Paper:

Store Scene Recognition and Classification via Multi-Stage Neural Network Model

Kejuan Yang, Sihan Gao, and Hongbin Ma

School of Automation, Beijing Institute of Technology
E-mail: 1120181103@bit.edu.cn, 1120181338@bit.edu.cn, mathmhb@139.com
[Received 00/00/00; accepted 00/00/00]

Abstract. In store scene classification, the variability and versatility lie in abnormal picture shooting angles, diverse text fonts, weak text context connection and samples lacking; hence, to tackle these problems, we propose a multi-stage model based on neural network architecture, which mainly features text information. This network combines three sub-modules—text detection, text recognition and semantic classification. First, a text detection and recognition module based on DBNet and CRNN is designed with a text orientation classifier. Meanwhile, augmented datasets are enlarged to improve the quality of text extraction from images. Then, the derived text acts as input to the Ernie text recognition transfer model. Finally, the store scene classification label is derived. Of note, our model is tested on the DC platform dataset, and the experiment shows that the proposed method extracts the features of store scene images well and classifies them effectively according to the derived context. Compared with current algorithms, our method prompts both classification robustness and generalization ability of the trained model.

Keywords: image recognition, text detection, text recognition, DBNet, CRNN, NLP, Ernie

1. Introduction

In the automatic classification task of store scenes, the key feature that conveys semantic meanings is the context within pictures. Detecting and recognizing text in the wild remains largely challenged due to its diversity patterns in sizes, light, font styles, and shapes. Thanks to the exploding of deep neural networks, abundant progress has been made in detection and recognition.

The text,as the state-of-the-art research did so far, is not alike other modalities such as image and sound where feature representation of can be easily derived via stochastic neural network method. For text semantic recognition, a dictionary needs to be organized first for Word frequency counting; then by referring to the external vocabulary, we train the network for store scene classification in a typical natural language processing manner. The methods above all highly rely on the quality of datasets, it is recommended to perform fine-tuning on a pre-trained model trained from groups of public datasets, in this way we successfully transfer and target our task of store scene recognition.

2. Related Work

Text spotting metrics are developed from spotting regular scene text to spotting arbitrarily-shaped scene text. Lee et al.[1] proposed an end-to-end text recognition model which only detects horizontal text and requires relatively complex training procedures. To tackle the problem of multi-orientated text, Sun et al.[2] generated text proposals in quadrangles, and encoded the aligned RoI features into context information with a recurrent neural network to generate the text sequences, however, it may suffer from background noises.

Lyu et al.[3] proposed to detect and recognize text of arbitrary shapes by segmenting the text regions and character regions. However, this method is costly for character-level annotations. Feng et al.[4] used a text detector to describe the shape of text with a series of quadrangles, and a feature operator to extract and rectify arbitrary shapes of text from feature maps. PGNet[5], proposed and optimized by Paddle, is a light text recognizer of arbitrary style form. PGNet predicts the text locations in fine-scale and explores rich information through images straightly. Nevertheless, These methods are all RoI-based methods and may involve NMS, RoI cropping, and pooling operations, which is time-consuming and may reduce the performance.

Semantic recognition aims to determine the correct relation between two entities between the given sentences. Pei Zhili et al.[6] introduced convolutional neural network methods employed in text classification, however, the multi-layer network could cause the problem of gradient disappearing. Vu et al.[7] demonstrated that CNN and RNN provide complementary information-while RNN computes a weighted combination of all words in the sentence, CNN extracts the most informative n grams for the relation and only considers their resulting activation.

For pre-trained language models, Mikolov et al.[8] proposed the feature-based approaches focusing on the compositionality of text representation. Dai et al.[9] proclaimed the result that it is possible to use unlabeled data

from related tasks to improve the generalization of a subsequent supervised model.

Different from all these methods, our metric divides scene recognition into three modules-text detection, text recognition and semantic recognition, with DBNet, CRNN, ERNIE structure respectively applied in those models to achieve better or competitive performance than end-to-end performance, as experiment shows in Section 4.

