, pp. 1 – 16, 2021.

[35] A. Raychaudhuri, S. Mallick, A. Sircar, and S. Singh, "Identifying Influential Nodes Based on Network Topology: A Comparative Study," in *Information, Photonics and Communication*. Springer, 2020, pp. 65–76.

[36] S. Chechik, E. Cohen, and H. Kaplan, "Average Distance Queries through Weighted Samples in Graphs and Metric Spaces: High Scalability with Tight Statistical Guarantees," in *the Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM'15)*, vol. 40. LIPIcs, 2015, p. 659–679.

[37] S. Goyal and F. Vega-Redondo, "Structural Holes in Social Networks," *Journal of Economic Theory*, vol. 137, no. 1, pp. 460–492, 2007.

[38] V. Arnaboldi, M. Conti, M. La Gala, A. Passarella, and F. Pezzoni, "Information Diffusion in OSNs: The Impact of Nodes' Sociality," in *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, 2014, pp. 616–621.

[39] Y. Yang, Z. Wang, T. Jin, J. Pei, and E. Chen, "Tracking Top-k Influential Users with Relative Errors," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. ACM, 2019, pp. 1783–1792.

[40] C. S. Campbell, P. P. Maglio, A. Cozzi, and B. Dom, "Expertise Identification Using Email Communications," in *Proceedings of the Twelfth International Conference on Information and Knowledge Management*, 2003, pp. 528–531.

[41] M. Everett and S. P. Borgatti, "Ego Network Betweenness," *Social Networks*, vol. 27, no. 1, pp. 31–38, 2005.

[42] P. V. Marsden, "Egocentric and Sociocentric Measures of Network Centrality," *Social Networks*, vol. 24, no. 4, pp. 407–422, 2002.

[43] Y. Chen, J. Hu, Y. Xiao, X. Li, and P. Hui, "Understanding the User Behavior of Foursquare: A Data-Driven Study on a Global Scale," *IEEE Transactions on Computational Social Systems*, 2020.

[44] G. Ghoshal and A.-L. Barabási, "Ranking Stability and Super-Stable Nodes in Complex Networks," *Nature communications*, vol. 2, p. 394, 2011.

[45] D. Hristova, M. J. Williams, M. Musolesi, P. Panzarasa, and C. Mascolo, "Measuring Urban Social Diversity Using Interconnected Geo-Social Networks," in *Proceedings of the 25th International Conference on world wide web*. International World Wide Web Conferences Steering Committee, 2016, pp. 21–30.

[46] J. Cranshaw, E. Toch, J. Hong, A. Kittur, and N. Sadeh, "Bridging the Gap Between Physical Location and Online Social Networks," in *Proceedings of the 12th ACM International Conference on Ubiquitous Computing*. ACM, 2010, pp. 119–128.

[47] A. Chin and D. Zhang, *Mobile Social Networking. An Innovative Approach*. Springer-Verlag, 2013.

[48] 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.

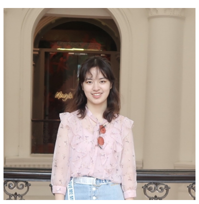
[49] J. Pang and Y. Zhang, "Quantifying Location Sociality," in *Proceedings of the 28th ACM Conference on Hypertext and Social Media*. ACM, 2017, pp. 145–154.

[50] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding Deep Learning Requires Rethinking Generalization," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

