




Article

Identifying Human Daily Activity Types with Time-Aware Interactions

Ren Yao Chen ¹, Hong Yao ² , Runjia Li ², Xiaojun Kang ², Shengwen Li ^{1,3} , Lijun Dong ² and Junfang Gong ^{1,4,*} 

¹ School of Geography and Information Engineering, China University of Geosciences, Wuhan 430074, China; cryao@cug.edu.cn (R.C.); swli@cug.edu.cn (S.L.)

² School of Computer Science, China University of Geosciences, Wuhan 430074, China; yaohong@cug.edu.cn (H.Y.); lirunjia@cug.edu.cn (R.L.); kangxj@cug.edu.cn (X.K.); ljdong@cug.edu.cn (L.D.)

³ National Engineering Research Center for Geographic Information System, China University of Geosciences, Wuhan 430074, China

⁴ Key Laboratory of Urban Land Resources Monitoring and Simulation, Ministry of Natural Resources, Shenzhen 518034, China

* Correspondence: jfgong@cug.edu.cn; Tel.: +86-027-6788-3728

Received: 13 November 2020; Accepted: 9 December 2020; Published: 14 December 2020



Abstract: Human activities embedded in crowdsourced data, such as social media trajectory, represent individual daily styles and patterns, which are valuable in many applications. However, the accurate identification of human activity types (HATs) from social media is challenging, possibly because interactions between posts and users at different time are overlooked. To fill this gap, we propose a novel model that introduces the interactions hidden in social media and synthesizes Graph Convolutional Network (GCN) for identifying HAT. The model first characterizes interactions among words, posts, dates, and users, and then derives a Time Gated Human Activity Graph Convolutional Network (TG-HAGCN) to predict the HATs of social media trajectory. To examine the proposed model performance, we built a new dataset including interactions between post content, post time, and users from the open Yelp dataset. Experimental results show that exploiting interactions hidden in social media to recognize HATs achieves state-of-the-art performance with high accuracy. The study indicates that interactions among social media promotes ability of machine learning on social media data mining and intelligent applications, and offers a reference solution for how to fuse multi-type heterogeneous data in social media.

Keywords: human activity recognition; social media; Graph Convolutional Network

1. Introduction

Urban spaces, where citizens live, move, and engage in different activities, are socialized, and dynamic [1]. Understanding the complexities underlying the emerging behaviors of human travel patterns on the city level is essential for making informed decision-making pertaining to urban transportation infrastructures [2]. Emerging crowdsourced data provides available data sources for capturing human behaviors and modeling urban systems. The human activity types (HATs) (e.g., sports, shopping, and travel) in human trajectory characterize the user's daily behaviors, and therefore, present human lifestyles and patterns [3,4]. The trajectory with HATs contributes not only to understanding human society and urban systems, but also to performing many tasks for different purposes, such as customer recommendation, traffic forecasting, travel demand modeling, urban planning, and so on. However, identifying HATs from the trajectory on a large scale by experts is always labor-intensive and time-consuming. Hence, it is significant to develop an approach, which can automatically identify accurate HATs on large-scale.

Benefiting from the advancement of machine learning technology, many studies have promoted the performance of automatically identifying [5,6]. These studies regard HAT extraction as an ordinary text classification and focus on improving its performance using text contents and attributes, such as post time and position. However, the accurate identification of HATs from the trajectory of social media remains challenging, possibly because interactions between humans and posts at different times are overlooked. It should be noted that social media posts are not independent. Several interactions exist among words, posts, dates, and users, as shown in Figure 1. These interactions may contribute to the accurate identification of HATs from social media. Besides, previous models mainly take trajectory text as samples, and are not capable of fusing heterogeneous data that are not at a one-to-one correspondence with the samples for predicting HATs.



Figure 1. Interactions among words, posts, dates, and users.

This study aims to improve the prediction accuracy of HATs by characterizing time-aware interactions in social media. To the best of our knowledge, this study is the first to highlight the interactions in social media and introduce a graph convolutional model for predicting HATs. The original contributions of this study are as follows:

- (1) For effectively modeling the interactions hidden in social media, this study models the implicit interactions from two views: text view and time view. With two views, this model not only contacts the post content and users, but also helps capture the user's activity patterns at different times with a date encoding method.
- (2) To synthesize time-aware interactions and trajectory texts in social media, a Time Gated Human Activity Graph Convolutional Network (TG-HAGCN) is developed to fuse the embeddings of two GCN components through a gating mechanism. It offers a reference solution for how to fuse multi-type heterogeneous data with multiple views.
- (3) The proposed method achieves state-of-the-art performance in identifying HATs, with high accuracy, and less time expenditure. The results indicate that time-aware interactions among social media promote ability of machine learning on social media data mining and intelligent applications.

The rest of this paper is organized as follows. Section 2 overviews related works. Section 3 demonstrates the proposed approach in detail. We introduce the datasets and experimental results in Section 4, and discuss the effects from hyperparameters and network components in Section 5. Section 6 concludes this study.

2. Related Work

One goal of activity recognition research is to identify physical activities [7–11]. They can be used for human trajectory tracking, recommended system, regions-of-interest detection [12,13], and so on. Researchers use popular machine learning methods, such as support vector machine (SVM), convolutional neural network (CNN), and long short-term memory network (LSTM) to identify human activities and propose a fixed approach that treats activities as sequential text or multi-dimensional images, and uses context-sensitive grammar to identify physical activities. Actually, this research heavily depends on video cameras, wearable sensor, or monitors, and human behavior in urban scope is difficult to capture because data collection is time consuming and maybe conflicts with peoples' privacy.

Different from physical activities and events, human daily activities are high-level logical activities that can be obtained from many types of crowdsourced data. These data are easily obtained and closely connected with human behavior and urban systems. In this paper, we aim to identify human daily activities (e.g., sports, shopping, and travel) from social media trajectory. User's daily logical activities can be inferred from social media through several methods. These works can be classified into two groups, with and without Point of Interest (POI).

In works with POI, the trajectory data under study contain or correlate directly with POI data. Given the close correlation between POIs and human activities, information from POIs can be leveraged to infer HATs. Some studies used activities and POIs interchangeably [14] because activities performed at a location usually define its meaning to some degree [15] (e.g., café is a place where people drink coffee). A common practice is to use semantic location labels as a proxy to high-level activities (e.g., office → working) [16,17]. Dominant functional places, such as home or work can be detected from people's daily routines [18]. Thus, such an approach is an understandable simplification. Nevertheless, a considerable number of locations can be a scene of more than one activity at the same moment (e.g., at a shopping mall), and one activity can be a characteristic of various locations (e.g., "taking picture"). Defu Lian [19] attempted to discover the relationship categories between POIs and activities. However, most trajectory data in social media do not contain location or POIs because of limitations in privacy, devices, and so on.