3. Network Architecture

Store recognition towards complicated street scenes exhibits variable fonts in texts, sequence characteristics in contexts, and abstract brand semantic information. These challenges are compounded by how to extract effective features from store images, then convert them into interpretable context information. To employ an end-to-end model or multi-stage model (text detection-text recognition-semantic recognition), it is a tradeoff between lightweight structure and interpretability of the model.

According to the information black-box in a store recognition task, it is difficult to directly train an end-to-end model for matching and classifying stores to their real categories, only relying on the information of images. Therefore, we propose a hybrid network based on the idea of "divide and conquer" to better interpret the store recognition task, which is followed by several subsequent steps: distinguishing text lines from the background by the detector and then extracting features in the text proposed by the decoder for further recognition. Experimental results have shown that the learning performance of our multi-stage hybrid network is much more powerful than that of previous end-to-ends, and each sub-module has strong interpretability. Furthermore, if back optimization is required to refine the trained model, it is easier to realize in a multi-stage structure.

3.1. Multi-Stage Model

The multi-stage model proposes that the task of text information extraction is divided into two sub-tasks: text detection and text recognition, that is, first detect the text proposal, and then extract the information of the text proposal for text recognition.

A typical target detection network is composed of a transformer, a backbone, a neck, a head, and other modules. The transformer is mainly used in text box detection tasks, which is responsible for pre-processing features. For instance, in a text scene with bending deformation, we use TPS (thin-plate splines) interpolation algorithm to separate control points, correct the text area into a rectangle, and then recognize.

For backbone networks, we use the mainstream network structures such as ResNet, MobileNet, and VGG, which are popular in deep learning. We load its pre-training parameters to fine-tune, to obtain good feature extraction ability.

Table 1. DB vs. EAST.

Model	Backbone	Precision	Recall
EAST	ResNet34-vd	85.80%	86.71%
EAST	MobileNetV3	79.42%	80.64%
DB	ResNet34-vd	86.41%	78.72%
DB	MobileNetV3	77.29%	73.08%

Head, that is, detector, is the module to obtain the network output content, and also the last layer of the whole network framework. It takes advantage of the features extracted before to predict labels for the classification results. Neck, located between the backbone and the head, extracts information of different scales from different feature maps and fuses them. It makes full use of all the feature information extracted by the backbone, enabling the network to detect objects of distinctive distances, scales, and perspectives.

Based on the framework of PaddleOCR[10], we divide the task of extracting image information in store recognition into three networks: text detector, text direction classifier, text recognizer, and train their models respectively. All the modules above adopt the underlying architecture of transformer-backbone neck-head.

3.2. Text Detection

The text detector is responsible for the box regression of the input image and the prediction of the location coordinates of the detection box. The DB[11] network, which performs well in ICDAR2015 public dataset, is the prime structure adopted in this stage. From the comparison in Table 1, it can be concluded that the performance of DB is slightly better than that of East.

DB is different from the previous network in dealing with the threshold. Instead of calculating a fixed value, DB exploits the deep network to learn the distribution of a threshold graph, which is expert in separating the background from the foreground. However, such an operation brings the problem that the gradient is non-differentiable while training, so a differentiable binary function is constructed to approximate the threshold learning process. Thus, a similar distribution is obtained, and the robustness of the model to the threshold is greatly improved.

$$\hat{B}_{i,j} = \frac{1}{1 - e^{-k(P_{i,j} - T_{i,j})}} \quad \ldots\ldots\ldots \quad (1)$$

Where B is the approximate binary graph, T is the adaptive threshold graph learned by the network, K is the amplification factor, often taking the empirical value 50.

