Traffic Information Mining From Social Media Based on the MC-LSTM-Conv Model

Yiwen Wang, Zhi He, *Senior Member, IEEE*, and Jie Hu

***Abstract*— Social media (e.g., Sina Weibo) have the advantage of reflecting traffic information, including the reasons for jams, illegal behaviors, and emergency recourses on roads. However, there remains a challenging issue regarding how to sufficiently mine traffic information. In this paper, we propose a deep learning-based method that uses social media data for traffic jam management. The core ideas of the proposed method are twofold. First, a multichannel network with a Long Short-Term Memory layer (LSTM-layer) and a Convolution layer (Conv-layer) (termed as MC-LSTM-Conv) is proposed. This model consists of two information channels for extracting abstract features from input text. Each channel includes two Conv-layers, and an LSTM-layer is added to one of the four Conv-layers. The MC-LSTM-Conv model is used to extract check-in microblogs reflecting traffic jams from mass Sina Weibo data. Second, a series of matching rules are constructed based on the keywords that are related to traffic-jam scenes. These rules further classify the microblogs extracted by the first step into four classes, and each of the classes reflects a specific road condition (i.e., traffic accidents or large-scale activities, road construction, traffic lights, and the low efficiency of government agencies). Experiments on Sina Weibo data demonstrate that the proposed multichannel network has superior performance in extracting microblogs about traffic jams. The keyword fuzzy matching method can fetch detailed information about traffic jams efficiently.**

***Index Terms*— Sina Weibo, convolution neural networks, long short-term memory, traffic jam.**

I. Introduction

WITH the rapid development of the economy and the continuous advancement of urbanization, the number of motor vehicles keeps increasing. Traffic problems have become an "urban disease" that troubles the world's major cities. Guangdong province has 21 prefecture-level municipalities (including two sub-provincial ones). Since the Chinese economic reform, the residents' travel demand has increased with the rapid development of Guangdong's economy. According to data from Seligence Market Research, the number of daily trips per capita in Guangdong province reached 4.48 in 2017, which is much higher than that of European developed countries. Under such heavy traffic pressure, traffic jams are common on roads. These problems not only affect the logistic transportation between cities, but also dramatically hinder the improvement of citizens' living quality, which has become a bottleneck restricting urban development.

Due to the rapid development of the Internet industry in the 21st century, social media have gradually become popular and act as an important means of information exchange among various groups of people. A large amount of data reflecting the users' life dynamics are generated in the process of using social media software. The data also reveal the operational status of the urban system. As a result, much work has been carried out to mine the information from social media data for urban management [1], [2], such as epidemic monitoring, urban waterlogging, and crowd sentiment analysis during disaster events.

Social media data have also been applied in traffic studies. Compared with traditional road monitoring means such as camera monitoring and police patrols, social media data can reflect the reasons for jams, the discontent, and the opinions of the public in the way of short texts or pictures [3], [4]. Social media data provide a convenient way for government departments to discover the travel problems of citizens in time, thus improving the efficiency of service-oriented government. Over the past decades, social media data have demonstrated their capabilities in a series of traffic studies. Agarwal and Toshniwal [5] mined social media tweet text to identify the complaints regarding various road transportation issues including traffic, accidents, and potholes. Huang *et al*. [6] modeled the relationship between the characteristics of business clusters and check-in activities for Los Angeles County, California. Cao *et al*. [7] used social network data instead and analyzed traffic conditions based on users' sentiment in Chinese microblogs. Maghrebi *et al*. [8] showed how social media data could be used to extract travel mode choices that could serve as a supplementary source of information for house travel surveys. Recently, traffic incident detection [9]–[12] and traffic prediction [13]–[16] have also received considerable attention. Moreover, how to improve the accuracy of traffic event detection and extract the comprehensive information about traffic from massive social media data have become important research topics.

Manuscript received January 14, 2019; revised May 27, 2019, September 29, 2019, December 13, 2019, and March 22, 2020; accepted August 13, 2020. Date of publication September 16, 2020; date of current version February 2, 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB0505500 and Grant 2018YFB0505503, in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2019A1515011877, in part by the Fundamental Research Funds for the Central Universities under Grant 19lgzd10, in part by the Guangzhou Science and Technology Planning Project under Grant 202002030240, in part by the National Natural Science Foundation of China under Grant 41501368, and in part by the Southern Marine Science and Engineering Guangdong Laboratory (Zhuhai) under Grant 99147-42080011. The Associate Editor for this article was Z. Duric. *(Corresponding author: Zhi He.)*

The authors are with the Guangdong Provincial Key Laboratory of Urbanization and Geo-simulation, Center of Integrated Geographic Information Analysis, Sun Yat-sen University, Guangzhou 510275, China, and also with the Southern Marine Science and Engineering Guangdong Laboratory (Zhuhai), School of Geography and Planning, Sun Yat-sen University, Guangzhou 510275, China (e-mail: hezh8@mail.sysu.edu.cn).

Digital Object Identifier 10.1109/TITS.2020.3021096

To fully utilize social media data for traffic event detection, researchers have proposed various approaches that can be divided into two categories. The first category is the conventional methods, including the support vector machine (SVM) [17]–[19] and hinge loss Markov random fields (HLMRFs) [20]. In greater detail, D'Andrea *et al*. [17] represented each tweet text as a vector and applied a SVM for classification. Based on HLMRFs, Chen *et al*. [20] combined a topic modeling based language model and a collaborative inference model to observe traffic conditions from tweet text. Although these conventional methods have achieved promising results in their studies, they rely on manual features and domain knowledge.

The latter are the deep learning models, including convolutional neural networks (CNNs) [21], deep belief networks (DBNs) [22], generative adversarial networks (GANs) [7], recurrent neural networks (RNNs), Long Short-Term Memory (LSTM), and the combination of LSTM and CNNs [23], [24]. In recent years, these deep learning neural networks have demonstrated their ability in monitoring traffic conditions. For instance, Zhang *et al*. [22] chose to combine a DBN with LSTM to detect specific traffic accidents from Twitter data. Chen *et al*. [21] used the word embedding method to represent the semantics of social media text and applied CNNs to obtain traffic information. The experiments in [21] proved that the proposed CNN model was superior to the SVM. Cao *et al*. [7] combined a GAN and gated recurrent unit (GRU) to study the sentiment of Sina Weibo users in traffic conditions and used the sentiment index to predict the possibility of traffic jams. CNN, LSTM and other deep learning models can solve the problem end to end, independent of the artificial feature selection method, such as the Document Frequency (DF), Mutual Information (MI), and Information Gain (IG).

In this paper, we propose a deep learning-based method to process the social media data for traffic information including the reasons for jams, illegal behaviors and emergency recourses on roads, etc. First, a multichannel network with a Long Short-Term Memory layer (LSTM-layer) and a Convolution layer (Conv-layer) (termed as MC-LSTM-Conv) is proposed to detect the information about traffic jams from mass social media data on microblogs. This multichannel network contains four Conv-layers in parallel, and an LSTM-layer is placed in front of one of these Conv-layers. Second, a series of rules are established to fetch detailed information about the traffic jams. The keywords are selected from the high-frequency words of microblogs about traffic jams, supplemented by life experience and personal knowledge. Moreover, the main social media data used in this paper are crawled from Sina Weibo, one of the most popular social media software in China. In terms of social relations, similar to Twitter, Sina Weibo is more focused on strangers' social contact, and the published content is relatively open. Anyone can access the stock of information on Sina Weibo anytime and anywhere. On the other hand, the social relationships between Facebook and WeChat users in China emphasize social contact between acquaintance and the privacy of personal information. The proposed method can extract traffic information from millions of published check-in microblogs. The primary contributions of this paper are as follows.

- This paper proposes a deep learning model named MC-LSTM-Conv. The multichannel network structure and the combination of the LSTM-layer and Conv-layer can efficiently extract the semantic features from microblogs. Compared with other deep learning models, the MC-LSTM-Conv model performs better in the binary classification problem (traffic jam versus non-traffic jam).
- This paper also uses a keyword fuzzy matching method to classify the microblogs relevant to traffic jams into four classes, each of which reflects a specific road condition. By using this method, we overcome the insufficient number of labeled samples for the training model and reduce the workload of artificial identification in a particular event type.

The remainder of the paper is organized as follows. In Section II, the proposed method is described in detail. The effectiveness of the method is investigated and illustrated by the experimental results shown in Section III. Finally, Section IV draws the conclusion and shows the future steps to be done.

II. Methodology

By crawling Sina Weibo web pages, we can collect millions of check-in microblogs with keywords such as "jam(du-塞)" and "plug(sai-堵)", which may be related with traffic jams. To acquire the needed information as fully as possible, we design a traffic information mining method for dealing with the mass Sina Weibo data. As shown in Fig. 1, this method mainly consists of two parts, i.e., MC-LSTM-Conv to extract the microblogs about traffic jams from mass data and keyword fuzzy matching method to mine detailed traffic information.