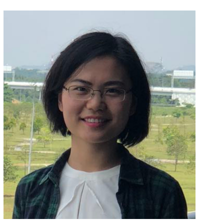
[51] J. McAuley and J. Leskovec, "Hidden Factors and Hidden Topics: Understanding Rating Dimensions with Review Text," in *Proceedings of the 7th ACM Conference on Recommender systems*, 2013, pp. 165–172.
[52] Y. Yao, B. Viswanath, J. Cryan, H. Zheng, and B. Y. Zhao, "Automated Crowdturfing Attacks and Defenses in Online Review Systems," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 1143–1158.
[53] Y. R. Tausczik and J. W. Pennebaker, "The Psychological Meaning of Words: LIWC and Computerized Text Analysis Methods," *Journal of Language and Social Psychology*, vol. 29, no. 1, pp. 24–54, 2010.
[54] J. W. Pennebaker, C. K. Chung, J. Frazee, G. M. Lavergne, and D. I. Beaver, "When Small Words Foretell Academic," *PloS one*, vol. 9, no. 12, p. e115844, 2014.
[55] Q. Gong, Y. Chen, X. He, Z. Zhuang, T. Wang, H. Huang, X. Wang, and X. Fu, "DeepScan: Exploiting Deep Learning for Malicious Account Detection in Location-based Social Networks," *IEEE Communications Magazine*, vol. 56, no. 11, pp. 21–27, 2018.
[56] Q. Gong, J. Zhang, Y. Chen, Q. Li, Y. Xiao, X. Wang, and P. Hui, "Detecting Malicious Accounts in Online Developer Communities Using Deep Learning," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. ACM, 2019, pp. 1251–1260.
[57] L. Breiman, *Classification and Regression Trees*. Routledge, 2017.
[58] Breiman, Leo, "Random Rorests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
[59] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016, pp. 785–794.
[60] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: Unbiased Boosting with Categorical Features," in *Advances in Neural Information Processing Systems*, 2018, pp. 6638–6648.
[61] Y. Kim, "Convolutional Neural Networks for Sentence Classification," in *2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2014, p. 1746–1751.
[62] S. Hochreiter and J. Schmidhuber, "Long Short-term Memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[63] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and Their Compositionality," in *Advances in Neural Information Processing Systems*, 2013, pp. 3111–3119.
[64] Y. Bengio, P. Simard, P. Frasconi *et al.*, "Learning Long-Term Dependencies with Gradient Descent is Difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.
[65] A. Graves and J. Schmidhuber, "Framewise Phoneme Classification with Bidirectional LSTM and Other Neural Network Architectures," *Neural Networks*, vol. 18, no. 5-6, pp. 602–610, 2005.
[66] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," in *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*. ICLR Workshop, 2013.
[67] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, no. Oct, pp. 2825–2830, 2011.
[68] T. Fawcett, "An Introduction to ROC Analysis," *Pattern recognition letters*, vol. 27, no. 8, pp. 861–874, 2006.
[69] C. D. Manning, H. Schütze, and P. Raghavan, *Introduction to Information Retrieval*. Cambridge university press, 2008.
[70] D. Neil, M. Pfeiffer, and S.-C. Liu, "Phased LSTM: Accelerating Recurrent Network Training for Long or Event-based Sequences," in *Advances in Neural Information Processing Systems*, 2016, pp. 3882–3890.
[71] Q. McNemar, "Note on the Sampling Error of the Difference Between Correlated Proportions or Percentages," *Psychometrika*, vol. 12, no. 2, pp. 153–157, 1947.
[72] J. Ma, W. Gao, P. Mitra, S. Kwon, B. J. Jansen, K.-F. Wong, and M. Cha, "Detecting Rumors from Microblogs with Recurrent Neural Networks," in *the Twenty-Fifth International Joint Conference on Artificial Intelligence. IJCAI*, 2016, pp. 3818–3824.
[73] G. Katsimpras, D. Vogiatzis, and G. Paliouras, "Determining Influential Users with Supervised Random Walks," in *Proceedings of the 24th International Conference on World Wide Web*. ACM, 2015, pp. 787–792.
[74] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *Journal of Machine Learning Research*, vol. 3, no. Jan, pp. 993–1022, 2003.