3.3. Text Direction Classification

The text direction classifier is responsible for detecting the reading order of the text. Considering that the rotation direction of most text proposals in store recognition is close horizontal and vertical, and in order to simplify

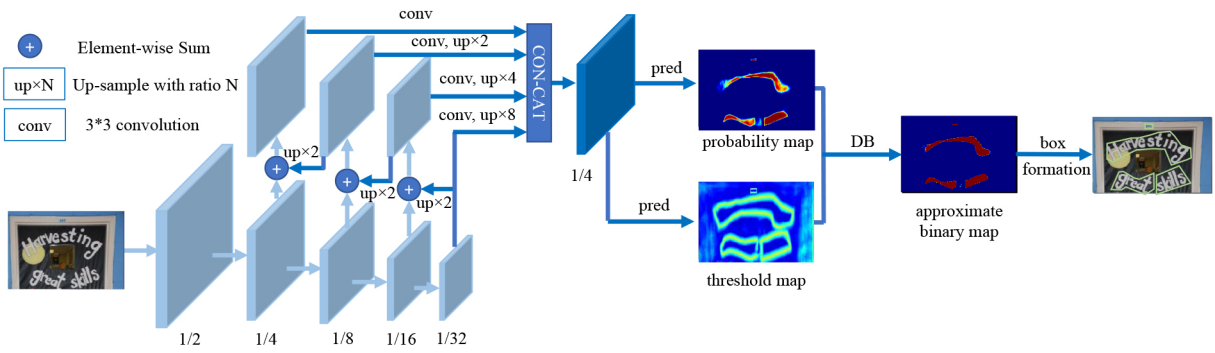

Fig. 1. Architecture of DBNet[11], where "pred" consists of a 3×3 convolutional operator and two de-convolutional operators with stride 2. The "1/2", "1/4", ... and "1/32" indicate the scale ratio compared to the input image.

the model and speed up the prediction, the output of the direction classifier used in this model is divided into four categories: 0°, 90°, 180° and 270°. Experiments show that the detection of horizontal and vertical text already meets most of the scene requirements.

3.4. Text Recognition

Scene text recognition aims at decoding the detected or cropped image regions into character sequences. The text recognizer is responsible for text extraction from images line by line according to the text proposals that has been annotated. Since text extraction involves the processing of variable length sequence information, the cross-network structure of CNN and RNN is usually employed for training. For Chinese context recognition, commonly used strategies are CTC and Attention mechanism.

In terms of effectiveness, the recognition effect of CTC in the general scene is better than that of Attention, because there are abundant common characters in the Chinese dictionary, which is different from English grammar based on letters, so if the training samples are insufficient, it is difficult to mine the sequence relationship between these characters, and the advantage of Attention model cannot be reflected. Attention's serial decoding structure limits the prediction speed, while the CTC network structure is more efficient and has more advantages in prediction speed.

For the text recognizer, we adopt the network CRNN[12] based on CTC, and after comparing with network Rosetta, it is found that the performance of CRNN is slightly better than Rosetta, and the effect of using ResNet34-vd as the backbone presents the best, as shown in Table 2. CRNN is an effective and light-structured model that handles variable length sequences, not limited to any predefined dictionary. It is very practical for application scenarios such as sequence recognition.

The architecture of CRNN consists of three modules, including the convolution layer, circulation layer and conversion layer; In the upper layer of convolution layer, the RNN network is responsible for predicting each frame information of feature sequence; Although CRNN is composed of different networks, it uses the same loss function

Table 2. Rosetta vs. CRNN.

Model	Backbone	Precision
Rosetta	ResNet34-vd	80.90%
Rosetta	MobileNetV3	78.05%
CRNN	ResNet34-vd	82.76%
CRNN	MobileNetV3	79.97%

for training, which greatly reduces the cost he complexity of parameters tuning during training procedure. The network structure of CRNN is shown in Figure 3.