The introduction to MC-LSTM-Conv mainly includes two parts: 1. text representation, and 2. feature extraction and classification. Text representation is the transformation of text into a mathematical vector or matrix, which can be understood by deep learning models. In this paper, we use word embedding to represent every sentence inputted as a sentence matrix. Feature extraction and classification automatically extracts features from the previously obtained sentence matrix, and uses these features to better identity microblogs concerning traffic jams from mass data. This part details the feature extraction principle and the characteristics of LSTM and the CNN, and proposes a multichannel structure to accommodate the Conv-layers and LSTM-layers.

The introduction to the keyword fuzzy matching method mainly explains how to use keywords and keyword combinations to establish matching rules for each detailed congestion category and presents the standards established by regular expression in English. The reason for choosing the keyword fuzzy matching method instead of deep learning models for multilabel classification is as follows: 1. the number of labeled samples is insufficient and thus may lead to over-fitting, and 2. the samples are unevenly distributed among the classes, which may result in model performing poorly in certain

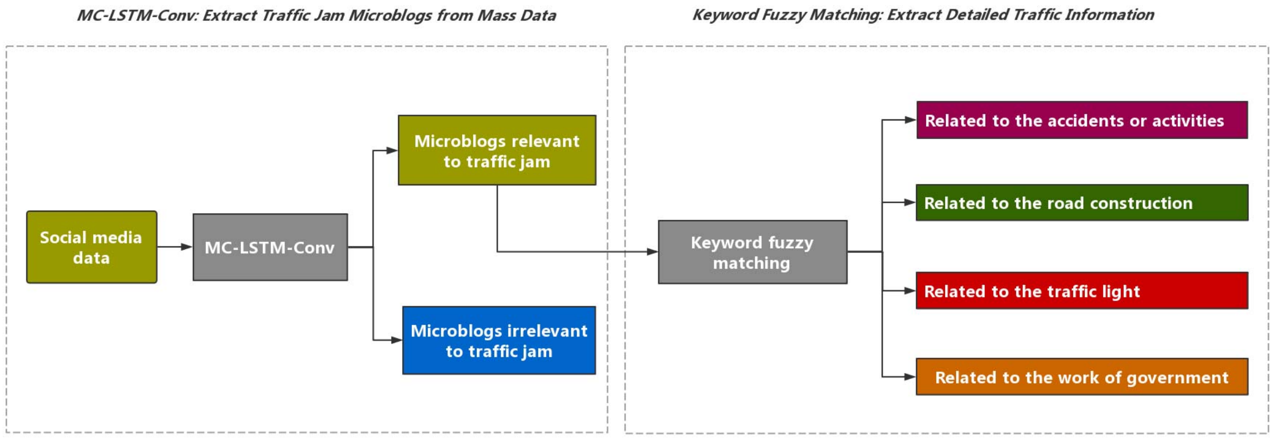


Fig. 1. Schematic illustration of the proposed traffic information mining method.

classes without enough training samples. By summarizing the keywords reflecting the traffic information, as well as the collocation of keywords, the keyword fuzzy matching method overcomes the above two problems easily and extracts the needed information effectively.

A. MC-LSTM-Conv to Extract Microblogs about Traffic Jams From Mass Data

1) Text Representation: Note that the check-in microblogs we crawled are textual data. The first step of the MC-LSTM-Conv model is text representation, which can convert the text into a vector or matrix. Compared with English and other languages whose basic unit is words, Chinese text representation should first solve the word segmentation problem. Therefore, we choose Python's third-party library *jieba* to segment sentences. To improve the word segmentation accuracy, we establish a user dictionary based on the information, including Points of Interest (POIs), roads, and place names in the major cities of Guangdong province. For example, when *jieba* segments the sentence "Guangzhou haizhu district ruibao road da-gan-wei haizhu creative park section (广州海珠区瑞宝路大干围海珠创意园路段)", without loading the user dictionary, the word segmentation result is "Guangzhou, haizhu district, ruibao road, dagan, wai, haizhu, creative, park, road section (广州, 海珠区, 瑞宝路, 大干, 围, 海珠, 创意, 园, 路段)". After loading the user dictionary, the result is "Guangzhou, haizhu district, ruibao road, da-gen-wai, haizhu, creative park, road section (广州, 海珠区, 瑞宝路, 大干围, 海珠, 创意, 园, 路段)". Since "da-gan-wei (大干围)" is a complete place name, loading the user dictionary helps to reduce the number of incorrect segmentations in proper nouns.

Traditional text representation includes one-hot, the Bag of Words (BoW), etc [25]. The one-hot method supposes a dictionary of n words, and each word can be represented as a vector of length n. Only one vector element is 1, and all other elements are 0. The index of the 1 element indicates the location of the corresponding word in the dictionary. The BoW represents every sentence as a vector of n, each vector element corresponds to one word in the dictionary, and the

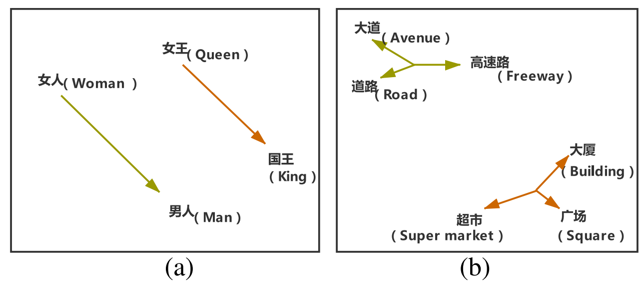


Fig. 2. (a) shows that the direction and distance from the word "Queen" to the word "King" are similar to those from the word "Woman" to the word "Man". (b) shows that words including "avenue", "freeway", "highway", etc. cluster together.

value of the vector element equals the frequency of the corresponding word in the sentence. This kind of method has the characteristics of high dimension and high sparsity, resulting in a weak feature expression ability. In this paper, we use the word embedding to represent the word identity [26]–[29], which can automatically realize the semantic analogy and the word semantic similarity measure. After projecting the word embedding into the two-dimensional coordinate system, the above two relations correspond to (a) and (b) in Fig. 2, respectively. Fig. 2(a) shows that the direction and distance from the word "Queen" to the word "King" are similar to those from the word "Woman" to the word "Man". Fig. 2(b) shows that words including "avenue", "freeway", "highway", etc. cluster together.

Two widely-used neural network language models, namely the Skip-Gram and continuous bag of words (CBOW), can convert each word of a sentence into a word embedding. The CBOW model infers the target word from the original contextual statement. The Skip-Gram model, by contrast, speculates about the original contextual statement from the target word. In this paper, the Skip-Gram model is adopted to obtain the word embedding because this model performs better than the CBOW when the scale of the corpus is not significant. We use two corpora to train the Skip-Gram model. One is randomly collected from the Internet including advertisements, novels, comments, etc. The other is 1 million

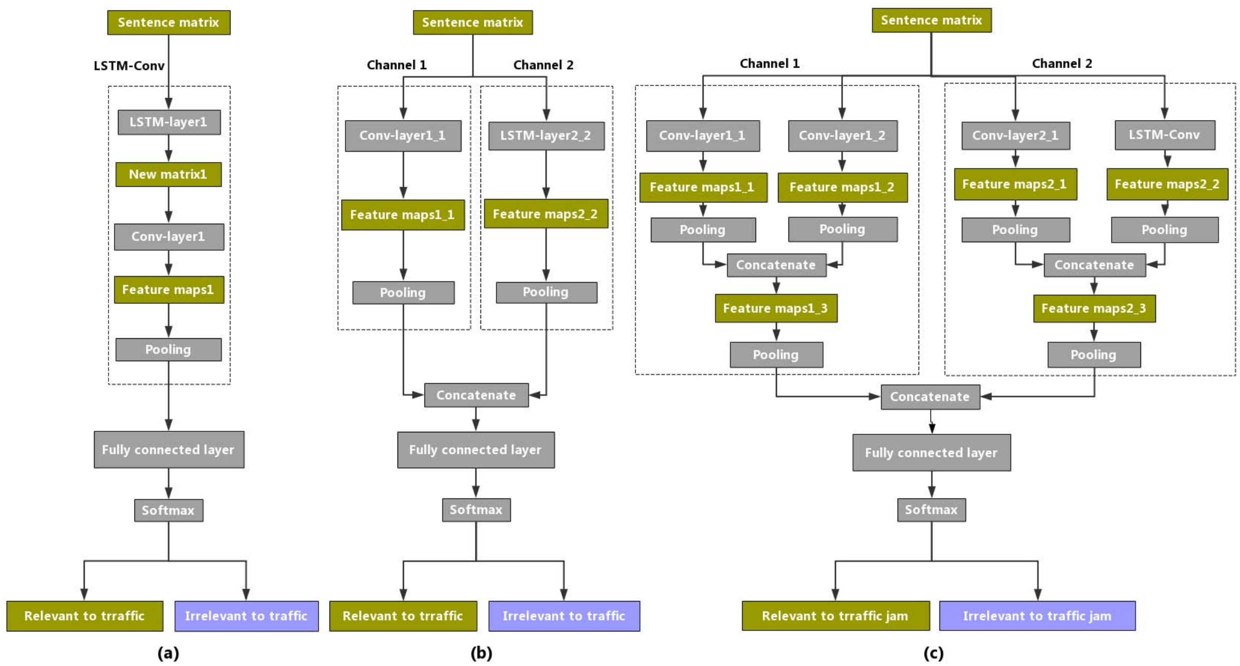


Fig. 3. The LSTM-CNN models proposed in [23], [24] and the MC-LSTM-Conv model proposed in this paper.