Works without POIs mostly regard HATs as a text classification task in machine learning. Marco Aurelio Beber integrated trajectory data and social media to reveal individual activities in daily life [20]. However, extracting knowledge from Twitter and matching trajectories to knowledge base. Weerkamp tried to establish a set of activities that will occur at a future time such as tonight, tomorrow, and next week, by using a future time-window and keywords related to activities [21]. They only extracted unexpected activities, but ignored frequently recurring activities. Song proposed a novel collaborative boosting framework comprising a text-to-activity classifier for each user to recognize user activities [22]. Zhu built a multi-label classifier using tweets manually annotated with activities to predict up to three activities [23]. The classifier considers the tweet text, the tweet posting time, the POI type from Foursquare, and the POI name from Foursquare. Their approach relies on crowdsourced annotations with the limited scalability of human labor involved in the process. Mazumder interpreted the concept of activities as verb-noun phrase pairs (e.g., drinking coffee), which they extracted using a pattern-based paradigm [24]. Those methods applying various supervised machine learning or natural language processing (NLP) techniques can capture semantic information in local consecutive word sequences well. However, the texts of trajectory in social media are short and are not always connected directly with HATs, which makes it extremely difficult to accurately infer a user's HAT by using only text features.

HATs should be connected with interactions among social media, such as the posts of one user may have the same text style. However, the above researches overlook their interaction that may contribute to identifying HAT. To fill the gap of, the study tries combine texts and interactions in social media trajectory to improve the performance of HAT prediction. Compared with text classification works based on graph neural networks (GNNs), HAT related interactions involve users and time, in addition to interactions between documents and words. This study constructs two GCN modules from two different views, and finally fuses them through a gate mechanism to achieve HAT recognition. The details will be introduced in Section 3.

3. Methods

This study proposes a model that introduces the interactions in social media into HAT identification. As shown in Figure 1, most objects in social media are one of four types, word, posts, users, and dates. We first build interactions under a text view, including between words and posts (word–post interaction), between words and words (word–word interaction), and between a user and a post (user–post interaction). Therefore, through these interactions, we can get the “user–post–word” graph under the text view and the “user–date–word” graph (built by replacing the post nodes in the “user–post–word” graph with the date nodes) under a time view. Then, a graph neural network-based method for human activity recognition is described, which is derived from Graph Convolutional Network (GCN) [25], named Time Gated Human Activity Graph Convolutional Network (TG-HAGCN). In the following subsections, we will explain why and how to model the three types of interactions and how to build the TG-HAGCN model to identify human daily activity types.

3.1. Modeling Words Interaction Based on Point-Wise Mutual Information (PMI)

The study uses word–word interaction to present the relationship between words. In classical NLP, the co-occurrence of two words reflects the two words that have a close connection, especially when two words co-occur in near location. It is highly similar to the first law of geography [26], whose assumption is that everything is related to everything else, but near things are more related to each other. To gather the word co-occurrence information, we use a fixed-size sliding window for each post. Then, the weight of a word–word interaction is calculated by PMI [27], a popular measure for word associations, as follows:

$$\begin{aligned} PMI(i, j) &= \log \frac{p(i, j)}{p(i)p(j)}, \\ p(i, j) &= \frac{\#W(i, j)}{\#W}, \\ p(i) &= \frac{\#W(i)}{\#W}, \end{aligned} \quad (1)$$

where i and j present two different words, $PMI(i, j)$ is the weight of interaction between words i and j , $\#W(i)$ is the number of sliding windows in posts that contain word i , $\#W(i, j)$ is the number of sliding windows that contain words i and j , and $\#W$ is the total number of sliding windows in the whole dataset. If $PMI(i, j)$ is larger than 0, then a high semantic correlation exists between word i and j , and vice versa.

The point-wise mutual strategy builds the relation among the word pairs. Thus, many interactions will have a negative weight. Obviously, more interactions mean more time costs for follow-up operations. Thus, we only add an interaction between two words when their PMI value is positive, because a negative PMI means little or no relation between two words.

3.2. Modeling Word-Post Interaction Based on Term Frequency–Inverse Document Frequency (TFIDF)

The post–word interaction is built between a word and a post when the word occurs in the post. The number of word occurrence in posts reflects the relation between posts and words, and can be derived to examine the weight of post–word interaction. Actually, each word in a post may contribute differently to human activity recognition because some words occur in many posts, whereas others

occur only in a few posts. A word that occurs in a few posts should contribute more semantics than others [28].

Based on this idea, we employ the Term Frequency–Inverse Document Frequency (TFIDF) model to calculate the weight of the post–word edge. Here, term frequency is the number of times that the word appears in the post, and inverse document frequency is the logarithmically scaled inverse fraction of the number of posts that contain the word. Using TFIDF weight is better than using term frequency only. The mathematical expressions of TF , IDF , and $TFIDF$ are defined as follows:

$$\begin{aligned} TF_w &= \frac{\#w \text{ appears in a post}}{\#w \text{ in a post}}, \\ IDF &= \log\left(\frac{\# \text{ posts in whole corpus}}{\# \text{ posts containing word } w + 1}\right), \\ TFIDF &= TF * IDF. \end{aligned} \quad (2)$$

3.3. Modeling User–Post Interaction Based on Frequency

Each user has his or her own activity pattern and special expression style in text, which means that the activity in a post could be inferred better by introducing the related user information. Thus, we build a user–post interaction between a post and its owner.

All of the words in the posts are collected, forming a vocabulary set. The weight of user–post interaction is calculated by the number of words in the post that appear in the vocabulary set, which reflects the post importance of a user. The equation to calculate user–post interaction weight is derived as follows:

$$UC_{ij} = \frac{S^i(j)}{\sum_{k=1}^n \sum_{t=1}^m S^t(k)}, \quad (3)$$

where $S^i(j)$ represents the function that counts the number of words in post j published by user i , n is the number of users, and m is the number of posts published by the same user k .

In the study, the user only links to post, and no edges exist between users and words, because the relationship between users and words could be indirectly associated with connected posts.

3.4. Identify HATs

The unique capability of graphs enables capturing the structural relations among posts, thereby allowing us to harvest more insights compared with analyzing post texts in isolation. As shown in Figure 1, a heterogeneous graph module can be formed by taking the words, posts, dates, and users as nodes and the interactions between them as edges. Through the three types of interactions defined in Sections 3.1–3.3, we can get the “user–post–word” graph. The number of nodes in the graph is the number of posts (dataset size) plus the number of unique words (vocabulary size) and the number of users. Then, the “user–date–word” graph module can be built by replacing the post nodes in the “user–post–word” graph with the date nodes. The built “user–post–word” graph and “user–date–word” graph can be defined as $G_{post} = (V_{post}, E_{post})$ and $G_{date} = (V_{date}, E_{date})$, respectively, where V and E are the sets of nodes and edges.