3.5. Semantic Recognition

Semantic recognition, also known as text categorization, is a classical problem in natural language processing, which aims to assign labels or tags to textual units such as sentences, queries, paragraphs, and documents[13]. The workflow of text classification includes text pre-processing, text representation, feature extraction and classifier training[14]. This paper focuses on the combination of RNN and CNN. Considering that the given dataset has small size, a pre-trained ERNIE model trained on other open-source datasets is introduced, and particularly, we use deep learning network structure for semantic classification of Chinese text. ERNIE effectively maintains a better effect in recognizing text in a new scene, indicating it is suitable for the requirements of store classification task.

3.5.1. CNN - Long Text Processing

The key process of CNN is to obtain the output sentence embedding through the convolutional layer, the maximum pooling layer and the nonlinear activation layer, and then send it to the Softmax classifier. The decision of classification operated by CNN is directly related to its kernel, where the sliding window incorporates the impact of the text before and after a single word segment and thus performs well on the classification task of long texts.

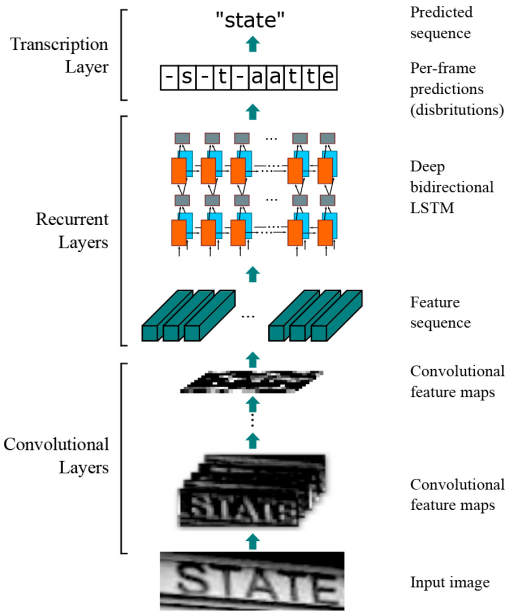

Fig. 2. Network architecture of CRNN[12]. The architecture consists of three parts: 1) convolutional layers, which extract a feature sequence from the input image; 2) recurrent layers, which predict a label distribution for each frame; 3) transcription layer, which translates the per-frame predictions into the final label sequence.

3.5.2. RNN – Classification With Memory

Due to the lack of the ability to recover continuity in CNN architecture, the gated neural network (GRU) and Long Short-Term Memory networks (LSTM) are introduced to process context in a memorable way.

LSTM extends the memory of the network structure for the previous text by introducing the structure of the "Forget Gate". In LSTM, the essence of the tokens' information shall be "memorized" sequentially and iteratively, which is important for the classification, such as entity names, verbs, etc. Compared with LSTM which employs four gate structures, GRU simply lowers the calculation cost through only one gate control. .

3.5.3. ERNIE - Pre-trained Transfer Model

ERNIE pre-trains model on both large text corpora and knowledge graph. The pre-training inputs include contextual-pair data based on conversation sentences collected from websites based on the semantic knowledge graph derived by Baidu's lexical analysis technique according to text content with different granularities such as different words, phrases, and entities. Then the plaintext is mapped to the label according to the dictionaries from external database, and thus, the text content is encoded. Next, we perform random mask operations to obtain training set data according to the segmentation boundary[15].

First, three different forms of masking operations are performed on the original datasets for the word segmentation part. These masking operations are articulated as following to improve the masking mechanism (Fig.6).

Text	小	辣	椒	烧	菜	馆
Basic-level Masking	[mask]	辣	椒	[mask]	菜	[mask]
Entity-level Masking	小	[mask]	[mask]	[mask]	[mask]	馆
Phrase-level Masking	小	辣	椒	[mask]	[mask]	[mask]

Fig. 3. An example of masking.

Basic-level- randomly selects a word in the text. Phrase level- mask some words in the text and then let the model predicts these words. Entity-level- mask entities including the name of a person, organization and product.[16].