microblogs randomly crawled from the Sina Weibo online platform without coordinates and specific subjects. Suppose the input of the Skip-Gram model is a sentence consisting of n words $w_1, w_2, w_3 \ldots w_{(n-2)}, w_{(n-1)}, w_n$. In the process of training the Skip-Gram model, using target word w_t as an input to the model, the selection of the neighbor window size is set to 1 and the contexts $w_{(t-1)}$ and $w_{(t+1)}$ are labeled as positive samples. Meanwhile, the selection of negative samples as noise is random. At each iteration, 64 negative samples were randomly selected to update the corresponding weights in the hidden layer. We define the objective function of the Skip-gram model as follows.

$$L = \sum_{w \in C} \log p(Context(w_t)|w_t). \tag{1}$$

where L is the sum value of the log-probability (use target word w_t to infer its context), w_t represents the target word, $Context(w_t)$ represents the context of w_t, and C denotes the set of the target words. For the Skip-gram model, there is only one word in C. According to the thought of the hierarchical soft-max [26], the structure of the conditional probability function is as follows.

$$\begin{aligned} p(Context(w)|w) = \prod_{u \in Context(w)} \prod_{j=1}^{l^u-1} [\delta(v(u)^T \theta_j^u)]^{1-d_{j+1}^u} \\ *[1-\delta(v(u)^T \theta_j^u)]^{d_{j+1}^u} \end{aligned} \tag{2}$$

where $v(u)$ represents the vector of word u, and θ_j^u denotes the vector of the j_{th} non$-$leaf node on the path from the root to word w in the Huffman tree. d_j^u is the code corresponding to the j_{th} node, and we set the number of nodes in the path to word u as $L(u)$, $j = \{1, 2, \ldots, L(u) - 1\}$. Substitute the conditional probability function (2) into the objective function (1) to yield

$$\begin{aligned} \text{Ł} &= \sum_{w \in C} \log \prod_{u \in Context(w)} \prod_{j=1}^{L(u)-1} [\delta(v(u)^T \theta_j^u)]^{1-d_{j+1}^u} \\ &\quad *[1-\delta(v(u)^T \theta_j^u)]^{d_{j+1}^u} \\ &= \sum_{w \in C} \sum_{u \in Context(w)} \sum_{j=1}^{L(u)-1} (1-d_{j+1}^u) \log[\delta(v(u)^T \theta_j^u)] \\ &\quad + d_{j+1}^u \log[1-\delta(v(u)^T \theta_j^u)] \end{aligned} \tag{3}$$

The log-likelihood function (3) is the final objective function. Our training aims to maximize L. Moreover, the stochastic gradient descent algorithm is adopted here. After training for 500,000 iterations, a word embedding dictionary of 9,990 words is obtained with each word corresponding to a 128-dimensional vector, which can reflect the semantic feature of this word. To ensure that all sentences have the same length, those sentences with less than 125 words will be padded to 125 words with null values, and the sentences with more than 125 words will discard words after the 125th word. Using this dictionary, we can convert each sentence to a matrix with a size of 125×128. The matrix mentioned above is called the sentence matrix in this paper, and we can extract the abstract feature from the sentence matrix X.

2) Feature Extraction and Classification: Through the text representation step, we can convert the microblog text into a sentence matrix. Given the sentence matrix X, we propose MC-LSTM-Conv to conduct feature extraction and use these features to extract microblogs about traffic jams from mass data. Fig. 3 depicts the structures of the LSTM-CNN methods proposed in [23], [24] and the MC-LSTM-Conv used in this paper. As shown in Fig. 3(a), the sentence matrix X is firstly fed to the LSTM layer and the output matrix of the

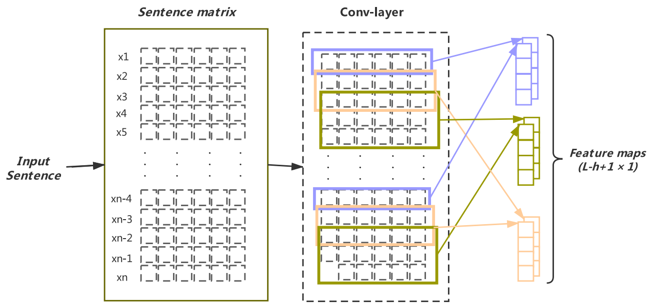


Fig. 4. Illustration of the Conv-layer for semantic analysis.

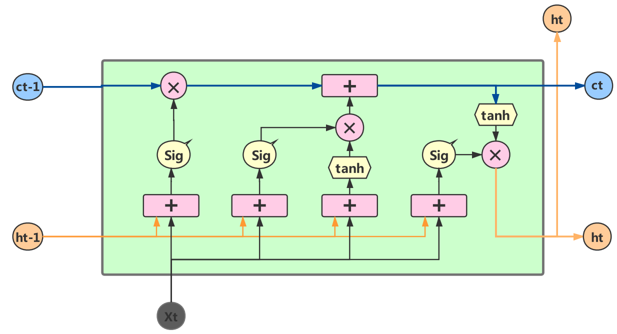


Fig. 5. Illustration of the memory cell in the LSTM-layer.

LSTM layer is passed to the Conv-layer. We name this part consisting of the LSTM layer and Conv-layer as LSTM-Conv, which is shown by the dotted line box of Fig. 3(a). Then, a fully connected (FC) layer is connected to the LSTM-Conv to perform classification. As shown in 3(b), the features of the sentence matrix X are simultaneously extracted by the LSTM-layer and Conv-layer in the multichannel approach. Fig. 3(c) shows that the proposed MC-LSTM-Conv model adopts the two information channels. We define the channel as a pipelined structure containing the Conv-layers, LSTM-layer and pooling layers. Given an input sentence, each channel has its way of capturing target information into feature vectors. The effectiveness and specificity of the extracted information depend on the structural design of the information channel. Channel 1 is connected to two parallel Conv-layers, which share the same parameter setting for filters. The feature maps outputted by these two Conv-layers are then processed by the 1-max pooling strategy and concatenated as a new feature map, which represents the local dependencies between neighboring words. Channel 2 is connected to two Conv-layers in parallel, which also share the same parameter setting for filters. With the LSTM-layer, the word embedding of each word in the sentence is influenced by the preceding words and has a different meaning. The same as Channel 1, the results outputted by two Conv-layers are pooled and concatenated, and they represent the contextual information of the sequential data. Finally, the results of the two channels are concatenated and connected to an FC layer. Fig. 3(c) shows that the MC-LSTM-Conv model contains two essential components, i.e., the Conv-layer and the LSTM layer. The detailed interpretation of the two components is given in the following.

The Conv-layer is adopted to extract the useful local dependencies among adjacent words. Fig. 4 gives a detailed illustration of the Conv-layer for semantic analysis. The colored boxes in Fig. 4 denote the filters used in the Conv-layer. We set the numbers of filters as n. The size of each filter is represented as $h \times k$, where h indicates the number of adjacent words covered by a filter and k is the length of the word vector. The convolution operation can be formulated as

$$c_i = f(\sum_{j,k} w_{j,k}(x_{[i:i+h-1]})_{j,k} + b). \tag{4}$$

where b represents the bias, $f(X)$ represents the rectifier linear unit (ReLU) activation used in this paper, and the size of the output feature map c_i is $L - h + 1 \times 1$. The 1-max pooling strategy is adopted for each feature map to calculate the maximum value and obtain the most important critical value of c_i. Next, we concatenate the pooled results as a vector with a size of n, and input it to the FC layer. The final classification can be implemented by using a soft-max layer.