Recently emerging GCN [25] model can capture the local features and classify nodes in graphs effectively, which is a multiple layer neural network that generally operates a graph network and generates embedding vectors for nodes based on the properties of their neighbor nodes. It has made a significant improvement in node classification and provides a new opportunity to achieve our task. Thus, the local and global node co-occurrence can be explicitly modeled. Meanwhile, the relation between four types of nodes (users, posts, dates, and words) can be modeled. As a result, human activity types can be examined from local and global factors.

In this paper, the built graphs (“user–post–word” graph under text view and “user–date–word” graph under time view) are fed into a multi-view activity GCN (TG-HAGCN) as illustrated in Figure 2. Thus, the identification of human activities in a post can be regarded as a node classification task.

The TG-HAGCN model contains two GCN modules, which are used to process “user–post–word” graph and “user–date–word” graph respectively. Then, for the post embeddings and date embeddings obtained by the two GCN modules, this model uses a gating mechanism to fuse them. At last, the fused embeddings are fed to a Softmax classifier for predicting HATs.

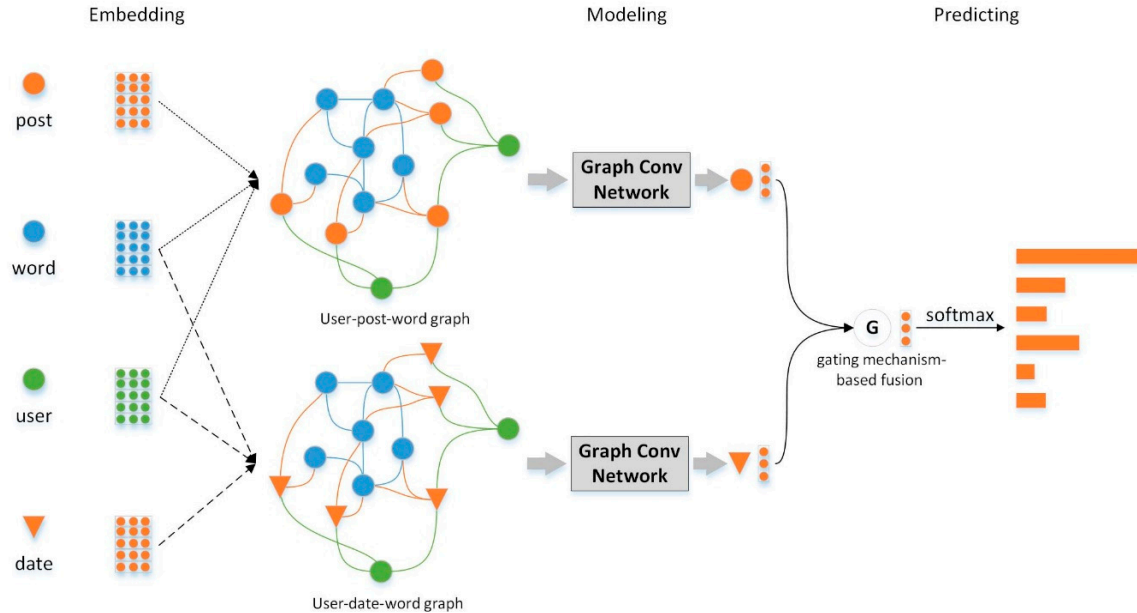


Figure 2. Time Gated Human Activity Graph Convolutional Network (TG-HAGCN) model.

The model defines $X \in R^{n \times m}$ as a feature matrix containing n nodes with their feature vectors and m as the dimension of the feature vectors. The initial features of date nodes are their corresponding date vectors. Specifically, each date vector is encoded by four parts: 4-digit season, 12-digit month, 1-digit working day, 1-digit weekend day, and 7-digit week, where each part is represented by a one-hot vector. The initial feature vectors of words and users for GCN input is random initialized. The initial features of post nodes are the mean value of all the words they contain.

The GCN modules capture information from neighbor nodes, which are two-layer GCNs and can be defined as follows:

$$L^{(j+1)} = \rho(\tilde{A}L^{(j)}W_j), \quad (4)$$

where j is the layer number and $L^{(0)} = X$; $\tilde{A} = D^{-\frac{1}{2}}AD^{\frac{1}{2}}$ is the symmetric normalized adjacency matrix; A is the adjacency matrix of graph G whose diagonal elements are 1; D is the degree matrix of G , where $D_{ii} = \sum_j A_{ij}$; $W_j \in R^{m \times k}$ is a weight matrix; and ρ is an activation function.

By taking “user–post–word” graph G_{post} and “user–date–word” graph G_{date} as the input of two-layer GCN, the embeddings of post nodes and date nodes, L_{post} and L_{date} , can be obtained, respectively.

$$\begin{aligned} L_{post} &= f_{post_mask}(\tilde{A}_{G_{post}} ReLU(\tilde{A}_{G_{post}} X_{G_{post}} W_0^{G_{post}}) W_1^{G_{post}}), \\ L_{date} &= f_{date_mask}(\tilde{A}_{G_{date}} ReLU(\tilde{A}_{G_{date}} X_{G_{date}} W_0^{G_{date}}) W_1^{G_{date}}), \end{aligned} \quad (5)$$

where f denotes the filter function, f_{post_mask} is used to get the post embeddings from all the output embeddings and f_{date_mask} is used to get the date embeddings from the output embeddings.

Then, A gating mechanism-based fusion $\sigma(L_{date}) \odot L_{post}$ is proposed to fuse post embeddings L_{post} and date embeddings L_{date} , where σ is an approximated gating function such as sigmoid and \odot is element-wise product. Moreover, softmax classifier Z can be defined as follows:

$$Z = softmax(L_{post} + \sigma(L_{date}) \odot L_{post} + L_{date}), \quad (6)$$

where $\text{softmax}(x_i) = \frac{1}{Z} \exp(x_i)$ and $\hat{z} = \sum_i \exp(x_i)$.

Finally, the loss function is defined as the cross-entropy overall labeled posts:

$$L = - \sum_{d \in y_D} \sum_{f=1}^F Y_{df} \ln Z_{df}, \quad (7)$$

where y_D is the set of posts that have labels, F is the dimension of the output features that have the same size as the number of types, and Y is the activity type indicator matrix. The weight matrixes W_0 and W_1 are trained using the gradient descent method.

4. Experiment & Results

4.1. Dataset

To examine the proposed model performance, we built a new dataset extracted from the open Yelp dataset (<https://www.yelp.com/dataset>) that includes interactions on post content and users. We selected 300 posters (users) from the downloaded dataset and removed two users whose attributed data are missing. Then, the 298 user items and their posts were collected. The collected data consisted of 23,701 posts published by the 298 users. Finally, each post was labeled into one activity type over 14 types, which are “Food”, “Beauty&spa”, “Entertainment”, “Travel”, “Shopping”, “Service”, “Sports”, “Health”, “Car”, “Nightlife”, “Pets”, “Education”, “Religious” and “Mass media”, and the dataset was built.