As shown in Figure 4, the ERNIE model architecture is composed of two modules:

The low-level T-encoder is responsible for capturing lexical and semantic information from the input tags.The high-level K-Encoder is responsible for integrating the external token-oriented knowledge information into the lower-level text information. N represents the number of layers of the T-Encoder, and M indicates the number of layers of the K-Encoder.

Given a pre-defined word sequence $\{w_1,...,w_n\}$ and its corresponding entity sequence $\{e_1,...,e_m\}$, the text encoder first fuses the token embedding, segment embedding, and position embedding of each token to obtain the input embedding.The vocabulary and semantic features are calculated as following:

$$\{w_1,...,w_n\}' = \text{T-encoder}(\{w_1,...,w_n\}) \quad . \quad . \quad . \quad (2)$$

$$\{e_1,...,e_m\}' = \text{T-encoder}(\{e_1,...,e_m\}) \quad . \quad . \quad . \quad (3)$$

The knowledge encoder K-Encoder is composed of stacked aggregators, fusing heterogeneous features (different length of the segments, various sentence formations, etc.). In the i_{th} aggregator, the token embedding and entity embedding input from the previous aggregators are computed with self-attention method. The entity sequence is performed the same.

$$\{\tilde{w}_1^{(i)},...,\tilde{w}_n^{(i)}\}' = \text{MH\_ATT}(\{\tilde{w}_1^{(i-1)},...,\tilde{w}_n^{(i-1)}\}') \quad (4)$$

Implementation of the mutual integration of token and entity sequence are performed through the information fusion layer (i_{th} aggregator):

$$h_j = \sigma(\tilde{W}_t^{(i)}\tilde{w}_j^{(i)} + \tilde{W}_t^{(i)}\tilde{e}_k^{(i)} + \tilde{b}^{(i)}) \quad . \quad . \quad . \quad . \quad . \quad (5)$$

$$\tilde{w}_t^{(j)} = \sigma(\tilde{W}_t^{(i)}h_j + \tilde{b}_t^{(i)}) \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (6)$$

$$\tilde{e}_k^{(i)} = \sigma(\tilde{W}_e^{(i)}h_j + \tilde{b}_e^{(i)}) \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (7)$$

h_j represents the internal hidden layer state after integrating token and entity, $\sigma(\cdot)$ represents the nonlinear function mapper which uses GELU.

For the token with no corresponding entity, the information fusion layer directly calculates the output embedding:

$$h_j = \sigma(\tilde{W}_t^{(i)}\tilde{w}_j^{(i)} + \tilde{b}^{(i)}) \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (8)$$

$$\tilde{w}_j^{(i)} = \sigma(\tilde{W}_t^{(i)}h_j + \tilde{b}_t^{(i)}) \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (9)$$

Table 3. Impact of different masking levels in pre-training stage

Pre-train dataset size	Mask strategy	Dev. accuracy	Test. accuracy
10% of all	word-level (Chinese)	77.70%	76.80%
10% of all	word-level& phrase-level	78.30%	77.30%
10% of all	word-level& phrase-level& entity-level	78.70%	77.60%
All	word-level& phrase-level& entity-level	79.90%	78.40%

In the pre-training stage, for a given token sequence and corresponding entity sequence, this model expresses the aligned entity distribution as:

$$p(e_j|w_t) = \frac{\exp(\text{linear}(w_i^o)e_j)}{\sum_{k=1}^{m}\exp(\text{linear}(w_i^o)e_k)} \quad . \quad . \quad . \quad . \quad . \quad (10)$$

The above formula is to calculate the cross-entropy loss function of the encoder.

Table 4. Comparison of the precision of validation sets.