Note that the Conv-layer cannot effectively exploit the long-distance dependencies of the sequential data; therefore, an LSTM-layer, which can remember the contextual information over multiple time steps, is added to the Conv-layer of LSTM-Conv. As shown in Fig. 5, when feeding the information to the LSTM cell, it will be first judge whether the information is useful according to the algorithm rules. Three non−linear gates, namely, the input gate, output gate and forget gate, are placed in the cell. The input gate controls the input of the current cell, the forgot gate controls the historical information stored in the cell at the last time, and the output gate controls the output. The state of the input gate, output gate and forgot gate at time t are denoted as i_t, o_t, and f_t respectively. The state update calculation method of these three gates can be defined as

$$i_t = \delta(W_i x_t + U_i h_{t-1} + b_i). \tag{5}$$

$$o_t = \delta(W_o x_t + U_o h_{t-1} + b_o). \tag{6}$$

$$f_t = \delta(W_f x_t + U_f h_{t-1} + b_f). \tag{7}$$

where δ respectively denotes a logistic sigmoid function: W_i, U_i, b_i, W_o, U_o, b_o, W_f, U_f and b_f represent the two weight matrices and the bias learned by the input gate, output gate and forgot gate, respectively. The input gate and the forgot gate together constitute the update gate. With W_c, U_c and b_c representing the two weight matrices and the bias of the update gate, the state of this cell at time t, c_t, can be defined as

$$c_t = i_t \otimes tanh(W_c x_t + U_c h_{t-1} + b_c) + f_t \otimes c_{t-1}. \tag{8}$$

where h_t denotes the final output of the LSTM unit, and the calculation formula is defined as

$$h_t = o_t \otimes tanh(c_t). \tag{9}$$

These gates can regulate the information flow into and out of the memory cell and effectively avoid gradient explosion and gradient diffusion. Therefore, we can use the LSTM-layer to dispose the sentence matrix and obtain a new matrix with the same size, which is shown in Fig. 6. The new matrix can be regarded as a new encoding of the original sentence matrix.

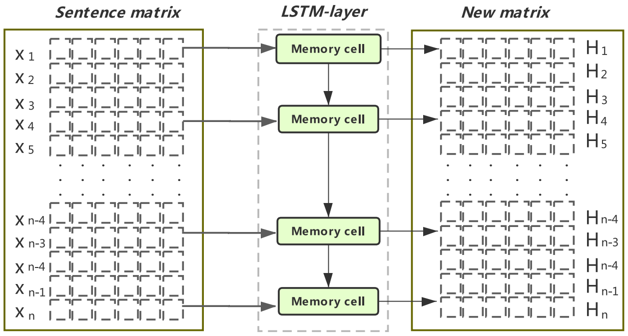


Fig. 6. Using the LSTM-layer to process the sentence matrix and obtain a new matrix with the same size.

In the new matrix, the word embedding of every word is influenced by the preceding words and has a different meaning.

B. Keyword Fuzzy Matching Method to Mine Detailed Traffic Information

With the MC-LSTM-Conv method, we have obtained a set of check-in microblogs about traffic jams from the mass Sina Weibo data. Further on, to provide more detailed information about the traffic jams in various regions, we classify the microblogs reflecting traffic jam events into four classes as follows:

1. Related to traffic accidents or large-scale activities (including car crashes, illegal lanes, concerts, etc.),
2. Related to road construction,
3. Related to traffic light failures or unreasonable settings, and
4. Related to the work of government agencies.

Therefore, we need a simple and effective method to matching the microblogs of one of the above categories from the obtained check-in microblogs about traffic jams. For the users of the Google Search Engine, they can set one or more keywords, and limit the length of the interval between the keywords, to query the information with the highest matching degree from the Internet. We refer to the advanced search syntax of Google and establish the corresponding rules by summarizing the keywords that best reflect the traffic information, as well as the common collocation of these keywords.

For the first class of microblogs, the matching rules include the following: 1. the noun traffic light and the description of its damaged state, and 2. the noun traffic light and opinions on the length of time. For the second class of microblogs, the matching rules require the following: 1. the noun road and the description of the maintenance behavior, and 2. a phrase or word describing road construction. For the third class of microblogs, the matching rules require the following: 1. a phrase or word describing a vehicle failure and lane occupying behaviors, 2. a phrase or word describing traffic accidents and lane occupying behaviors, and 3. a noun related to a major event, such as concerts, food festivals, or the names of Chinese stars. For the fourth class of microblogs, the matching rules require the following: 1. a phrase or word describing traffic violations, 2. the nouns government and taxpayers, and 3. a phrase or word describing public dissatisfaction with the work

TABLE I
MATCHING RULES FOR EACH CLASS

Class	Rules
1. Related to traffic light setting	1.1. (Red. * light |green. * light | traffic light | red. * green | signal | traffic light). {0,15}(Destroy | bad | stop | failure | black | off | flash) 1.2. (Design | set | unreasonable | remove | install | hang | without | adjust | meaning). {0,15}(Red. * light | green. * light | traffic light | red. * green | signal | traffic light) 1.3. (Red. * light | green. * light | traffic light | red. * green | signal | traffic light). {0,15}(Minute | second | long | half a day)
2. Related to road construction	2.1. (Fix | maintain | arrangement). *(road | bridge | high speed | road) 2.2. (Construct | pipeline | widen | repair road | trim)
3. Related to traffic accidents or large-scale activities	3.1. Stop|Occupy.{0,5}(channel|road|way | exit) 3.2. Scratch | hits the car | gets hit|break down 3.3. Accident | rear-end collision | accident | over turned | randomly parked | illegally parked 3.4. Fire | fire | spontaneous combustion 3.5. Emergency | dedicated lane | racketeer | yellow line | obstruct traffic | illega | solid line 3.6. May Day | Chen yi xun | jay | zuer | dehua | deng ziqi | leehom— Marathon | food street | concert
4. Related to the work of government agencies	4.1. Occupy.{0,5}(channel|road|way|direction) 4.2. Government | administration |traffic police | uniforms | leaders 4.3. (Violate. *rules)|illegal 4.4. Manage | regulate | deal with | control | manage | punish | check 4.5. Ignore | taxes | complaints 4.6. Citizens | taxpayers | the public | citizens

of the government. Based on these requirements, we establish several matching rules for each class according to regular expressions. Since there are many ways to express the same meaning in Chinese, part of these rules are translated into English (see Table I). In these matching rules, "|" represents the logic "or". In addition, "." denotes matching any character. "*" means to match the previous expressions any times. "{m, n}" means to match the previous expression at least m and at most n times.

Notably, all the microblogs reflect traffic jam events, and so the simple rules displayed in Table I can classify the microblogs into four classes.

III. EXPERIMENT

Two experiments are designed to evaluate the performance of the proposed method. In experiment 1, we explore the optimal parameter setting for the Conv-layers in the MC-LSTM-Conv model and verify its effectiveness and superiority through a comparison with other models. In experiment 2, we investigate the performance of the keyword fuzzy matching method in extracting the detailed information of traffic jams.

A. Dataset Description

Using the Sina Weibo application programming interface (API) with some keywords, we obtain mass Sina Weibo data consisting of two parts. One part is check-in microblogs, which are the primary social media data used in this paper. These kind of data are vast and reflect diverse perspectives about the traffic conditions. The other part consists of approximately 1 million common microblogs that are randomly crawled, without coordinates and specific subjects. These microblogs can be used as the corpus to train the Skip-Gram model. These data are available via the Sina Weibo API for reading public microblogs, and thus its collection is not detailed here. The first sort of microblogs is obtained through the Sina Weibo API for reading geographic information. Without setting keywords, we have received a total of 80,000,000 pieces of check-in microblogs in the region of Guangdong province from 2013 to 2017, including 110,000 pieces that contain keywords, such as "jam (du-堵)", "plug (sai-塞)" and "accident (shigu-事故)". We select the data with the time range of 2016-2017 for the training, verification and testing of the MC-LSTM-CNN. After the preliminary filtering, whether a sample is positive or negative is labeled manually. To ensure the consistency of the sample labeling standards, the specific rules are set as follows.

Most samples are divided into positive samples that could reflect a road jam with timeliness and negative samples that could not. Those microblogs irrelevant to traffic jams or reflect traffic conditions but without timeliness are all classified as negative samples. Neither the positive samples nor negative samples contain a large number of obscure words and dialects. In addition, the positive sample must be able to describe the a jam directly, and the people participating in manual labeling should not speculate whether a jam occurs. For example, take "我花了一小时才开了一公里, 真心塞 (It took me an hour to drive a kilometer and it was real distressed)". In this case, the reason for the slow driving speed may be a road jam or a vehicle failure. Another example "好堵的一天啊!!(What a day in traffic jam!)" cannot convey that the author is experiencing a traffic jam at that moment.

B. Performance Indexes of Classification Models

To examine the deep learning model proposed in this paper, four statistical indicators are adopted: Accuracy(Acc), Precision(Pre), Recall(Rec) and F1-score($F1$). The mathematical formulation of the four statistical indicators are shown in Table II.