As illustrated in Table 1, each item composes of a post text, date, an activity type, and the id of the user who published the post. The average word count of one post was 84.51.

Table 1. Example of an item in the dataset, which consists of the post text, user id, and activity type.

| Attribute | Value | Posts (Description) |
|---------------|--|--------------------------------------|
| post | The pizza was ok. Not the best I’ve had. | The post that user published on YELP |
| User id | msQe1u7Z_XuqiGoqhB0J5g | Unique id for every user |
| Activity type | Food | The activity type of the post |
| Date | 2011-02-25 | Date when the post was published |

Social media activity is closely connected with human daily behavior, which means that the number of each type of activity is not balanced. As shown in Table 2, food and shopping are the two most popular activities. The unbalanced distribution between them challenges the accurate identification of HATs and requires a powerful identity model.

Table 2. Number of posts in each HAT.

| Activity Type | Train | Test | Total |
|----------------|--------|------|--------|
| Food | 11,577 | 4962 | 16,539 |
| Beauty and spa | 146 | 62 | 208 |
| Entertainment | 661 | 283 | 944 |
| Travel | 497 | 213 | 710 |
| Shopping | 1602 | 687 | 2289 |
| Service | 566 | 242 | 808 |
| Sports | 501 | 214 | 715 |
| Health | 83 | 36 | 119 |
| Car | 221 | 95 | 316 |
| Nightlife | 616 | 264 | 880 |
| Pets | 47 | 20 | 67 |
| Education | 24 | 10 | 34 |
| Religious | 26 | 11 | 37 |
| Mass media | 25 | 10 | 35 |

4.2. Experiment Settings

We used a random sampling method to extract 70% of the items from the dataset built in the last section for training, and the remaining 30% of the items for the test, which was 16,592 records for the training and 7109 records for the test.

For the proposed model HAGCN, the embedding size m of each node in the first convolution layer was set to 200, and the sliding window size in Section 3.1 was set to 20. The model learning rate and the dropout rate were set to 0.1 and 0.5, respectively. The proposed model was trained with 300 epochs by using Adam network optimizer [29]. Several classic models, CNN and activity LSTM with dictionary embedding (ALSTM-DE) model [6], were selected as baselines in the experiment. In CNN and ALSTM-DE, the word embedding vector of each word is trained using the Word2Vec model. Training was run for up to 300 epochs with early stopping if the validation loss did not improve within the last 10 epochs.

The widely used metrics, including macro/micro-average precision, recall, and F1 score, were used as evaluation metrics to examine the results of the proposed methods and the baseline models. The metrics were calculated as follows:

$$\begin{aligned}
 \text{Mic_Precision} &= \frac{TP}{TP+FP}, \\
 \text{Mac_Precision} &= \frac{1}{n} \sum_1^n P_i, \\
 \text{Mic_Recall} &= \frac{TP}{TP+FN}, \\
 \text{Mac_Recall} &= \frac{1}{n} \sum_1^n R_i, \\
 \text{Mic_F1} &= 2 \cdot \frac{\text{Mic_Precision} \cdot \text{Mic_Recall}}{\text{Mic_Precision} + \text{Mic_Recall}}, \\
 \text{Mac_F1} &= 2 \cdot \frac{\text{Mac_Precision} \cdot \text{Mac_Recall}}{\text{Mac_Precision} + \text{Mac_Recall}},
 \end{aligned} \tag{8}$$

where TP , FP , and FN represent the true positive rate, the false positive rate, and the false-negative rate of all samples, respectively. n represents the number of activities types. P_i and R_i are the corresponding prediction probability and the recall rate for each activity type, respectively. The calculations of P_i and R_i are similar to those of Mic_Precision and Mic_Recall, but for samples of each activity type.

4.3. Results

The built graph structure has 41,917 nodes, including 17,918 word nodes, 23,701 post nodes, and 298 user nodes. These nodes are linked by 12,436,864 word–word edges, 1,666,595 word–post edges, and 23,701 user–post edges. We conducted experiments on the built dataset using baseline models and the proposed model. The initial feature vectors of unknown words are random initialized during evaluation, and the evaluation results of these models on the test dataset are shown in Table 3.

Table 3. Comparison with baseline models (where “Acc” means accuracy, “F1” means F1-score, “Mac” and “Mic” are the prefixes of “Acc” and “F1” to indicate macro/micro-average accuracy and F1-score). For each row, the best Acc and F1 are indicated in bold.

| HAT | CNN | | ALSTM-DE | | TG-HAGCN | |
|---------------|--------------|--------------|--------------|-------|--------------|--------------|
| | Acc | F1 | Acc | F1 | Acc | F1 |
| Food | 0.806 | 0.832 | 0.963 | 0.966 | 0.963 | 0.970 |
| Beauty | 0.167 | 0.192 | 0.719 | 0.689 | 0.914 | 0.883 |
| Entertainment | 0.585 | 0.441 | 0.756 | 0.691 | 0.831 | 0.788 |
| Travel | 0.622 | 0.578 | 0.822 | 0.833 | 0.849 | 0.859 |
| Shopping | 0.742 | 0.654 | 0.786 | 0.829 | 0.876 | 0.874 |
| Services | 0.782 | 0.760 | 0.530 | 0.557 | 0.708 | 0.709 |
| Sports | 0.649 | 0.577 | 0.740 | 0.755 | 0.841 | 0.866 |
| Health | 0.083 | 0.124 | 0.765 | 0.491 | 0.963 | 0.825 |
| Car | 0.640 | 0.689 | 0.800 | 0.706 | 0.832 | 0.804 |

Table 3. Cont.

| HAT | CNN | | ALSTM-DE | | TG-HAGCN | |
|------------|--------------|--------------|----------|-------|--------------|--------------|
| | Acc | F1 | Acc | F1 | Acc | F1 |
| Nightlife | 0.991 | 0.583 | 0.766 | 0.693 | 0.805 | 0.768 |
| Pets | 0.048 | 0.080 | 0.600 | 0.514 | 0.938 | 0.833 |
| Education | 1.000 | 0.462 | 0.500 | 0.167 | 1.000 | 0.333 |
| Religious | 1.000 | 0.167 | 0.714 | 0.556 | 0.909 | 0.909 |
| Mass media | 1.000 | 0.462 | 0.500 | 0.167 | 1.000 | 0.462 |
| Mic | 0.757 | 0.757 | 0.898 | 0.898 | 0.926 | 0.926 |
| Mac | 0.651 | 0.526 | 0.712 | 0.637 | 0.888 | 0.777 |

As shown in Table 3, TG-HAGCN model achieves the best performance on both Mic accuracy and Mac accuracy. It should be noted that CNN models make very lower Mac accuracy, which may be raised by the unbalance of training sample.