[75] A. Sheikhahmadi and M. A. Nematbakhsh, "Identification of Multi-spreader Users in Social Networks for Viral Marketing," *Journal of Information Science*, vol. 43, no. 3, pp. 412–423, 2017.
[76] J. Qiu, J. Tang, H. Ma, Y. Dong, K. Wang, and J. Tang, "DeepInf: Social Influence Prediction with Deep Learning," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2018, pp. 2110–2119.
[77] H. Bo, R. McConville, J. Hong, and W. Liu, "Social Network Influence Ranking via Embedding Network Interactions for User Recommendation," in *Companion Proceedings of the Web Conference 2020*, 2020, pp. 379–384.
[78] L. F. Ribeiro, P. H. Saverese, and D. R. Figueiredo, "struc2vec: Learning Node Representations from Structural Identity," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, pp. 385–394.
[79] C. Donnat, M. Zitnik, D. Hallac, and J. Leskovec, "Learning Structural Node Embeddings Via Diffusion Wavelets," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2018, pp. 1320–1329.
[80] G. Amati, S. Angelini, G. Gambosi, G. Rossi, and P. Vocca, "Influential Users in Twitter: Detection and Evolution Analysis," *Multimedia Tools and Applications*, vol. 78, no. 3, pp. 3395–3407, 2019.
[81] J. E. Hirsch, "An Index to Quantify an Individual's Scientific Research Output," *Proceedings of the National Academy of Sciences*, vol. 102, no. 46, pp. 16 569–16 572, 2005.
[82] Y. Ding, E. Yan, A. Frazho, and J. Caverlee, "PageRank for Ranking Authors in Co-citation Networks," *Journal of the American Society for Information Science and Technology*, vol. 60, no. 11, pp. 2229–2243, 2009.
[83] H. Wei, Z. Pan, G. Hu, L. Zhang, H. Yang, X. Li, and X. Zhou, "Identifying Influential Nodes Based on Network Representation Learning in Complex Networks," *PloS one*, vol. 13, no. 7, 2018.
[84] M. M. Keikha, M. Rahgozar, M. Asadpour, and M. F. Abdollahi, "Influence Maximization Across Heterogeneous Interconnected Networks Based on Deep Learning," *Expert Systems with Applications*, vol. 140, p. 112905, 2020.
[85] C. K. Leung, A. Cuzzocrea, J. J. Mai, D. Deng, and F. Jiang, "Personalized DeepInf: Enhanced Social Influence Prediction with Deep Learning and Transfer Learning," in *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, 2019, pp. 2871–2880.
[86] F. Erlandsson, P. Bródka, A. Borg, and H. Johnson, "Finding Influential Users in Social Media Using Association Rule Learning," *Entropy*, vol. 18, no. 5, p. 164, 2016.
[87] A. Guille, H. Hacid, C. Favre, and D. A. Zighed, "Information Diffusion in Online Social Networks: A Survey," *ACM Sigmod Record*, vol. 42, no. 2, pp. 17–28, 2013.
[88] G. Zhao, P. Jia, A. Zhou, and B. Zhang, "InfGCN: Identifying Influential Nodes in Complex Networks with Graph Convolutional Networks," *Neurocomputing*, vol. 414, pp. 18–26, 2020.
[89] Q. Cao, H. Shen, J. Gao, B. Wei, and X. Cheng, "Popularity Prediction on Social Platforms with Coupled Graph Neural Networks," in *Proceedings of the 13th International Conference on Web Search and Data Mining*, 2020, pp. 70–78.
[90] A. Sankar, X. Zhang, A. Krishnan, and J. Han, "Inf-VAE: A Variational Autoencoder Framework to Integrate Homophily and Influence in Diffusion Prediction," in *Proceedings of the 13th International Conference on Web Search and Data Mining*, 2020, pp. 510–518.
[91] M. Ou, P. Cui, J. Pei, Z. Zhang, and W. Zhu, "Asymmetric Transitivity Preserving Graph Embedding," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1105–1114.
[92] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling, "Modeling Relational Data with Graph Convolutional Networks," in *European Semantic Web Conference*. Springer, 2018, pp. 593–607.
[93] A. Chen, Y. Lu, P. Y. Chau, and S. Gupta, "Classifying, Measuring, and Predicting Users' Overall Active Behavior on Social Networking Sites," *Journal of Management Information Systems*, vol. 31, no. 3, pp. 213–253, 2014.