Tp means the number of positive samples correctly classified into the positive class by the classifier. Tn means the number of negative samples correctly classified into the negative class. Fp means the number of negative samples wrongly classified into the positive class, and Fn means the number of positive samples wrongly classified into the negative class.

C. Results of Experiment 1

To obtain the best hyperparameters for the MC-LSTM-Conv model, we conduct multiple experimental investigations on the numbers and sizes of the filters in each channel. The performance is mainly measured by the classification accuracy on the same test dataset. We conduct only one experiment for each group of parameters, with the same samples for training and testing.

TABLE II
PERFORMANCE INDEXES

$$Accuracy(Acc) = \frac{T_p + T_n}{T_p + T_n + F_p + F_n}. \tag{10}$$

$$Precision(Pre) = \frac{T_p}{T_p + F_p}. \tag{11}$$

$$Recall(Rec) = \frac{T_p}{T_p + F_n}. \tag{12}$$

$$F1\text{-}Score(F1) = 2 * \frac{Pre \times Rec}{Pre + Rec}. \tag{13}$$

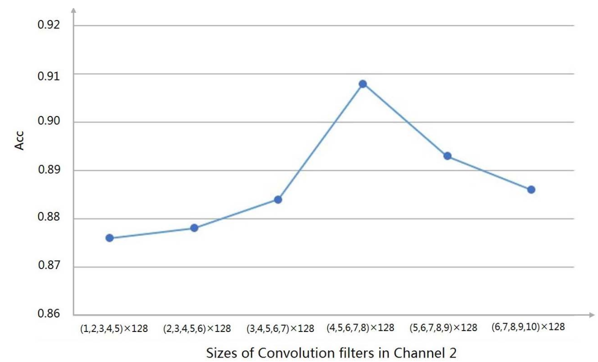


Fig. 7. Filter size setting for Conv-layers in two channels.

1) Analysis of the Filter Size in Each Channel: To explore the appropriate filter sizes for each channel, we set the numbers of filters of various sizes to 100. The width of all filters is set to 128, which equals to the width of the sentence matrix X. Channel 1 is designed to extract the n-gram features of words in the small neighborhood. Therefore, the filter sizes in Channel 1 are set to 1×128, 2×128, 3×128, 4×128, and 5×128, respectively, which are mainly concentrated in the range of smaller sizes. Moreover, Fig. 7 plots the performance of multigroup experiments with different filter sizes. As shown in Fig. 7, as the filter sizes in Channel 2 increase, the classification accuracy first increases and then decreases. The experimental results demonstrate that the Conv-layer works better with the LSTM-layer under the fourth set of filter sizes, achieving the most accurate model.

2) Analysis of the Number of Filters in Each Channel: When exploring the suitable numbers of filters for two channels, the sizes of the filters in Channel 1 are configured to 1×128, 2×128, 3×128, 4×128, and 5×128; and the sizes of the filters in Channel 2 are set to 4×128, 5×128, 6×128, 7×128, and 8×128. Fig. 8 shows the varying

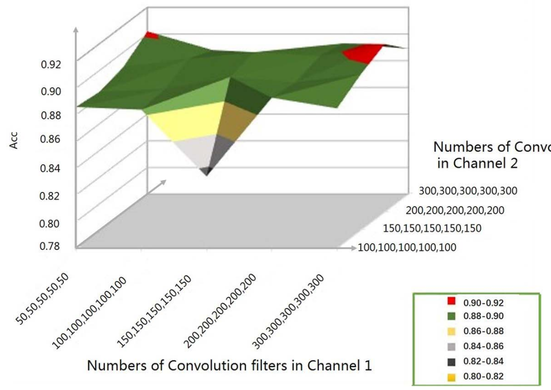


Fig. 8. Setting of the numbers of filters for the Conv-layers in two channels.

performance of the MC-LSTM-Conv model with different numbers of filters. In Fig. 8, several sets show the promising performance, especially this one (50, 50, 50, 50, 50, 300, 300, 300, 300, 300), which not only achieves ideal accuracy but also uses fewer filters. To further reduce the computations and improve the performance, we conduct several more sets of experimental investigations and each set of parameters is validated by 10 times 10-fold cross validation. Finally, we obtain the best hyperparameters for the Conv-layers as follows: in Channel 1, the filter lengths are set to 1, 2, 3, 4, and 5, and the numbers of filters are 300, 300, 300, 200, and 200, respectively; and in Channel 2, the filter lengths are set to 4, 5, 6, and 7, and the numbers of filters are 50,50,50, and 50, respectively. As a result, the Acc and the Pre both reach 0.923 and the Rec reaches 0.920.

3) The Classification Effects of MC-LSTM-Conv and Other Comparative Models: The superiority of MC-LSTM-Conv originates from the local dependency modeling by the convolutional network and the long-distance dependency by LSTM. To validate this point, we have conducted a comparative experiment, where the LSTM-Conv in channel 2 is replaced by an ordinary Conv-layer. The multichannel network with the mere Conv-layer is termed as MC-Conv. After conducting 10 times 10-fold cross-validation, the Acc is 0.917, the Pre is 0.919, and the Rec is 0.912. Moreover, we also compare the classification performance of MC-LSTM-Conv with other models, such as the CNN, LSTM, RNN, SVM, LSTM-CNN [23] and another LSTM-CNN [24]. Each method mentioned above is also evaluated with 10 times 10-fold cross validation.

In the comparison of MC-LSTM-Conv with other models, we kept the parameter settings of each model as similar as possible, especially regarding the size and number of filters in the deep learning models. Regarding the optimal hyperparameters of MC-LSTM-Conv, the number of convolution filters is 1500, and the length of the convolution filters varies from 1 to 7. Moreover, the width of all filters is set to 128. Therefore, the filter sizes applied in CNNs and LSTM-CNNs [23], [24] are set as 1×128, 2×128, 3×128, 4×128, 5×128, 6×128, and 8×128, and the numbers of filters are 300, 300, 300, 200, 200, 100, and 100, respectively. The memory dimension of LSTM and MC-LSTM-Conv are both set to be 128, which must equal the width of sentence matrix X. Except for the

TABLE III
THE NUMBER OF TRAINABLE PARAMETERS FOR EACH MODEL

Id	Method	Parameters
1	MC-LSTM-Conv	739,184
2	MC-Conv	607,600
3	CNN	646,000
4	LSTM-CNN [23]	777,584
5	LSTM-CNN [24]	971,084
6	LSTM	33,280
7	RNN	131,968
8	SVM	-

TABLE IV
THE CLASSIFICATION PERFORMANCES OF DIFFERENT METHODS

Id	Method	Acc	Pre	Rec	F1
1	MC-LSTM-Conv	**0.923**	**0.923**	0.920	**0.921**
2	MC-Conv	0.917	0.919	0.912	0.915
3	CNN	0.919	0.914	**0.922**	0.918
4	LSTM-CNN [23]	0.905	0.902	0.903	0.902
5	LSTM-CNN [24]	0.910	0.907	0.908	0.907
6	LSTM	0.893	0.885	0.896	0.890
7	RNN	0.869	0.870	0.870	0.870
8	SVM	0.757	0.755	0.752	0.753

SVM, the learning rate of all other models in the compared experiments is 1e-4, and the dropout rate is 0.5. During the training process of the MC-LSTM-Conv and all comparative models, the number of epochs is set to 30.

For MC-LSTM-Conv and other deep learning models, the number of trainable parameters (i.e., the parameters whose values update in the back-propagation process) is one of the important indexes for calculating the model's space complexity. Therefore, we list the number of trainable parameters for each model in Table III. Among them, the SVM requires few parameters. On the contrary, deep learning based models require many more parameters to extract the features of the input sentence matrix. Among them, the LSTM-CNN [24] has the most parameters because in the LSTM channel, an extra Conv-layer is designed to keep the output shape consistent with other channels. The number of trainable parameters of MC-LSTM-Conv is close to that of other deep learning models. Therefore, the increase in the spatial complexity of MC-LSTM-Conv is not significant.

As shown in Table IV, the MC-LSTM-Conv model is significantly superior to LSTM, the RNN and the SVM. Compared with the CNNs, MC-LSTM-Conv also shows a competitive Acc and Pre while Rec is slightly lower, which means that MC-LSTM-Conv can more effectively and accurately identify traffic jam information from a large number of microblogs. When performing the classification task on the test dataset with 1,055 microblogs, the CNN costs 2.97 seconds, and the MC-LSTM-Conv requires 4.78 seconds. In addition, MC-LSTM-Conv performs better than the LSTM-CNN [23] and the LSTM-CNN [24], showing its adaptability on our microblog dataset.