To observe the misclassify item, we plotted the confusion matrixes of different models in Figure 3. The figure shows that our model performed the best for dealing with category confusion in the unbalanced dataset. We argue that TG-HAGCN model may perform better in other tasks.

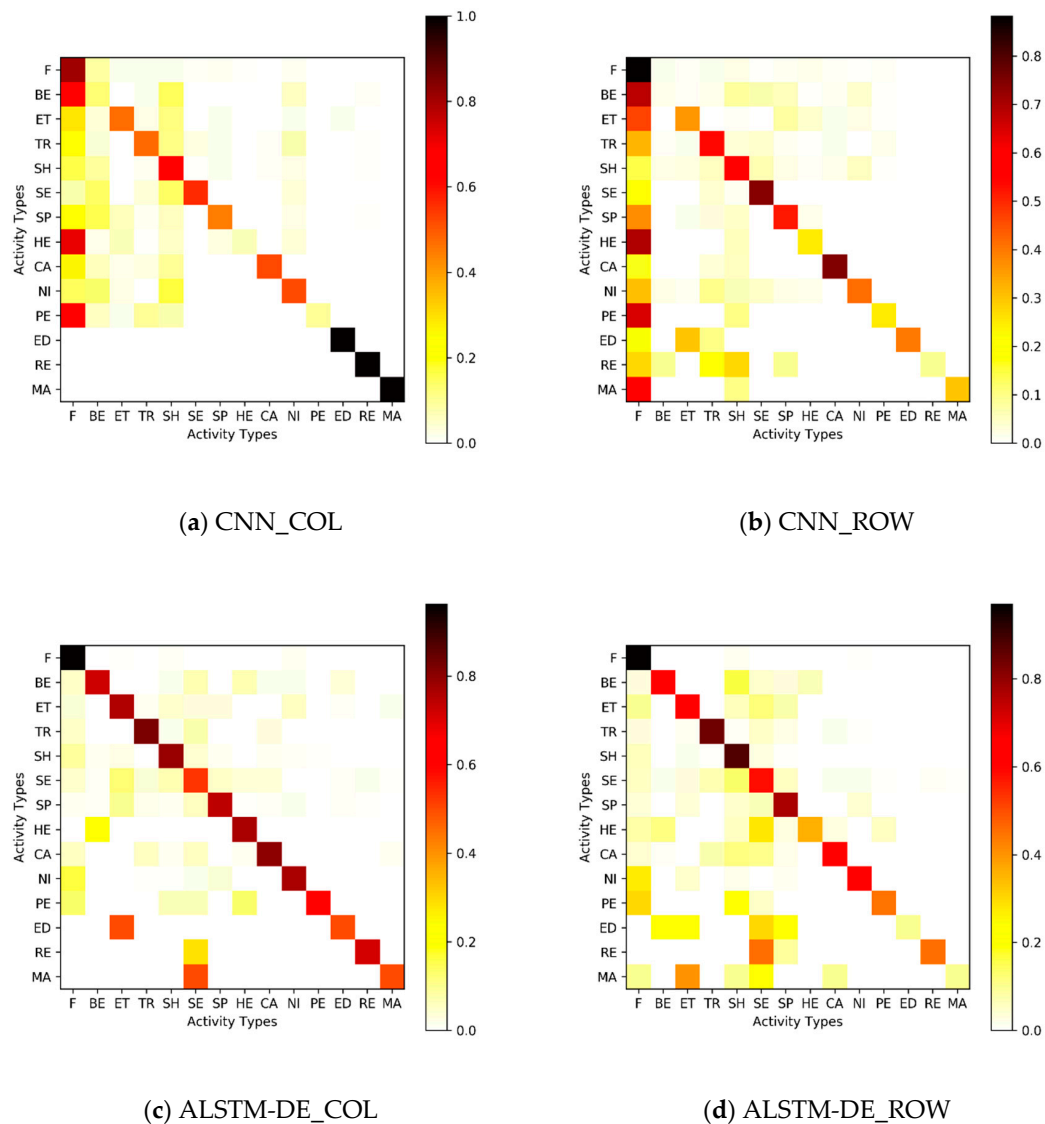


Figure 3. Cont.

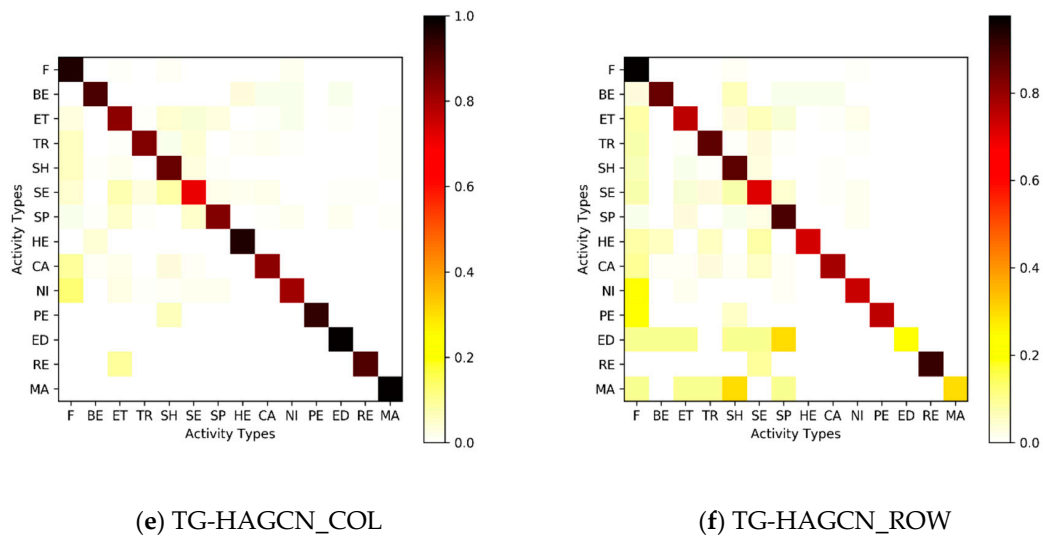


Figure 3. Heat maps of activities for the models. Since it is an unbalanced dataset, we have performed column-wise normalization (COL) and row-wise normalization (ROW) on the original heat maps. The values on the diagonal of “COL” correspond to precision score, and the values on the diagonal of “ROW” correspond to recall score.