Wanda Li received her B.S. degree (with honor) from the School of Computer Science, Fudan University. She has been a research assistant in the Mobile Systems and Networking (MSN) group since 2018. She is now a graduate student in data science and information technology at Tsinghua-Berkeley Shenzhen Institute (TBSI). Her research interests include data mining, machine learning, and user behavior analysis.

Zhiwei Xu is currently working toward the Bachelor degree in the School of Computer Engineering and Science, Shanghai University. He has been a research intern in the Mobile Systems and Networking (MSN) group at Fudan University since 2019. His research interests include online social network mining, user behavior modeling and machine learning systems.

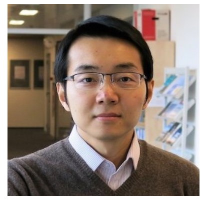


Yi Sun is an undergraduate in Computer Science at Huazhong University of Science and Technology. She has been a research intern in the Mobile Systems and Networking (MSN) group at Fudan University since 2019. Her research interests include online social network mining and deep learning.



Qingyuan Gong received her PhD degree in Computer Science at Fudan University in 2020. She is now working as a Postdoc at Fudan University. Her research interests include network security, user behavior analysis and computational social systems. She published referred papers in IEEE Communications Magazine, ACM TWEB, IEEE TDSC, Springer WWW Journal, ACM CIKM and ICPP. She has been a visiting student at the University of Göttingen in 2015 and 2019, also at the University of Chicago in 2018.

Yang Chen is an Associate Professor within the School of Computer Science at Fudan University, China. He leads the Mobile Systems and Networking (MSN) group since 2014. Before joining Fudan, he was a postdoctoral associate at the Department of Computer Science, Duke University, USA, where he served as Senior Personnel in the NSF MobilityFirst project. From September 2009 to April 2011, he has been a research associate and the deputy head of Computer Networks Group, Institute of Computer Science, University of Göttingen, Germany. He received his B.S. and Ph.D. degrees from Department of Electronic Engineering, Tsinghua University in 2004 and 2009, respectively. He visited Stanford University (in 2007) and Microsoft Research Asia (2006-2008) as a visiting student. His research interests include online social networks, Internet architecture and mobile computing. He serves as an Associate Editor-in-Chief of the Journal of Social Computing. He is a senior member of the IEEE.



Aaron Yi Ding is leading the Cyber-Physical Intelligence (CPI) Lab at TU Delft as Tenured Asst Professor and Adjunct Professor (permanent) in Computer Science at University of Helsinki. Prior to TU Delft, he has worked at TU Munich in Germany, Columbia University in the USA, and University of Cambridge in the UK. His research focuses on edge computing, edge AI, and data-driven IoT services. He obtained his PhD from the Department of Computer Science at University of Helsinki. His research work has received best paper awards and recognition from ACM SIGCOMM, ACM EdgeSys, ACM SenSys CCIoT, and IEEE INFOCOM. He is a two-time recipient of the Nokia Foundation Scholarships.

Xin Wang is a professor at Fudan University, Shanghai, China. He received his BS Degree in Information Theory and MS Degree in Communication and Electronic Systems from Xidian University, China, in 1994 and 1997, respectively. He received his Ph.D. Degree in Computer Science from Shizuoka University, Japan, in 2002. His research interests include quality of network service, next-generation network architecture, mobile Internet and network coding.

Pan Hui is the Nokia Chair in Data Science and a full Professor in Computer Science at the University of Helsinki since September 2017. He is also a faculty member of the Department of Computer Science and Engineering at the Hong Kong University of Science and Technology since 2013 and an adjunct Professor of social computing and networking at Aalto University Finland since 2012. He received his Ph.D. degree from Computer Laboratory, University of Cambridge, and earned his MPhil and BEng both from the Department of Electrical and Electronic Engineering, University of Hong Kong. He has published over 200 research papers with over 12,500 citations and has around 30 granted / filed European patents. He is an associate editor for IEEE Transactions on Mobile Computing and IEEE Transactions on Cloud Computing, a guest editor for IEEE Communication Magazine, an IEEE Fellow, and an ACM Distinguished Scientist.