4) The Detailed Comparison Between MC-LSTM-Conv and the CNN: We statistically analyze the classification results of the MC-LSTM-Conv and CNN models on the test dataset, and we find that once the text length is more than 100, the indexes

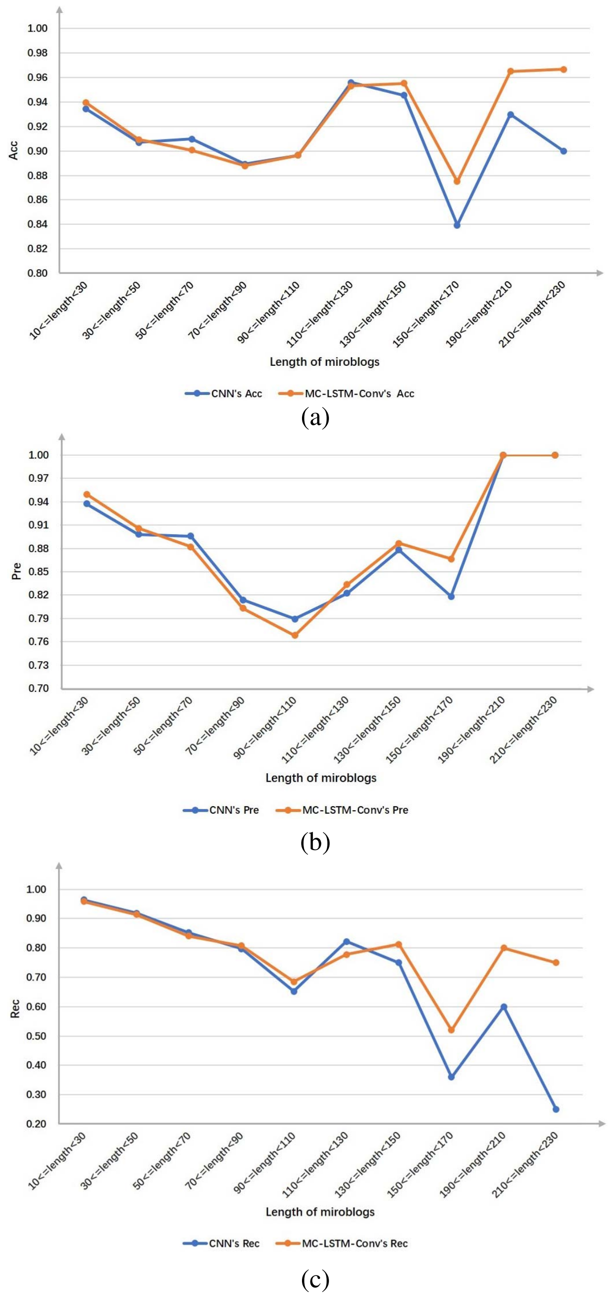


Fig. 9. The Acc,Pre and Rec of the CNN and MC-LSTM-Conv vary with the length of the microblog.

of MC-LSTM-Conv are significantly better than those of the CNN, including the Rec, and MC-LSTM-Conv also performs well in the samples when the text length is less than 100. It proves that our model is more adaptable to the text length, and it is more suitable for the task with a variable length of text. Fig. 9 shows that the Acc, Pre and Rec of the CNN and MC-LSTM-Conv vary with the length of the microblog.

To validate that MC-LSTM-Conv performs better for longer microblog text, we group the false classifications of the MC-LSTM-Conv and CNN models according to the length of the microblog text, and study the performance of the model for different length intervals. The results are listed in Table V. In the range of [10, 50), the misclassification of MC-LSTM-Conv is less than that of CNN, especially for the Fps, the

TABLE V

THE NUMBERS OF Fps AND Fns FOR DIFFERENT TEXT LENGTH RANGES

Length of text	Fn(CNN)	Fn(MC-LSTM-Conv)	Fp(CNN)	Fp(MC-LSTM-Conv)
$[10, 30)$	108	125	193	152
$[30, 50)$	105	111	135	123
$[50, 70)$	66	71	44	50
$[70, 90)$	39	37	35	38
$[90, 110)$	32	29	16	19
$[110, 130)$	8	10	8	7
$[130, 150)$	12	9	5	5
$[150, 170)$	16	12	2	2
$[170, 130)$	5	4	0	0
$[190, 150)$	4	2	0	0
$[210, 230)$	3	1	0	0
$[230, 250)$	0	0	0	0

MC-LSTM-Conv performs better. In the range of [50, 70), the CNN's performance is slightly better. In the range of [70, 130), the difference between the CNN and MC-LSTM-Conv is not apparent. When the length of the microblog goes beyond 130, the classification effect of MC-LSTM-Conv is significantly better, which is more reflected in its correct identification of positive samples. MC-LSTM-Conv's advantage in recognizing longer sentences should be attributed to the addition of the LSTM-layer. The existence of the LSTM-layer enhances the model's ability to recognize and remember the contextual information in sentences so that the model can keep the information concerning traffic jams in the output matrix of Channel 2 rather than let it be ignored.

5) Applying MC-LSTM-Conv to Extract Historical Microblogs About Traffic Jams: The proposed MC-LSTM-Conv model is used to classify all 110,000 check-in microblogs from 2013-2017 consisting of "jam(du-堵)", "plug(sai-塞)", "accident(shigu-事故)", "traffic(jiaotong-交通)", etc. Approximately 40,000 microblogs relevant to traffic jams are obtained, as well as 70,000 microblogs irrelevant to traffic jams. According to the longitude and latitude information described in text, we use World Geodetic System 1984 as the spatial reference and visualize the microblogs relevant to traffic jams in Fig. 10, which simultaneously uses Guangdong's road network map as the base map. We can easily find that the two kinds of microblogs' spatial distribution characteristics are different in Fig. 10. The microblogs reflecting road jams are concentrated in urban construction areas, including Shenzhen, Guangzhou, Foshan, Maoming, Shantou and Chaozhou, etc. They are distributed along the road network and concentrated at road intersections and surrounding areas. The distribution of the other kind is also mostly concentrated in urban construction areas, but they are relatively scattered, with less spatial correlation to the road network. This indicates that the density of microblogs is higher in areas where human activities are more frequent.

D. Results of Experiment 2

1) The Classification Effects of the Keyword Fuzzy Matching Method: We use a dataset of 5,300 positive samples, which can reflect traffic jams, to train the MC-LSTM-Conv model in experiment 1. However, apart from revealing the existence of traffic jams, most of these positive samples do not contain

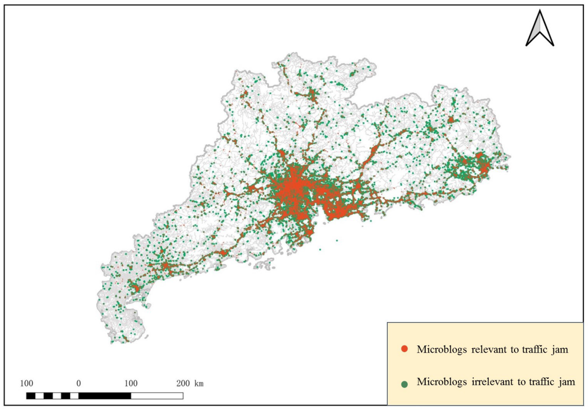


Fig. 10. Distribution of microblogs relevant to jams and microblogs irrelevant to jams.

TABLE VI
THE CLASSIFICATION EFFECTS FOR DIFFERENT CLASSES

Id	Pre	Rec	F1
1	0.886	0.802	0.840
2	0.816	0.886	0.849
3	0.879	0.804	0.837
4	0.895	0.993	0.942

other information about road conditions. Only 600 microblogs with detailed road information are manually selected and classified into four classes: 1. related to traffic accidents or large-scale activities (including car crashes, illegal occupation of lanes, concerts, etc.), 2. related to road construction, 3. related to traffic light setting (including faults, positions or time, etc.), and 4. related to the work of government agencies. These four classes are associated with each other, and there are some microblogs that simultaneously belong to more than one category. The numbers of samples vary greatly for the four classes, which are 200, 50, 50, and 450, respectively. These 600 labeled samples are used to validate the keyword fuzzy matching method's capability to extract detailed traffic information.

When we evaluate the performance of the rules for a specific class, neither the performance of the regulations related to other classes nor whether the sample belongs to other classes is considered. Statistically, the value of the Acc in each class is above 0.98. However, when the negative samples are far more than the positive samples, the value of the accuracy mainly reflects the recognition ability of the negative samples. Therefore, when evaluating the performance of the keyword fuzzy matching method, we focus on the Pre,Rec, and $F1$. The calculation results for each class are shown in Table VI.