Comparatively, the proposed model captures the network structure (interaction) between nodes and edges by exchanging features with their neighbors in training process. In this way, the samples in each category are not trained independently any more, hence, achieving best Mac accuracy in those models. In addition, the TG-HAGCN model has enough features from interactions to make accurate activity classification in a small dataset. Besides, the model performs well in the recognition of activity types in the unbalanced dataset. The results suggest that interactions and date information can supplement additional information, and this information together with methods to fuse them should be considered on improving the model performance in related works.

4.4. Training Time and Convergence

We also calculated the time consumption for different models. Detailed results are presented in Table 4. Our model has the similar time consuming compared with CNN model and obtain the best accuracy compared with conventional machine learning methods.

Table 4. Comparison of runtime (seconds).

| Model | Run Time |
|----------|----------|
| CNN | 5622 |
| ALSTM-DE | 15,233 |
| TG-HAGCN | 5795 |

We illustrated the accuracy and loss curves in Figure 4 to examine the performance changes of different models during the training process. As shown in Figure 4a, the model accuracy gradually increases as the training epoch increases. It demonstrates the recognition ability of our model to improve continuously. Figure 4b shows that the model loss gradually decreases as the training epoch increases. It demonstrates that the errors in our model decrease gradually. The most important is that our two models can converge before 150 training epochs. Although other models have less loss and higher accuracy before 30 epochs training, their values always have small fluctuations. This result indicates that the training process of the proposed methods are stable.

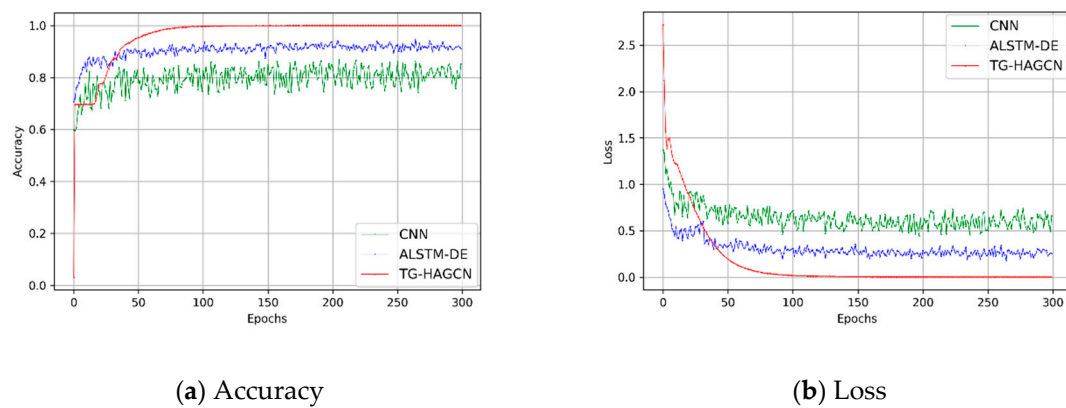


Figure 4. Curves of (a) accuracy (b) loss of different models.

5. Discussion

5.1. Effect of Word Vector Size and Sliding Window Size

We performed two sets of experiments to evaluate the influence of different embedding dimensions and sliding window sizes.

The sliding window size defines the number of words associated with each word. If the sliding window size is too small, this model cannot capture enough information; if the sliding window size is too large, this model may introduce excessive noise. In the first set of experiments, the word vector dimension was set to 200, and the sliding window size ranged from 5 to 20. As shown in Table 5, the change in sliding window size slightly influences on the final experimental results, and it is only slightly increased. This finding suggests that our model is insensitive to the sliding window size.

Table 5. Precision and F1 score in different sliding window sizes for 200-dimension word embedding.

| Metric | Sliding Window Size = 5 | Sliding Window Size = 10 | Sliding Window Size = 15 | Sliding Window Size = 20 |
|---------------|-------------------------|--------------------------|--------------------------|--------------------------|
| Mic-Precision | 0.9224 | 0.9232 | 0.9232 | 0.9259 |
| Mac-Precision | 0.8661 | 0.8631 | 0.8641 | 0.8877 |
| Mic-F1 | 0.9224 | 0.9232 | 0.9232 | 0.9259 |
| Mac-F1 | 0.7539 | 0.7662 | 0.7541 | 0.7774 |

The dimension of the word vector will directly determine the number of parameters in the hidden layer. When the word vector dimension is small, the trained model may fit poorly; when the word vector dimension is large, the model is likely over-fitted. In this set of experiments, the sliding window size is 20, and the word vector dimension ranges from 100 to 300. As shown in Table 6, the model performs best at 200 dimensions and has slightly improves compared with other dimensions.

Table 6. Precision and F1 score in different dimension word embedding for 20 sliding window size.

| Metric | 100-Dimension | 200-Dimension | 300-Dimension |
|---------------|---------------|---------------|---------------|
| Mic-Precision | 0.9219 | 0.9259 | 0.9229 |
| Mac-Precision | 0.8007 | 0.8877 | 0.8915 |
| Mic-F1 | 0.9219 | 0.9259 | 0.9229 |
| Mac-F1 | 0.7333 | 0.7774 | 0.7522 |

In a word, the results are insensitive to the word vector dimension and sliding window size when the two hyperparameters are set to the values commonly used in existing studies, suggesting that the proposed model is robust.

5.2. Effect of Different Edge Weights

In Section 3, we designed three strategies, PMI, TFIDF, and UC, to quantitatively model the interaction between three types of nodes. An alternative simple solution is to set the edge to constant 1. To examine the three strategies, we built different edges among the three types of nodes by using PMI, TFIDF, and UC in our model, as illustrated in Section 3. To verify whether or not the edges built by PMI, TFIDF, and UC are better than using fixed constant only, we changed every edge as constant 1 in the whole graph and compared it with our models. The experimental results are listed in Table 7.

Table 7. Precision and F1 score in our models with different weights.

| Model | Precision | F1 |
|---------------------------------|---------------|---------------|
| Mic (P_W=1,W_W=1,P_U=1) | 0.9058 | 0.9058 |
| Mic (P_W=1, W_W=PMI,P_U=UC) | 0.9110 | 0.9110 |
| Mic (P_W= TFIDF,W_W=1,P_U=UC) | 0.9216 | 0.9216 |
| Mic (P_W= TFIDF,W_W=PMI,P_U=1) | 0.9204 | 0.9204 |
| Mic (P_W= TFIDF,W_W=PMI,P_U=UC) | 0.9259 | 0.9259 |
| Mac (P_W=1,W_W=1,P_U=1) | 0.9058 | 0.5889 |
| Mac (P_W=1, W_W=PMI,P_U=UC) | 0.7430 | 0.5630 |
| Mac (P_W= TFIDF,W_W=1,P_U=UC) | 0.8074 | 0.7294 |
| Mac (P_W= TFIDF,W_W=PMI,P_U=1) | 0.8662 | 0.7621 |
| Mac (P_W= TFIDF,W_W=PMI,P_U=UC) | 0.8877 | 0.7774 |

As shown in Table 7, the edges built by PMI, TFIDF, and UC are better than constant 1 in the final result (F1 score). It shows that using different models to build different types of edges can capture favorable features between nodes. When all the edge weights are set to 1, the model has the best precision according to the Mac value, and the F1 score is worse. The reason is that in this setting, the differences between the nodes are weakened, resulting in high classification accuracy in categories with a large number of samples and poor results in other categories. When P_W weight is 1, the result is worse than others. This finding illustrates that using TFIDF to calculate weight is better than constant 1, and the edges between posts and words have the greatest influence on the final result because TFIDF can indicate the relation between posts and words accurately.