Text Recognition Model	Text Classification Model		Precision
CPTN	CNN		66%
	RNN-LSTM		55.79%
	RNN-GRU		56.65%
	ERNIE		85%
PGNet	CNN		67%
	RNN-LSTM		54.39%
	RNN-GRU		57.98%
	ERNIE	Original Dataset	95.10%
		Cross Validation	93.20%
DB+CRNN	CNN		80%
	RNN-LSTM		58.37%
	RNN-GRU		37.34%
	ERNIE	Original Dataset	98.22%
		Cross Validation	96.44%

4. Analysis and Conclusion

From Table 4, For text detection and recognition models, under the same text recognition model, the DB+CRNN model effectively improves the accuracy of text recognition compared with the CPTN and PGNet model, thereby improving the performance of the text classification model. It is not difficult to find that DB+CRNN network significantly improves the recognition accuracy of image samples with text rotating, and has better detection and recognition performance on other small characters.

For the semantic recognition model, ERNIE with pre-training and external knowledge significantly improves the classification accuracy as well as generalization ability. The applicability of ERNIE to the store scene classification task is illustrated and ERNIE through pre-training and transfer learning has a better performance.

4.1. Dataset Expansion

Due to the small capacity of the DC official dataset, it is difficult to increase the accuracy rate to more than 70%, even if the method of text recognition enhancement is included, indicating the problem lays in lack of samples leading to a relatively weak generalization ability. Based on this feature, expansion of the input dataset and accession of screened images in the text detection public dataset for training are presented, which including Chinese street view text recognition, Chinese document text recognition, ICDAR2019-LSVT, ICDAR2017 -RCTW-17, ICDAR2019-ArT. Making use of Paddle's pre-training framework for fine-tuning greatly improves the diversity of the dataset resource, leading to more robust model performance.

4.2. Pre-training and transfering

According to the experiment data, Neither of CNN, GRU or LSTM can satisfy the accuracy required. Due to the particularity of store scene text, pre-processing of the model is limited to single-word classification which does not have appropriate word segmentation and is lack of the understanding of the text context. Thus, we choose ERNIE that is trained with massive data so that its ability of feature extraction suits such text task.

4.3. Error sample analysis

When analyzing the final classification results, this multi-stage model still has room for improvement in text detection and recognition and text classification.

Due to the limitation that the text detection model is merely available on horizontal and vertical texts, a reasonable explanation is that introduction of multiple template datasets for matching, including different swashes and handwritten fonts, is of great essence. However, for some misleading picture text information, the current model cannot 'intelligently' judge its correct classification. (See form Fig.6)

Additionally, subsequent optimization of the model should include more image information, such as food, beverages, etc., then fuse the multi-modality features. However, this method might suffer from complexity of training and also raise the difficulty to extract the truly important features.

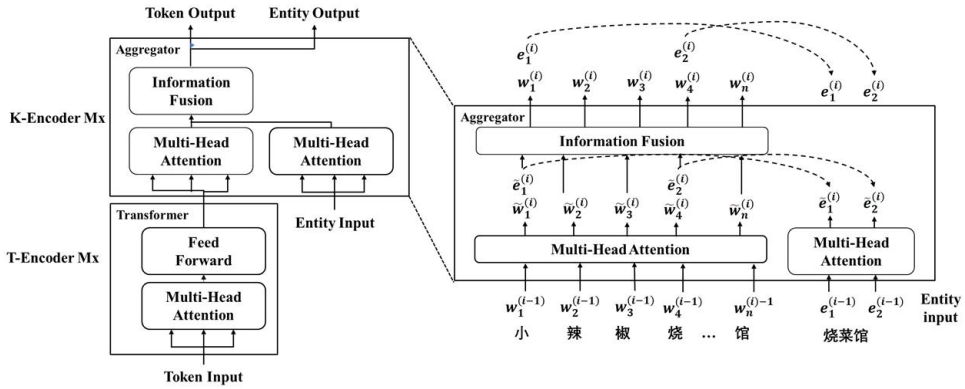

Fig. 4. The structure of ERNIE: The information fusion layer has two kinds of input: Token embedding, and the concatenation of token embedding and entity embedding. New token embedding and entity embedding of the next layer are the output after information fusion.