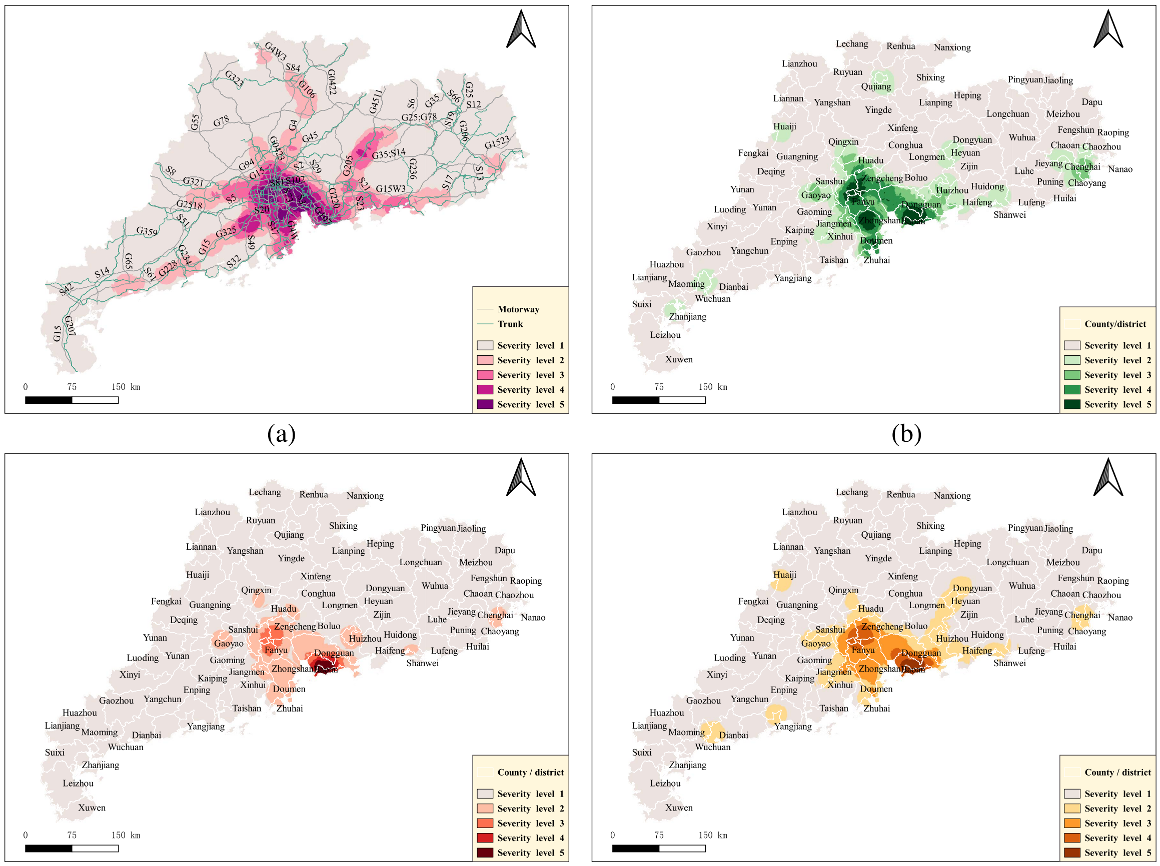
2) The Detailed Comparison of the Classification Effects for Different Classes: As shown in Table VI, the model's Pre, Rec, and $F1$ for the four classes are all higher than 0.80, proving that the microblog classification rule is effective. The classification effect for class-1 and class-3 are similar with a high Pre and a low Rec. The matching rules for these two classes have specific description objects, among which, the former describes the state of traffic lights, and the latter describes different accidents and activities. When the microblog contains these noun keywords, it is likely to belong to this class. Since the noun keywords in rules cannot be collected completely, part of the useful microblogs cannot be classified precisely to these two classes. The classification effects for class-2 result in a low Pre and high Rec. In Chinese, there are many synonyms for road construction. For example, both "拓宽(broaden)" and "加宽(widen)" can be used to mean to widen the road. To avoid missing information, we use the core keyword "宽" to establish the rules. However, such a process will lead to a bit of precision loss. Among the rules for class-4, one part is used to match microblogs that express dissatisfaction with the government and traffic police, and the other part is used to match reporting illegal behaviors. The former always has specific description objects, which make them easier to collect completely. In the latter, there are limited ways to describe these illegal traffic behaviors. Therefore, the classification effect on class-4 is superior to that on other classes.

3) Applying the Keyword Fuzzy Matching Method to Analyze Traffic jam Causes: We apply the keyword fuzzy matching method to 40,000 microblogs reflecting traffic jams obtained from experiment 1. This dataset is classified into four classes, analyzed by the point density method [30] and visualized as Fig. 11. These density figures are clipped by the boundaries of counties in the Guangdong province, and the density values are divided into five levels to evaluate the degree of congestion.

The four classes of microblogs are all highly concentrated in the heart of the region of the Pearl River Delta, including Guangzhou, Foshan, Shenzhen, Dongguan, Huiyang, Jiangmen, Zhuhai, etc. Since these places are the most densely populated regions in Guangdong province, and the probability of traffic jams is higher than in other areas. Moreover, the overall number of microblogs published there is higher than that in other regions, which is also the main reason for the spatial concentration phenomenon. Outside the heart of the region of the Pearl River Delta, these four classes of microblogs have their own spatial distribution characteristics.

The microblogs related to traffic accidents or large activities are shown in Fig. 11 (a). By overlaying the distribution map of trunk roads and motorways on the density map, we can find that these microblogs are more widely distributed along the main roads and they appear linearly. For example, the motorway named G15 and the trunk road named G325 pass by Kaiping, Enping, Yandong, Yangxi, Dianbai, Maoming, and other regions. In these regions, there are numerous microblogs describing traffic accidents causing congestion. The timeliness of microblogs makes them have advantages in dealing with accidents, which may cause serious blockages on the whole route in a short time. In addition, the information conveyed by microblogs, especially the cause and severity of the accident, is extremely important to the arrangement of the traffic police.

The microblogs related to road construction are shown in Fig. 11 (b). We utilize the city boundary map to partition the density map. Such microblogs have formed significant clusters in the Chaozhou and Shantou regions in eastern Guangdong. The others are distributed discretely in several areas, such as Zhanjiang, Maoming, Shaoguan, Heyuan, Haifeng, and Qujiang. This distribution characteristic means that road



(c) (d)

Fig. 11. Spatial distribution of the four classes of jam events: (a) shows traffic jam events related to the accidents on the road; (b) shows traffic jam events related to the road construction; (c) shows traffic jam events related to the traffic light; (d) shows traffic jam events related to the work of government.

construction events have frequently occurred in certain places. Since road congestion caused by construction often lasts for a long time, local transportation authorities need to prepare alternative channels for construction sections, so as to prevent serious congestion.

The microblogs related to the failure or unreasonable configuration of traffic lights are shown in Fig. 11 (c). Since the spatial distribution of Fig. 11 (c) is similar to that of Fig. 11 (b), we have also added the urban boundary to the map. It can be clearly seen from Fig. 11 (c) that such microblogs are mainly distributed in and around the Pearl River Delta, and there are few in other regions. When optimizing the time allocation or positions of traffic lights, the traffic department can take the public's suggestions conveyed by microblogs and other social media into account.

The microblogs related to the work of government agencies are shown in Fig. 11 (d). Such microblogs have a similar spatial distribution as that of Fig. 11 (a). A large number of micro-blogs accumulated near roads, extending from the Pearl River Delta. Another part of the microblogs is distributed discretely in several areas, such as Dianbai, Yangdong, Huaiji, and Chenghai. By inspecting the IDs and contents of microblogs in the same place, it can be proved that many microblogs belong to both classes. These microblogs express the public's opinions and dissatisfaction toward the work of the local government and the reports about accidents and violations on road.

TABLE VII

THE MICROBLOGS WHICH DESCRIBE THE ROAD CONDITION

Id	Microblog text
1	An illegal parking is happening at Chanchugang Road. @Guangzhou traffic police.
2	The male driver illegally parked on the road, causing the traffic jam. He also abused the traffic warden who asked him to leave.
3	Traffic accidents are frequent at the crossroads of the overpasses.
4	The daily traffic on Zhixin south road is too crowded, and the vehicles parked at will. Why not install surveillance cameras?
5	The reason buses cannot enter the bus station is that there are too many illegal vehicles blocking the street.
6	No one cares the casual parking. The government should pay attention to solve the problem.
7	There is a car going in the wrong way on Nonglin road. @Guangzhou traffic police.