5.3. Ablation Study

To further examine the benefit brought by each component of TG-HAGCN, an ablation study is performed in this study. The experiment results are listed in Table 8, as shown in Table 8, the “TG-HAGCN” model is better than the “TG-HAGCN /Date & /User” model, and the “TG-HAGCN /Date & /User” model is better than the “TG-HAGCN /Date” model and the “TG-HAGCN /User” model. This shows that both time and users are important for the TG-HAGCN model. When only time or only users are considered in the model, the model does not significantly improve the performance of predication. The reason may be that the HATs of different users are diverse, and the distribution of HATs in time is varied.

Table 8. Precision and F1 score of ablation study. “TG-HAGCN /Date” means that the date nodes are removed in the TG-HAGCN model and the input graph is only “user-post-word” graph. “TG-HAGCN /User” means to remove user nodes from TG-HAGCN model, that is, the input graphs are “post-word” graph and “date-word” graph. “TG-HAGCN/Date & /User” means that the input graph is only “post-word” graph.

| Model | Precision | F1 |
|-----------------------------|---------------|---------------|
| Mic (TG-HAGCN) | 0.9259 | 0.9259 |
| Mic (TG-HAGCN/Date) | 0.9216 | 0.9216 |
| Mic (TG-HAGCN/User) | 0.9226 | 0.9226 |
| Mic (TG-HAGCN/Date & /User) | 0.9218 | 0.9218 |
| Mac (TG-HAGCN) | 0.8877 | 0.7774 |
| Mac (TG-HAGCN/Date) | 0.8443 | 0.7593 |
| Mac (TG-HAGCN/User) | 0.8558 | 0.7356 |
| Mac (TG-HAGCN/Date & /User) | 0.8775 | 0.7739 |

5.4. The Relationship between Activity Types and Related Terms

While the proposed method trains the samples, various vectors, including post, word, and user vectors, are obtained, corresponding to the posts, words in posts, and users.

To examine the relationships between activity types and related text words, we considered the word vectors as a sample and employed semantic similarity evaluation to conduct experiments. We first manually select key words from the activity type name as the center word for each activity, respectively. The dataset contains 14 activity types, which are “Food”, “Beauty&spa”, “Entertainment”, “Travel”, “Shopping”, “Service”, “Sports”, “Health”, “Car”, “Nightlife”, “Pets”, “Education”, “Religious”, and “Mass media”. The selected 14 center words are “Food”, “Beauty”, “Entertainment”, “Travel”, “Shopping”, “Service”, “Sports”, “Health”, “Car”, “Nightlife”, “Pets”, “Education”, “Religious”, and “Mass media”. Then, we calculated the cosine distances between vectors of all the words to the vector of 14 center words and listed the top 7 closest words for each center word in Table 9.

Table 9. Words that have the closest semantics with the selected words (the semantic distance is increased from left to right).

| Center Word | Similar Words of Center Word | | | | | | |
|---------------|------------------------------|------------|----------|-----------|---------------|----------|-------------|
| Food | drink | foods | supplies | meat | fresh | eating | fish |
| Beauty | psyche | beautiful | beast | truth | her | pleasure | happiness |
| Entertainment | theater | movie | stage | museum | acrobatics | film | dancer |
| travel | travels | trip | journey | explore | go | visit | visitors |
| shopping | mall | restaurant | retail | shops | entertainment | stores | attractions |
| Service | clean | customer | carpet | realtor | professional | fix | technician |
| Sports | sport | teams | football | sporting | racing | clubs | basketball |
| Health | medical | care | mental | health | treatment | disease | benefits |
| Car | vehicle | oil | auto | repair | kia | drive | mechanic |
| Nightlife | bar | drink | night | club | dj | beers | pub |
| Pets | dogs | cat | cats | mouse | horses | mice | boss |
| Education | school | teacher | class | student | college | learning | instructor |
| Religious | political | religion | social | christian | spiritual | secular | moral |
| Mass media | radio | news | music | listen | channel | listener | tv |

As shown in Table 9, the top 7 closest words in each row usually appear in the same activity scene, as words in the same activity type can be correctly mapped into the same vector space. It suggests that the training process have successfully captured the semantic relationship between activity types and text words and the identified activity type is meaningful.

6. Conclusions

To fill the gap of overlooking the interaction that contributes to identifying HATs, this study proposes a novel Time Gated Human Activity Graph Convolutional Network (TG-HAGCN). The model identifies human activities from social media trajectory by synthesizing two GCN modules. Experimental results show that the proposed model significantly outperforms classical models, especial on Mac accuracy in unbalanced datasets. Furthermore, detailed analyses suggest that all three strategies, PMI, TFIDF, and UC, contribute to the improvement of the model. The results of the ablation study show that both users and time are important for the identification of human activities. The results of this study indicate that interactions among social media promotes ability of machine learning on social media data mining and intelligent applications. Benefiting from the fact that GCN can link different types of objects, it is easy to synthesize information from heterogeneous objects and is expected to serve a variety of domain-specific applications. The fruits of this study not only highlight the importance of interactions hidden in social media, promote ability of machine learning on social media data mining and intelligence, but also offer a reference solution for how to fuse multi-type heterogeneous data in social media.

In the future, other interactions, such as friend relationships between users, will be incorporated. Spatial information of posts can also be used as additional supplementary information. Finally, the performance of word, post, and user vectors need to be examined in downstream applications.