[17]

Fig. 5. Swash recognition example (Correct text: "Hanshu" detection text: None) .

Fig. 6. Misleading text example (Correct classification: Restaurant; detection text: Service) .

4.4. Conclusion

In this paper, a multi-stage model based on a neural network is proposed, which extracts scene context as the main feature. Classification tasks are divided into three sub-modules: text detection, text recognition, and semantic recognition. It has been proved by a large number of experiments that DBNet + CRNN + Ernie architecture performs best compared with previous networks, with an accuracy of more than 90%. It is noteworthy that the model is robust enough to avoid noise interference which is sufficient for scene identifications.

References:

[1] Lee, C.-Y.; and Osindero, S. 2016. Recursive recurrent nets with attention modeling for ocr in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2231–2239.

[2] Sun, Y.; Zhang, C.; Huang, Z.; Liu, J.; Han, J.; and Ding, E.2018. TextNet: Irregular Text Reading from Images with an End-to-End Trainable Network. In Asian Conf. Comp. Vis.(ACCV), 83–99. Springer.

[3] Lyu, P.; Liao, M.; Yao, C.; Wu, W.; and Bai, X. 2018. Mask TextSpotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. In Eur. Conf. Comp. Vis. (ECCV), 67–83.

[4] Feng, W.; He, W.; Yin, F.; Zhang, X.-Y.; and Liu, C.-L. 2019. TextDragon: An End-to-End Framework for Arbitrary Shaped Text Spotting. In IEEE Int. Conf. Comp. Vis. (ICCV), 9076–9085.

[5] Wang, P., et al. "PGNet: Real-time Arbitrarily-Shaped Text Spotting with Point Gathering Network." (2021).

[6] PEI Zhili, A Runa, Jiang Mingyang, et al. Text Classification based on convolutional Neural networks [J]. Journal of Inner Mongolia University for Nationalities (Natural Science edition), 2019(3).

[7] Vu N T , Adel H , Gupta P , et al. Combining Recurrent and Convolutional Neural Networks for Relation Classification[J]. 2016.

[8] Mikolov, Tomas & Sutskever, Ilya & Chen, Kai & Corrado, G.s & Dean, Jeffrey. (2013). Distributed Representations of Words and Phrases and their Compositionality. Advances in Neural Information Processing Systems. 26.

[9] Dai A M , Le Q V . Semi-supervised Sequence Learning[J]. MIT Press, 2015.

[10] Du Y , Li C , Guo R , et al. PP-OCR: A Practical Ultra Lightweight OCR System[J]. 2020.

[11] Liao, M.; Wan, Z.; Yao, C.; Chen, K.; and Bai, X. 2020.Real-Time Scene Text Detection with Differentiable Binarization. In AAAI, 11474–11481. 1, 5

[12] Shi, B.; Bai, X.; and Yao, C. 2016. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE transactions on pattern analysis and machine intelligence 39(11): 2298–2304. 1, 10

[13] Chengtao, Cai D , Guo. CNN-SELF-ATTENTION-DNN ARCHITECTURE FOR MANDARIN RECOGNITION[C], 2019.

[14] Tang Lin. Research on Key Technologies of WeChat Subscription Number Text Acquisition and Preprocessing [J]. Journal of Chifeng University (Natural Science Edition), 2019(11).

[15] Zhang Z , Han X , Liu Z , et al. ERNIE: Enhanced Language Representation with Informative Entities[C]// Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019

[16] Niu Yuting, Chen Boqi, Chen Bin. Chinese text classification based on improved ERNIE-DPCNN model [J]. Journal of Jiangsu Normal University (Natural Science Edition), 201,39(01):47-52.

[17] D. Cheng, R. Ortega, E. Panteley, On port controlled Hamiltonian systems. Advanced Robust and Adaptive Control –Theory and Applications, D. Cheng, Y. Sun, T. Shen, H. Ohmori, Eds. Beijing: Tsinghua University Press, 2005: 3-16.