The combination of MC-LSTM-Conv and Keyword Fuzzy Matching provides us with a new method to study the urban traffic problem. According to the speed data provided by Baidu Map, the area around the Zhixinnan Road has an average speed of approximately 13 km/h at the evening peak. Based on the keyword fuzzy method and the MC-LSTM-Conv, we obtain 25 microblogs that describe the local road conditions. Part of the microblogs are translated into English and showed in Table VII. As these microblogs describe, illegal parking, lane occupation and other traffic violations frequently occur, affecting the regular travel of citizens and easily causing congestion. After field visits and investigations, the above situation has been confirmed. The frequent occurrence of road violations reflects that the work efficiency of the road management departments in this region needs to be strengthened.

IV. Conclusion and Future Works

This paper aims to make the best of social media data in monitoring traffic conditions and acquire useful traffic information as fully as possible. We propose a deep learning-based method to extract traffic information. For the binary classification problem (traffic jam versus non-traffic jam), the MC-LSTM-Conv model has been proven to provide higher accuracy and precision than other deep learning neural networks (CNN, LSTM, RNN, SVM, LSTM-CNN, etc.). The advantages of MC-LSTM-Conv are that it is efficient at capturing the long-distance dependencies of the inputted sequential data and can to preserve the original semantic information to the greatest extent. These advantages make the new model suitable for dealing with text with an uncertain length (e.g. microblogs). Regarding the classification result, the model can exclude the microblogs without direct relationship with traffic jams. Based on the keyword fuzzy matching method, this paper classifies the extracted microblogs into four classes corresponding to the specific traffic jam scenes, which can help to improve the efficiency of traffic management.

Nonetheless, our study still has its limitations: First, the training process require a large number of manually labeled samples, which requires much manual labor; Second, there is subjectivity in the selection of the keywords for establishing the matching rules. Therefore, in future work, we will make the following improvements: First, we will use semi-supervised learning to reduce the dependence of the deep learning model on training sample size. Second, the matching rules will be more complete by adding more synonymous keywords calculated by word embedding.

References

[1] J. Li *et al.*, "Social media: New perspectives to improve remote sensing for emergency response," *Proc. IEEE*, vol. 105, no. 10, pp. 1900–1912, Oct. 2017.

[2] S. Swain and K. R. Seeja, "Analysis of epidemic outbreak in delhi using social media data," in *Proc. Int. Conf. Inf., Commun. Comput. Technol.* Singapore: Springer, 2017, pp. 25–34.

[3] H. Shekhar, S. Setty, and U. Mudenagudi, "Vehicular traffic analysis from social media data," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Sep. 2016, pp. 1628–1634.

[4] H. Mo, X. Hao, H. Zheng, Z. Liu, and D. Wen, "Linguistic dynamic analysis of traffic flow based on social media—A case study," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 9, pp. 2668–2676, Sep. 2016.

[5] A. Agarwal and D. Toshniwal, "Face off: Travel habits, road conditions and traffic city characteristics bared using Twitter," *IEEE Access*, vol. 7, pp. 66536–66552, 2019.

[6] A. Huang, L. Gallegos, and K. Lerman, "Travel analytics: Understanding how destination choice and business clusters are connected based on social media data," *Transp. Res. C, Emerg. Technol.*, vol. 77, pp. 245–256, Apr. 2017.

[7] D. Cao, S. Wang, and D. Lin, "Chinese microblog users' sentiment-based traffic condition analysis," *Soft Comput.*, vol. 22, no. 21, pp. 7005–7014, Nov. 2018.

[8] M. Maghrebi, A. Abbasi, and S. T. Waller, "Transportation application of social media: Travel mode extraction," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2016, pp. 1648–1653.

[9] A. Schulz, P. Ristoski, and H. Paulheim, "I see a car crash: Real-time detection of small scale incidents in microblogs," in *Proc. Extended Semantic Web Conf.*, 2013, pp. 22–33.

[10] A. Gal-Tzur, S. M. Grant-Müller, T. Kuflik, E. Minkov, S. Nocera, and I. Shoor, "The potential of social media in delivering transport policy goals," *Transp. Policy*, vol. 32, pp. 115–123, Mar. 2014.

[11] K. Fu, R. Nune, and J. X. Tao, "Social media data analysis for traffic incident detection and management," in *Proc. Transp. Res. Board 94th Annu. Meeting*, Washington, DC, USA, 2015, pp. 1–10.

[12] H. Nguyen, W. Liu, P. Rivera, and F. Chen, "Trafficwatch: Real-time traffic incident detection and monitoring using social media," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining*, 2016, pp. 540–551.

[13] L. Lin, M. Ni, Q. He, J. Gao, and A. W. Sadek, "Modeling the impacts of inclement weather on freeway traffic speed: Exploratory study with social media data," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2482, no. 1, pp. 82–89, Jan. 2015.

[14] M. Ni, Q. He, and J. Gao, "Forecasting the subway passenger flow under event occurrences with social media," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 6, pp. 1623–1632, Jun. 2017.

[15] X. Liu, X. Kong, and Y. Li, "Collective traffic prediction with partially observed traffic history using location-based social media," in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2016, pp. 2179–2184.

[16] L. Lin, J. Li, F. Chen, J. Ye, and J. Huai, "Road traffic speed prediction: A probabilistic model fusing multi-source data," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 7, pp. 1310–1323, Jul. 2018.

[17] E. D'Andrea, P. Ducange, B. Lazzerini, and F. Marcelloni, "Real-time detection of traffic from Twitter stream analysis," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 2269–2283, Aug. 2015.

[18] A. Salas, P. Georgakis, and Y. Petalas, "Incident detection using data from social media," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 751–755.

[19] C. Gutierrez, P. Figuerias, P. Oliveira, R. Costa, and R. Jardim-Goncalves, "Twitter mining for traffic events detection," in *Proc. Sci. Inf. Conf. (SAI)*, Jul. 2015, pp. 371–378.

[20] P.-T. Chen, F. Chen, and Z. Qian, "Road traffic congestion monitoring in social media with hinge-loss Markov random fields," in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2014, pp. 80–89.

[21] Y. Chen, Y. Lv, X. Wang, and F.-Y. Wang, "A convolutional neural network for traffic information sensing from social media text," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 1–6.

[22] Z. Zhang, Q. He, J. Gao, and M. Ni, "A deep learning approach for detecting traffic accidents from social media data," *Transp. Res. C, Emerg. Technol.*, vol. 86, pp. 580–596, Jan. 2018.

[23] P. M. Sosa. (2017). *Twitter Sentiment Analysis Using Combined LSTM-CNN Models*. Accessed: Dec. 23, 2019. [Online]. Available: https://www.academia.edu/35947062/Twitter Sentiment Analysis using combined LSTM-CNN Models

[24] Q.-H. Vo, H.-T. Nguyen, B. Le, and M.-L. Nguyen, "Multi-channel LSTM-CNN model for vietnamese sentiment analysis," in *Proc. 9th Int. Conf. Knowl. Syst. Eng. (KSE)*, Oct. 2017, pp. 24–29.

[25] W. Li and S. Hao, "Sparse lifting of dense vectors: Unifying word and sentence representations," 2019, *arXiv:1911.01625*. [Online]. Available: http://arxiv.org/abs/1911.01625

[26] T. Mikolov, I. Sutskever, C. Kai, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 26, 2013, pp. 3111–3119.

[27] T. Mikolov, W.-T. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2013, pp. 746–751.

[28] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013, *arXiv:1301.3781*. [Online]. Available: http://arxiv.org/abs/1301.3781

[29] X. Rong, "Word2vec parameter learning explained," 2014, *arXiv:1411.2738*. [Online]. Available: http://arxiv.org/abs/1411.2738

[30] N. Deshpande, I. Chanda, and S. S. Arkatkar, "Accident mapping and analysis using geographical information systems," *Int. J. Earth Sci. Eng.*, vol. 4, no. 6, pp. 342–345, 2011.

Yiwen Wang received the B.S. degree from the School of Mathematics, Jimei University (JMU), Xiamen, China, in 2017, and the master's degree with the School of Geography and Planning, Sun Yat-sen University (SYSU), Guangzhou, China, in 2019.

His research interests include geographic information systems, natural language processing, and social transportation systems.

Zhi He (Senior Member, IEEE) received the B.S. degree from the School of Mathematics, Harbin Institute of Technology (HIT), Harbin, China, in 2009, and the M.S. and Ph.D. degrees in control science and engineering from HIT in 2011 and 2015, respectively.

She is currently an Associate Professor with the School of Geography and Planning, Sun Yat-sen University (SYSU). Her research interests include Hilbert–Huang transform, sparse representation, and their applications to hyperspectral classification. She is a reviewer for several journals.

Jie Hu received the B.S. and master's degrees from the School of Geography and Planning, Sun Yat-sen University (SYSU), Guangzhou, China, in 2020.

Her research interests include geographic information systems, image super-resolution, deep learning neural networks, and their applications to hyperspectral classification.