Author Contributions: Data curation, R.L. and R.C.; methodology, H.Y., R.C., and J.G.; resources, X.K.; Software, R.C., R.L.; writing—review and editing, S.L., H.Y., X.K., L.D., and J.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (grant number 41801378, 61972365, 61672474, 42071382), Open Fund of Key Laboratory of Urban Land Resources Monitoring and Simulation, Ministry of Natural Resources (grant number: KF-2019-04-033), and in part by the Natural Science Foundation of Hubei Province, China (grant number: 2020CFB752).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Tu, W.; Cao, J.; Yue, Y.; Shaw, S.L.; Zhou, M.; Wang, Z.; Chang, X.; Xu, Y.; Li, Q. Coupling mobile phone and social media data: A new approach to understanding urban functions and diurnal patterns. *Int. J. Geogr. Inf. Sci.* **2017**, *31*, 2331–2358. [\[CrossRef\]](#)
2. Batty, M.; Axhausen, K.W.; Giannotti, F.; Pozdnoukhov, A.; Bazzani, A.; Wachowicz, M.; Ouzounis, G.; Portugali, Y. Smart cities of the future. *Eur. Phys. J. Spec. Top.* **2012**, *214*, 481–518. [\[CrossRef\]](#)
3. De Pessemier, T.; Doms, S.; Martens, L. Context-aware recommendations through context and activity recognition in a mobile environment. *Multimed. Tools Appl.* **2014**, *72*, 2925–2948. [\[CrossRef\]](#)
4. De Pessemier, T.; Martens, L. Heart rate monitoring, activity recognition, and recommendation for e-coaching. *Multimed. Tools Appl.* **2018**, *77*, 23317–23334. [\[CrossRef\]](#)
5. Hasan, S.; Ukkusuri, S.V. Urban activity pattern classification using topic models from online geo-location data. *Transp. Res. Part C Emerg. Technol.* **2014**, *44*, 363–381. [\[CrossRef\]](#)
6. Gong, J.; Li, R.; Yao, H.; Kang, X.; Li, S. Recognizing human daily activity using social media sensors and deep learning. *Int. J. Environ. Res. Public Health* **2019**, *16*, 3955. [\[CrossRef\]](#) [\[PubMed\]](#)
7. Saha, J.; Chowdhury, C.; Chowdhury, I.R.; Biswas, S.; Aslam, N. An ensemble of condition based classifiers for device independent detailed human activity recognition using smartphones. *Information* **2018**, *9*, 94. [\[CrossRef\]](#)
8. Salomón, S.; Tîrnăuică, C. Human Activity Recognition through Weighted Finite Automata. *Proceedings* **2018**, *2*, 1263. [\[CrossRef\]](#)
9. Wang, J.; Chen, Y.; Hao, S.; Peng, X.; Hu, L. Deep learning for sensor-based activity recognition: A survey. *Pattern Recognit. Lett.* **2019**, *119*, 3–11. [\[CrossRef\]](#)
10. Asghari, P.; Soleimani, E.; Nazerfard, E. Online human activity recognition employing hierarchical hidden Markov models. *J. Ambient Intell. Humaniz. Comput.* **2020**, *11*, 1141–1152. [\[CrossRef\]](#)

11. Demrozi, F.; Pravadelli, G.; Bihorac, A.; Rashidi, P. Human Activity Recognition using Inertial, Physiological and Environmental Sensors: A Comprehensive Survey. *IEEE Access* **2020**, *8*, 210816–210836. [[CrossRef](#)]
12. Cai, G.; Hio, C.; Bermingham, L.; Lee, K.; Lee, I. Sequential pattern mining of geo-tagged photos with an arbitrary regions-of-interest detection method. *Expert Syst. Appl.* **2014**, *41*, 3514–3526. [[CrossRef](#)]
13. Belcastro, L.; Marozzo, F.; Talia, D.; Trunfio, P. G-RoI: Automatic Region-of-Interest Detection Driven by Geotagged Social Media Data. *ACM Trans. Knowl. Discov. Data* **2017**, *3*, 1–22. [[CrossRef](#)]
14. Liu, H.; Luo, B.; Lee, D. Location type classification using tweet content. In Proceedings of the 11th International Conference on Machine Learning and Applications, Boca Raton, FL, USA, 12–15 December 2012; Volume 1, pp. 232–237.
15. Dearman, D.; Sohn, T.; Truong, K.N. Opportunities Exist: Continuous discovery of places to perform activities. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Vancouver, BC, Canada, 7–13 May 2011; pp. 2429–2438.
16. Benetka, J.R.; Balog, K.; Nørvag, K. Anticipating information needs based on check-in activity. In Proceedings of the 10th ACM International Conference on Web Search and Data Mining, Cambridge UK, February 2017; pp. 41–50.
17. Yang, D.; Zhang, D.; Zheng, V.W.; Yu, Z. Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs. *IEEE Trans. Syst. Man Cybern. Syst.* **2015**, *45*, 129–142. [[CrossRef](#)]
18. Krumm, J.; Rouhana, D.; Chang, M.W. Placer++: Semantic place labels beyond the visit. In Proceedings of the IEEE International Conference on Pervasive Computing and Communications, St. Louis, MO, USA, 23–27 March 2015; pp. 11–19.
19. Lian, D.; Xie, X. Collaborative activity recognition via check-in history. In Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location-Based Social Networks, Chicago, IL, USA, 1 November 2011; pp. 45–48.
20. Beber, M.A.; Ferrero, C.A.; Fileto, R.; Bogorny, V. Towards activity recognition in moving object trajectories from twitter data. In Proceedings of the Brazilian Symposium on GeoInformatics, São Paulo, Brazil, 27–30 November 2016; pp. 68–79.
21. Weerkamp, W.; Rijke, D. Activity Prediction: A Twitter-Based Exploration. In Proceedings of the SIGIR 2012 Workshop on Time-Aware Information Access, Portland, OR, USA, 4 June 2012.
22. Song, Y.; Lu, Z.; Leung, C.W.K.; Yang, Q. Collaborative boosting for activity classification in microblogs. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, IL, USA, 11–14 August 2013; pp. 482–490.
23. Zhu, Z.; Blanke, U.; Tröster, G. Recognizing composite daily activities from crowd-labelled social media data. *Pervasive Mob. Comput.* **2016**, *26*, 103–120. [[CrossRef](#)]
24. Mazumder, S.; Patel, D.; Mehta, S. ActMiner: Discovering location-specific activities from community-authored reviews. In Proceedings of the 16th International Conference, DaWaK 2014, Munich, Germany, 2–4 September 2014; Springer: Cham, Germany, 2014; pp. 332–344.
25. Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; Yu, P.S. A Comprehensive Survey on Graph Neural Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, 1–21. [[CrossRef](#)] [[PubMed](#)]
26. Tobler, W. On the First Law of Geography. *Ann. Assoc. Am. Geogr.* **2004**, *94*, 304–310. [[CrossRef](#)]
27. Khan, F.H.; Qamar, U.; Bashir, S. SentiMI: Introducing point-wise mutual information with SentiWordNet to improve sentiment polarity detection. *Appl. Soft Comput. J.* **2016**, *39*, 140–153. [[CrossRef](#)]
28. Shaw, S.-L.; Sui, D. *Human Dynamics in Smart Cities*; Springer: Berlin/Heidelberg, Germany, 2018; ISBN 9783319732466.
29. Bello, I.; Zoph, B.; Vasudevan, V.; Le, Q.V. Neural optimizer search with Reinforcement learning. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 712–721.

Publisher’s Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).